Unsupervised Hard Example Mining from Videos for Improved Object Detection

SouYoung Jin*, Aruni RoyChowdhury*, Huaizu Jiang, Ashish Singh,
Aditya Prasad, Deep Chakraborty, and Erik Learned-Miller

College of Information and Computer Sciences, University of Massachusetts, Amherst
{souyoungjin,arunirc,hzjiang,ashishsingh,
aprasad,dchakraborty,elm}@cs.umass.edu

Abstract. Important gains have recently been obtained in object detection by using training objectives that focus on *hard negative* examples, i.e., negative examples that are currently rated as positive or ambiguous by the detector. These examples can strongly influence parameters when the network is trained to correct them. Unfortunately, they are often sparse in the training data, and are expensive to obtain. In this work, we show how large numbers of hard negatives can be obtained *automatically* by analyzing the output of a trained detector on video sequences. In particular, detections that are *isolated in time*, i.e., that have no associated preceding or following detections, are likely to be hard negatives. We describe simple procedures for mining large numbers of such hard negatives (and also hard *positives*) from unlabeled video data. Our experiments show that retraining detectors on these automatically obtained examples often significantly improves performance. We present experiments on multiple architectures and multiple data sets, including face detection, pedestrian detection and other object categories.

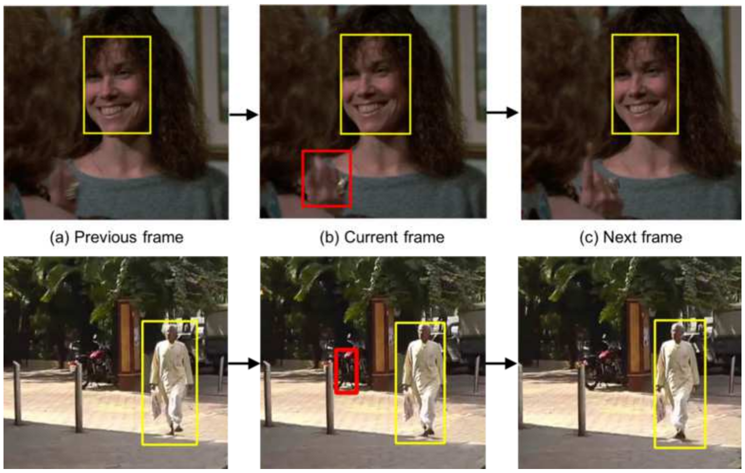
Keywords: object detection, face detection, pedestrian detection, semi-supervised learning, hard negative mining.

1 Introduction

Detection is a core computer vision problem that has seen major advances in the last few years due to larger training sets, improved architectures, end-to-end training, and improved loss functions [42,41,13,67]. In this work, we consider another direction for improving detectors – by dramatically expanding the number of hard examples available to the learner. We apply the method to several different detection problems (including face and pedestrian), a variety of architectures, and multiple data sets, showing significant gains in a variety of settings.

Many discriminative methods are more influenced by challenging examples near the boundary of a classifier than easy examples that have low loss. Some classifiers, such as support vector machines, are completely determined by examples near the classifier boundary (the “support vectors”) [45]. More recent

[*] Authors contributed equally



(a) Previous frame (b) Current frame (c) Next frame

Fig. 1: **Detector flicker in videos.** Three consecutive frames from a video are shown for face and pedestrian detection. On the top row, the boxes show face detections from the Faster R-CNN [42] (trained on WIDER face) [61,25]. On the bottom row are detections from the same detector trained on the Caltech pedestrian dataset [12]. Yellow boxes show true positives and red boxes show false positives. For the true positives, the same object is detected in all three frames whereas for the false positives, the detection is *isolated* – it occurs neither in the previous nor the subsequent frame. These detections that are "isolated in time" frequently turn out to be false positives, and hence provide important sources of hard negative training data for detectors.

techniques that emphasize examples near the boundary include general methods such as *active bias* [8], which re-weights examples according to the variance of their posteriors during training. In the context of class imbalance in training object detectors, on-line hard example mining (OHEM) [46] and the *focal loss* [33] were designed to emphasize hard examples.

In this paper, we introduce simple methods for automatically mining both hard negatives and hard positives from videos using a previously trained detector. To illustrate, Figure 1 shows a sequence of consecutive video frames from two videos containing a face and a pedestrian respectively. The results of the Faster R-CNN detector (trained for each class) run on each frame are marked as rectangles, with true positives as yellow boxes and false positives as red boxes. Notice that false positives are neither preceded nor followed by a detection. We refer to such isolated-in-time detections as **detector flickers** and postulate that these are usually caused by false positives rather than true positives.[1] This hypothesis stems from the idea that a false positive, caused by something that usually does not look like a face (or other target object), such as a hand, only

[1] Note we are *not* claiming that most false positives will be isolated, but only that flickers are likely to be false positives, a very different statement.

momentarily causes a detector network to respond positively, but that small deviations from these hard negatives will likely not register as positives. Similar observations can be found in the literature on adversarial examples, where many adversarial examples have been shown to be "unstable" with respect to minute perturbations of the image [36,37,3]. In addition, leveraging the continuity of labelling across space and time has a long history in computer vision. Spatial label dependencies are widely modeled by Markov random fields [18] and conditional random fields [53], while the smoothness of labels across time is a staple of tracking methods and other video processing algorithms [50,28,59].

As our experiments show, a large percentage of detector flickers are indeed false positives, and more importantly, they are hard negatives, since they were identified incorrectly as positives by the detector. Such an *automatically generated training set* of hard negatives can be used to fine-tune a detector, often leading to improved performance. Similar benefits are gained from fine-tuning with *hard positives*, which are obtained in an analogous fashion from cases where a consistently detected object "flickers off" in an isolated frame. While these flickers are relatively rare, it is inexpensive to run a modern detector on many hours of unlabeled video, generating essentially unlimited numbers of hard examples. Being an unsupervised process, training sets gathered automatically in this fashion do include some noise. Nevertheless, our experiments show that significant improvements can be gleaned by retraining detectors using these noisy hard examples. An alternative to gathering such hard examples automatically is, of course, to obtain them manually. However, the rarity of false positives for modern detectors makes this process extremely expensive. Doing this manually requires that every positive detection be examined for validity. With typical false positive rates around one per 1000 images, this process requires the examination of 1000 images per false positive, making it prohibitively expensive.

2 Related Work

Convolutional neural networks have recently been applied to achieve state-of-the-art results in object detection [20,19,21,41,40,34,6,32]. Many of these object detectors have been re-purposed for other tasks such as face detection [39,29,60,15], [31,63,62,25,57,23,66] and pedestrian detection [64,14,6,7,22,30,65], achieving impressive results [24,61,12].

Hard negatives in detection. Massive class imbalance is an issue with sliding-window-style object detectors — being densely applied over an image, such models see far more "easy" negative samples from background regions than positive samples from regions containing an object. Some form of hard negative mining is used by most successful object detectors to account for this imbalance [10,11,16,20,19,21,46,64,33,55,51]. Early approaches include *bootstrapping* [52] for training SVM-based object detectors [10,16], where false positive detections were added to the set of background training samples in an incremental fashion. Other methods [44,11] apply a pre-trained detector on a larger dataset to mine false positives and then re-train.

Hard negative mining has also improved the performance of deep learning based models [47,35,19,46,64,55,33]. Shrivastava *et al.* [46] proposed an *Online Hard Example Mining* (OHEM) procedure,training using only high-loss region proposals. This technique, originally applied to the Fast R-CNN detector [19], yielded significant gains on the PASCAL and MS-COCO benchmarks. Lin *et al.* [33] propose the *focal loss* to down-weight the contribution of easy examples and train a single-stage, multi-scale network [32]. The A-Fast-RCNN [56] does adversarial generation of hard examples using occlusions and deformations. While similar to our work, our model is trained with hard examples from *real* images and variations are not limited to occlusion and spatial deformations. Zhang *et al.* [64] show that effective bootstrapping of hard negatives, using a boosted decision forest [17,2], significantly improves over a Faster R-CNN baseline for *pedestrian detection*. Recent *face detection* methods, such as Wan *et al.* [55] and Sun *et al.* [51], have also used the bootstrapping of hard negatives to improve the performance of CNN-based detectors — a pre-trained Faster R-CNN is used to mine hard negatives; then the model is re-trained. However, these methods require a human-annotated dataset of suitable size. Our unsupervised approach does not rely upon bounding-box annotations and thus can be trained upon potentially unlimited data.

Semi-supervised learning. Using mixtures of labeled and unlabeled data is known as *semi-supervised learning* [4,9,58]. Rosenberg *et al.* [43] ran a trained object detector on unlabeled data and then trained on a subset of this noisy labeled data in an incremental re-training procedure. In Kalal *et al.* [27], constraints based on video object trajectories are used to correct patch labels of a random forest classifier; these corrected samples are used for re-training. Tang *et al.* [54] adapt still-image object detectors to video by selecting training samples from unlabeled videos, based on the consistency between detections and tracklets, and then follow an iterative procedure that selects the easy examples from videos and hard examples from images to re-train the detector. Rather than adapting to the video domain, we seek to improve detector performance on the source domain by selecting hard examples from videos. Singh *et al.* [48] gather discriminative regions from weakly-labeled images and then refine their bounding-boxes by incorporating tracking information from weakly-labeled videos.

3 Mining Hard Examples from Videos

This section discusses methods for automatically mining hard examples from videos, including data collection (Sec. 3.1), our hard negative mining algorithm (Sec. 3.2), statistics of recovered hard negatives (Sec. 3.3) and extension to hard positives (Sec. 3.4). Details of re-training the detector on these new samples are in the Experiments section (Sec. 4.1).

3.1 Video Collection

To mine hard examples for face detection, we used 101 videos from sitcoms, each with a duration of 21-25 minutes and a full-length movie of 1 hour 47 minutes,

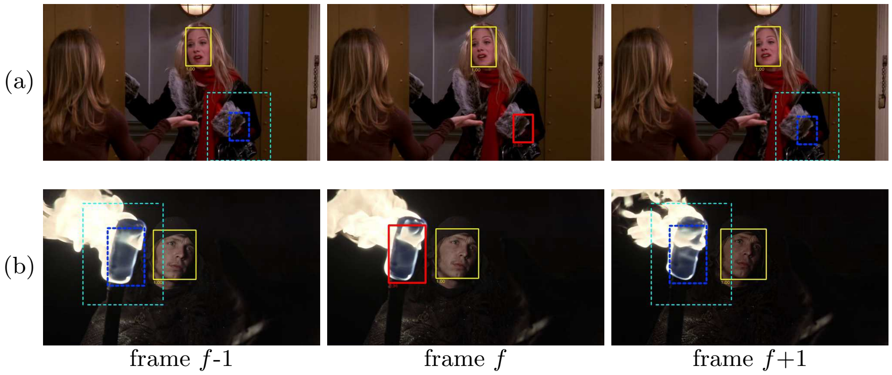


Fig. 2: **Mining hard negatives from detector-flicker.** The solid boxes denote detections, and the dashed boxes are associated with the tracking algorithm. Given all of the high-confidence **face detections** in a video (**yellow** boxes), the proposed algorithm generates a **tracklet** (**blue** *dashed* boxes) for the **current detection** (**red** box in frame f) by applying template matching within the **search regions** of the adjacent frames (**cyan** *dashed* boxes). As there are no matching detections in adjacent frames for the current detection (*i.e.* no yellow box matches the blue dashed boxes in frames f-1 or f+1), it is correctly considered to be an "isolated detection" and added to the set of ***hard negatives***. The remaining detections in frame f, which are temporally consistent, are added to the set of ***pseudo-positives***.

"Hannah and her sisters" [38]. Further, we performed YouTube searches with keywords based on: *public address*, *debate society*, *orchestra performance*, *choir practice* and *courtroom*, downloading 89 videos of durations ranging from 10 to 25 minutes. We obtained videos that were expected to feature a large number of human faces in various scenes, reflecting the everyday settings of our face benchmarks. Similarly, for pedestrian detection, we collected videos from YouTube by searching with the two key phrases: *driving cam videos* and *walking videos*. We obtained 40 videos with an average duration of about 30 minutes.

3.2 Hard Negative Mining

Running a pre-trained face detector on every frame of a video gives us a large set of detections with noisy labels. We crucially differ here from recent bootstrapping approaches [55,51] by (a) using large amounts of *unlabeled* data available on the web instead of relying only on the limited fully-supervised training data from WIDER Face [61] or Caltech Pedestrians [12], and (b) having a novel filtering criterion on the noisy labels obtained from the detector that retains the hard negative examples and minimizes noise in the obtained labels.

The raw detections from a video were thresholded at a relatively high confidence score of 0.8. For every detection in a frame, we formed a short tracklet by performing template matching in adjacent frames, within a window of ±5 frames — the bounding box of the current detection was enlarged by 100 pixels and this region was searched in adjacent frames for the best match using normalized cross correlation (NCC). To account for occlusions, we put a threshold on the NCC similarity score (set as 0.5) to reject cases where there was a lot of appearance-change between frames. Now in each frame, if the maximum intersection-over-union (IoU) between the tracklet prediction and detections in the adjacent frames was below 0.2, we considered it to be an isolated detection resulting from **detector flicker**. These isolated detections were taken as ***hard negatives***. The detections that *were* found to be consistent with adjacent frames were considered to have a high probability of being true predictions and were termed ***pseudo-positives***. For the purpose of creating the re-training set, we kept only those frames that had at least one pseudo-positive detection in addition to one or more hard negatives. Illustrative examples of this procedure are shown in Figure 2, where we visualize only the previous and next frames for simplicity.

3.3 Results of Automatic Hard Negative Mining

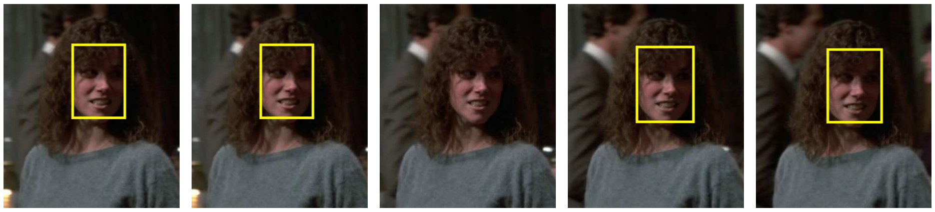
Our initial mining experiments were performed using a standard Faster R-CNN detector trained on WIDER Face [61] for faces and Caltech [12] for pedestrians. We collected 13,888 video frames for faces, where each frame contains at least one pseudo-positive and one hard negative (detector flicker). To verify the quality of our automatically mined hard negatives, we randomly sampled 511 hard negatives for inspection. 453 of them are true negatives, while 16 samples are true positives, and 42 samples are categorized as *ambiguous*, which correspond extreme head pose or severe occlusions. The precision for true negatives is 88.65% and precision for true negatives plus *ambiguous* is 96.87%.

For pedestrians, we collected 14,967 video frames. We manually checked 328 automatically mined hard negatives, where 244 of them are true negatives and 21 belong to *ambiguous*. The precision for true negatives is 74.48% and precision for true negatives plus *ambiguous* is 82.18%.

To further validate our method on an existing fully-annotated video dataset, we used the Hannah dataset [38], which has every frame annotated with face bounding boxes. Here, out of 234 mined hard negatives, 187 were true negatives, resulting in a precision of 79.91%. We note that the annotations on the Hannah movie are not always consistent and involve a significant domain shift from WIDER. Considering the fact no human supervision is provided, the mined face hard negatives are consistently of high quality across various domains.

3.4 Extension to Hard Positive Mining

In principle, the same concept for using detector flickers can be directly applied to obtaining ***hard positives***. The idea is to look for "off-flickers" of a detector in a video tracklet – given a series of detections of an object in a video, such as a face, we can search for single frames that have no detections but are surrounded



frame f-2 frame f-1 frame f frame f+1 frame f+2

Fig. 3: **Hard positive samples.** Given a sequence of video frames, the face of the actor is consistently detected except at frame f. Such isolated "off-flickers" can be harvested in an unsupervised fashion to form a set of *hard positives.*

by detections on either side. Of course, these could be caused by short-duration occlusions, for example, but a large percentages of these "off-flickers" are hard positives, as in Fig. 3. We generate tracklets using the method from [26] and show results incorporating hard positives on pedestrian and face detection in the experiments section. The manually calculated purity over 300 randomly sampled frames was 94.46% for faces and 83.13% for pedestrians.

4 Experiments

We evaluate our method on face and pedestrian detection and perform ablation studies analyzing the effect of the hard examples.For pedestrians, we show results on the Caltech dataset [12], while for face detection, we show results on the WIDER Face [61] dataset.

The Caltech Pedestrian Dataset [12] consists of videos taken from a vehicle driving through urban traffic, with about 350k annotated bounding-boxes from 250k video frames.

The WIDER dataset consists of 32,203 images having 393,703 labeled faces in challenging situations of scale, pose and occlusion. The evaluation set of WIDER is divided into *easy*, *medium*, and *hard* sets according to the detection scores of object proposals from EdgeBox [67]. From easy to hard, the faces get smaller and more crowded.

4.1 Retraining Detectors with Mined Hard Examples

We experimented with two ways to leverage our mined *hard negative* samples. In our initial experiments, a single mini-batch is formed by including one image from the original labeled training dataset and another image sampled from our automatically-mined hard negative video frames. In this way, positive region proposals are sampled from the original training dataset image, based on manual annotation, while negative region proposals are sampled from both the original dataset image and the mined hard negative video frame. Thus, we can *explicitly* force the network to focus on the hard negatives from the mined video frame.

However, this method did not produce better results in our initial experiments. An alternate approach was found to be more effective – we simply provided the *pseudo-positives* in the mined video frames as true object annotations during training and *implicitly* allowed the network to pick the hard-negatives. The inclusion of video frames with *hard positives* is more straightforward – we can simply treat them as additional images with object annotations at training time. The models were fine-tuned with and without OHEM, and we consistently chose the setting that gave the best validation results. While OHEM would increase the likelihood of hard negatives being selected in a mini-batch, it would also place extra emphasis on any mislabels in the hard examples. This would magnify the effect of a small amount of label noise and can in some cases decrease the overall performance.

4.2 Ablation Settings

In addition to the comparisons to the baseline Faster R-CNN detectors, we conduct various ablation studies on the Caltech Pedestrian and WIDER Face datasets to address the effectiveness of hard example mining.
Effect of training iterations. To account for the possible situation where simply training the baseline model longer may result in a gain in performance, we create another baseline by fine-tuning the original model for additional iterations with a lower learning rate, matching the number of training iterations used in our hard example trained models. We refer to this model as "`w/ more iterations`".
Effect of additional video frames. Unlike the baseline detector, our fine-tuned models use additional video frames for training. It's possible that just using the high-confidence detection results on unlabeled video frames as *pseudo-groundtruths* during training is sufficient to boost performance, without correcting the hard negatives using our detector flicker approach. Therefore we train another detector, "`Flickers as Positives`", starting from the baseline model, that takes exactly the same training set as our hard negative model, but where *all* the high-confidence detections on the video frames are used as positive labels.
Effect of automatically mined hard examples. We include the results from our proposed method of considering detector flickers as hard negatives and hard positives separately – "`Flickers as HN`" and "`Flickers as HP`". Finally, we report results from fine-tuning the detector on the union of both types of hard examples (`Flickers as HN + HP`).

4.3 Pedestrian Detection

For our `baseline` model, we train the VGG16-based ***Faster R-CNN*** object detector [42] with OHEM [46] for 150K iterations on the **Caltech Pedestrian** training dataset [12]. We used *all* the frames from set00-set05 (which constitute the training set), irrespective of whether they are flagged as "reasonable" or not by the Caltech meta-data. Following Zhang *et al.* [64], we set the IoU ratio for RPN training to 0.5, while all the other experimental settings are identical to [42]. The number of labeled Caltech images is 128,419 and our mining provides 14,967 hard negative and 42,914 hard positive frames. We fine-tune the baseline model with hard examples and the annotated examples from the Caltech Pedestrian *training* dataset, with a fixed learning rate of 0.0001 for 60K iterations, using OHEM. We evaluate our model on the Caltech Pedestrian testing dataset under the *reasonable* condition.

The ROC curves of various settings of our models are shown in Fig. 4(a). Fine-tuning the existing detector for more iterations gives a modest reduction in log average miss rate, from 23.83% to 22.4%. Using all detections without correcting the hard negatives (`Flickers as Pos`) also gives a small improvement – the extra training data, although noisy, still has some positive contribution during fine-tuning. Our proposed model, fine-tuned with the mined hard negatives (`Flickers as HN`), has a log average miss rate of **18.78%**, which outperforms the `baseline model` by **5.05%**. Fine-tuning with hard positives (`Flickers as HP`) also shows an improvement of **4.39%** over the baseline. Combining both hard positives and hard negatives results in the best performance of **18.72%** log average miss rate.

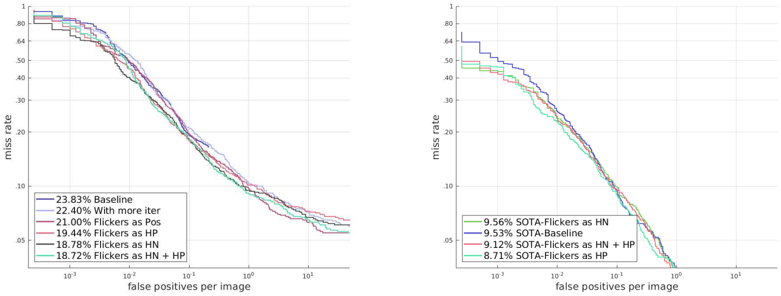
In Figure 4(b) we report results using the state-of-the-art ***SDS-RCNN*** [5] pedestrian detector [2]. Every 3rd frame is sampled from the Caltech dataset for training the original detector [5], and we keep this setting in our experiments. For SDS-RCNN, there are 42,782 labeled training images while the mining gives us 2,191 hard negative and 177,563 hard positive frames. The inclusion of hard negatives in training (`Flickers as HN`) improves the performance of SDS-RCNN in the low False Positives regime compared to the baseline – the detector learns to eliminate a number of false detections, thereby increasing precision, but it also ends up hurting the recall. Including mined hard positives (`Flickers as HP`) we get the best performance of **8.71%** log average miss rate, outperforming the model using both the mined hard negative and positive samples (`Flickers as HP + HN`), which gets 9.12%.

4.4 Face Detection

We adopt the Faster R-CNN framework, using VGG16 as the backbone network. We first train a baseline detector starting from an ImageNet pre-trained model, with a fixed learning rate of 0.001 for 80K iterations using the SGD optimizer, where the momentum is 0.9 and weight decay is 0.0005. For hard negatives, the model is fine-tuned for 50k iterations with learning rate 0.0001. For hard positives, and the combination of both types of hard examples, we train longer for 150k iterations. Following the **WIDER Face** protocol, we report Average Precision (AP) values in Table 1 on the three splits – ‘Easy’, ‘Medium’ and ‘Hard’. OHEM is not used as it was empirically observed to decrease performance.

Fine-tuning the baseline model for more iterations improves performance slightly on the Easy and Medium splits. Naively considering all the high confidence detections as true positives (`Flickers as Positives`) degrades performance substantially across all splits. Hard negative mining, `Flickers as HN`,

[2] Running the authors’ released code from https://github.com/garrickbrazil/SDS-RCNN



(a) (b)

Fig. 4: Results on the **Caltech Pedestrian** dataset [12] in *reasonable* condition. (a) Faster R-CNN results: using hard negative samples (`Flickers as HN`) and hard positive samples (`Flickers as HP`) improve the performance over the baseline in; using a combination of both gives the best performance. (b) State-of-the-art SDS-RCNN results: `Flickers as HN` improves the original SDS-RCNN results only in the low false positive regime, while `Flickers as HP` gives the best results.

slightly outperforms the baseline Faster R-CNN detector (`w/ more iterations`) on the Medium and Hard splits, retaining the same performance of 0.907 AP on the Easy split. Using the mined hard positives, `Flickers as HP`, we observe a significant gain in performance on all three splits. Using both hard positives and hard negatives jointly (`Flickers as HP + HN`) improves over using hard negatives and the baseline, but the improvement is less than the gains from `Flickers as HP`.

For faces, we additionally experimented with the recent RetinaNet [33] detector as a second high-performance baseline model. Unfortunately, inclusion of the unlabeled data hurt performance slightly using this model, despite the reasonably high purity of the mined examples. While the purity of our mined examples is high, it is not perfect. These incorrect samples would be strongly emphasized by the focal loss used in RetinaNet. Thus, it is possible that while RetinaNet outperforms the Faster R-CNN on standard benchmarks, it may be more susceptible to label noise and thus not a good candidate for our method. In the future, we will investigate different values of the focal loss parameter to see whether this can mitigate the effects of label noise.

5 Discussion

In this section, we discuss some further applications and extensions to our proposed hard example mining method.

On the Entropy of the False Positive Distribution. In mining thousands of hard negatives from unlabeled video, we noticed a striking pattern in the hard

Table 1: Average precision (AP) on the validation set of the **WIDER Face** [61] benchmark. Including hard examples improves performance over the baseline, with `HP` and `HP+HN` giving the best results.

		Easy	Medium	Hard
Faster R-CNN	Baseline	0.907	0.850	0.492
	w/ more iterations	0.910	0.852	0.493
	Flickers as Positives	0.829	0.790	0.434
	Ours: Flickers as HN	0.909	0.853	0.494
	Ours: Flickers as HP	**0.921**	**0.864**	0.492
	Ours: Flickers as HP + HN	**0.921**	**0.864**	**0.497**

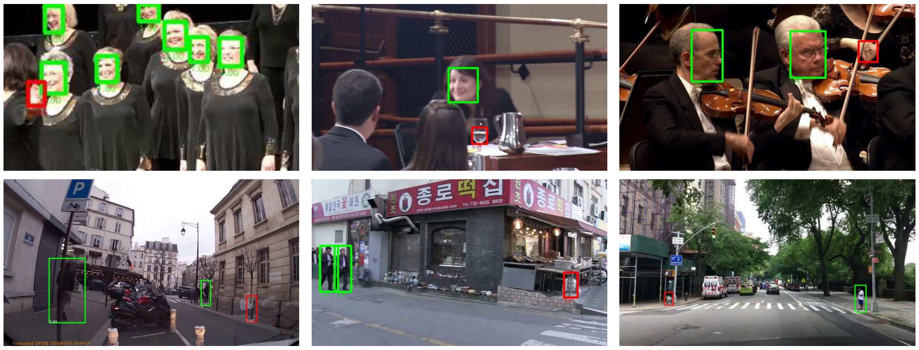


Fig. 5: **Examples of hard negatives.** Visualization of mined hard negatives for faces (*top row*) and pedestrians (*bottom row*). Red boxes denote the "detection-flicker cases" among the high confidence detections (green boxes).

negatives of face detectors. A large percentage of false positives were generated by a few types of objects. Specifically, a large percentage of hard negatives in face detectors seem to stem from human hands, ears, and the torso/chest area. Since it appears that a large percentage of the false positives in face detection are the result of a relatively small number of phenomena, this could explain the significant gains realized by modeling hard negatives. In particular, characterizing the distribution of hard negatives, and learning to avoid them, may involve a relatively small set of hard negatives.

Effect of Domain Shift on FDDB. The FDDB dataset [24] is comprised of 5,171 annotated faces in a set of 2,845 images taken from a subset of the Face in the Wild dataset. The images and the annotation style of FDDB have a significant *domain shift* from WIDER Face, which are discussed in Jamal et al. [1]. Fig. 7 compares our method with the Faster R-CNN baseline on FDDB, using the trained models from our experiments on WIDER Face (Sec. 4.4). Although hard negatives reduce false positives (Fig. 7(b)) and hard positives increase recall (Fig. 7(c)), the performance does not consistently improve over the baseline on FDDB. We hypothesize that the large amounts of new training data result in

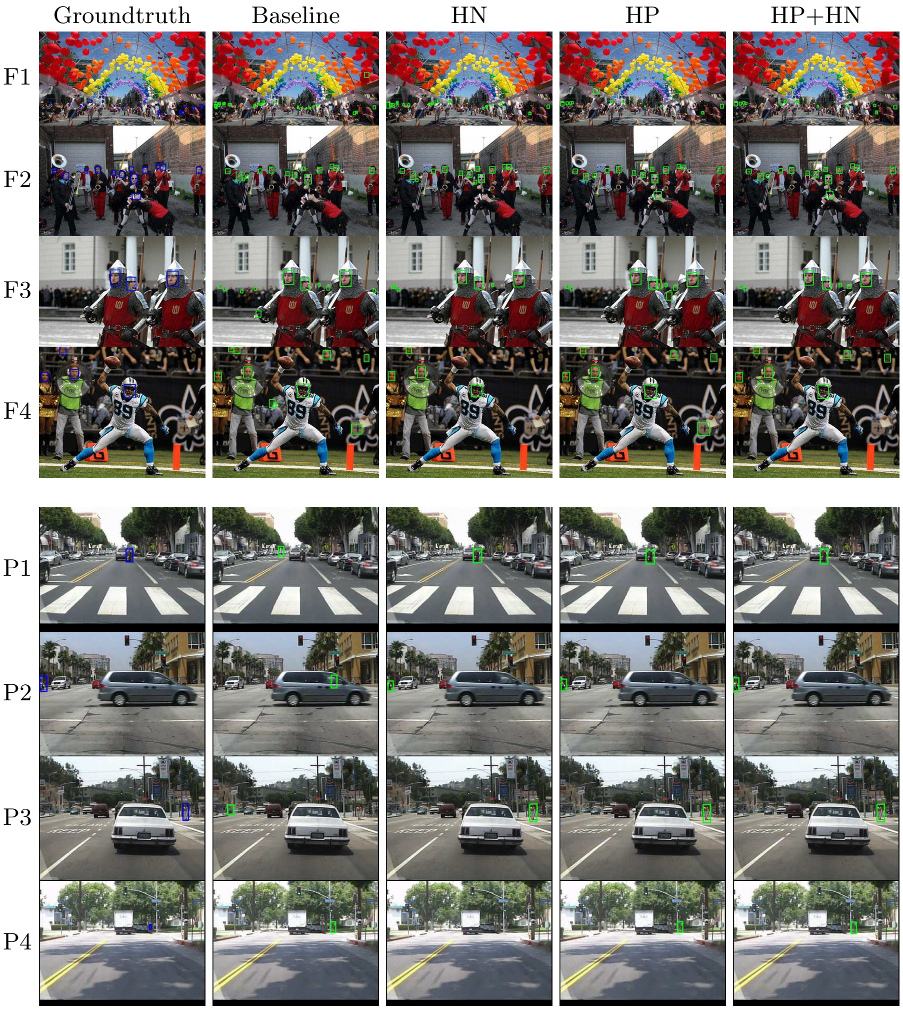
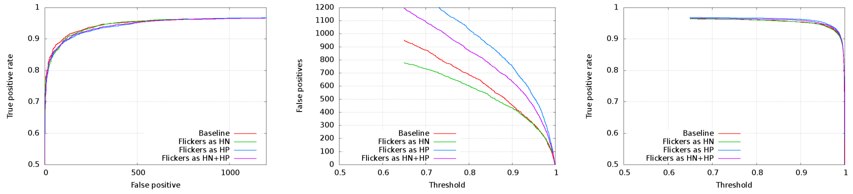


Fig. 6: **Qualitative comparison.** Faster R-CNN detections for faces (F1-4) and pedestrians (P1-4).The detector fine-tuned with hard negatives (HN) reduces false positives compared to the Baseline (F-1,3,4; P-1,2,3), but can sometimes lower the recall (P4). Hard positives (HP) increases recall (F2, P4) but can also introduce false positives (F4). Using both (HP+HN) the detector is usually able to achieve a good balance.

shifting the original detector further away from the target FDDB domain, and this domain shift leads to a loss in performance. This may not have hurt our performance as much on WIDER Face because the domain shift between the relatively unconstrained WIDER images and our videos downloaded from YouTube was not severe enough to subsume the advantages from the hard examples.



(a) (b) (c)

Fig. 7: Results on **FDDB**. (a) ROC curves comparing our hard example methods with the baseline Faster R-CNN detector; (b-c) separate plots showing False Positives and True Positive Rate with varying thresholds on detection score.

Extension to Other Classes. The simplicity of our approach makes it easily extensible to other categories in a one-versus-rest setting. YouTube is a promising source of videos for various MS-COCO or PASCAL categories; mining hard negatives after that is fully automatic. To demonstrate this, we selected categories from MS-COCO and ran experiments to check if inclusion of hard negatives improves the baseline performance of a Faster R-CNN detector. We used the training method deployed by Sonntag et al.[49], which allows for a convenient fine-tuning of the VGG16-based Faster R-CNN model on specific object classes of the MS-COCO dataset. The method was used to train a Faster R-CNN detector for a specific class vs background, starting from a multi-class VGG16 classifier pre-trained on Image-Net categories. This baseline detector was then used to mine hard negatives from downloaded YouTube videos of that category and then re-trained on the union of the new data and the original labeled training data. We show results for two categories: *dogs* and *trains*. A held out subset of the MS-COCO validation set was used for validating training hyper-parameters and the remainder of the validation data was used for evaluation.

For the *dog* category, the labeled data was divided into train/val/test splits of 3041/177/1521 images. We manually selected and downloaded about 22 hours of dog videos from YouTube. We used the baseline dog detector to obtain detections on about 15 hours (1,296,000 frames at 24 fps) of dog videos. The hard negative mining algorithm was then run at a detector confidence threshold of 0.8. This yielded 2611 frames with at least one hard negative and one positive detection. The baseline model was then fine-tuned for 30k iterations on the union of the labeled MS-COCO data and the hard negatives. The hyper-parameters and best model were selected using a validation set. Similar experiments with *trains* were performed, with train/val/test splits of 2464/157/1281 images. The results are summarized in the Table 2, where inclusion of hard negatives is observed to improve the baseline detector in both cases.

Table 2: Results on augmenting Faster R-CNN detectors with hard negatives for '*dog*' and '*train*' categories on MS-COCO.

Category	**Model**	**Training iterations**	**Training hyperparams**	**Validation set AP**	**Test set AP**
Dog	Baseline	29000	LR : 1e-3 for 10k, 1e-4 for 10k-20k, 1e-5 for 20k-29k	26.9	25.3
	Flickers as HN	22000	LR : 1e-4 for 15k, 1e-5 for 15k-22k	28.1	26.4
Train	Baseline	26000	LR : 1e-3, stepsize: 10k, lr-decay: 0.1	33.9	33.2
	Flickers as HN	24000	LR : 1e-3, stepsize: 10k, lr-decay: 0.1	35.4	33.7

6 Conclusion

This work leverages an existing phenomenon – detector flicker in videos – to mine hard negatives and hard positives at scale in an unsupervised manner. The usefulness of this method for improving an object detector is demonstrated on standard benchmarks for two well-known tasks – face and pedestrian detection, using various detector architectures and supported by several ablation studies. The simplicity of our hard example mining approach makes it widely applicable to a variety of practical scenarios – YouTube is a promising source of videos for almost any category and mining hard examples is a fully automatic procedure.

Acknowledgment

This research is based in part upon work supported by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA) under contract number 2014-14071600010 and in part on research sponsored by the Air Force Research Laboratory and DARPA under agreement number FA8750-18-2-0126. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of ODNI, IARPA, the Air Force Research Laboratory and DARPA or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purpose notwithstanding any copyright annotation thereon.

References

1. Abdullah Jamal, M., Li, H., Gong, B.: Deep face detector adaptation without negative transfer or catastrophic forgetting. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
2. Appel, R., Fuchs, T., Dollár, P., Perona, P.: Quickly boosting decision trees–pruning underachieving features early. In: International Conference on Machine Learning. pp. 594–602 (2013)
3. Athalye, A., Sutskever, I.: Synthesizing robust adversarial examples. arXiv preprint arXiv:1707.07397 (2017)
4. Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: Proceedings of the eleventh annual conference on Computational learning theory. pp. 92–100. ACM (1998)
5. Brazil, G., Yin, X., Liu, X.: Illuminating pedestrians via simultaneous detection & segmentation. arXiv preprint arXiv:1706.08564 (2017)
6. Cai, Z., Fan, Q., Feris, R.S., Vasconcelos, N.: A unified multi-scale deep convolutional neural network for fast object detection. In: European Conference on Computer Vision. pp. 354–370. Springer (2016)
7. Cai, Z., Saberian, M., Vasconcelos, N.: Learning complexity-aware cascades for deep pedestrian detection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3361–3369 (2015)
8. Chang, H.S., Learned-Miller, E., McCallum, A.: Active bias: Training more accurate neural networks by emphasizing high variance samples. In: Advances in Neural Information Processing Systems. pp. 1003–1013 (2017)
9. Chapelle, O., Scholkopf, B., Zien, A.: Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. IEEE Transactions on Neural Networks **20**(3), 542–542 (2009)
10. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR. pp. 886–893 (2005). https://doi.org/10.1109/CVPR.2005.177, http://dx.doi.org/10.1109/CVPR.2005.177
11. Dollár, P., Tu, Z., Perona, P., Belongie, S.: Integral channel features (2009)
12. Dollár, P., Wojek, C., Schiele, B., Perona, P.: Pedestrian detection: A benchmark. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. pp. 304–311. IEEE (2009)
13. Dollár, P., Zitnick, C.L.: Fast edge detection using structured forests. IEEE Trans. Pattern Anal. Mach. Intell. **37**(8), 1558–1570 (2015). https://doi.org/10.1109/TPAMI.2014.2377715, http://dx.doi.org/10.1109/TPAMI.2014.2377715
14. Du, X., El-Khamy, M., Lee, J., Davis, L.: Fused dnn: A deep neural network fusion approach to fast and robust pedestrian detection. In: Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on. pp. 953–961. IEEE (2017)
15. Farfade, S.S., Saberian, M.J., Li, L.: Multi-view face detection using deep convolutional neural networks. In: ICMR. pp. 643–650 (2015). https://doi.org/10.1145/2671188.2749408, http://doi.acm.org/10.1145/2671188.2749408
16. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE transactions on pattern analysis and machine intelligence **32**(9), 1627–1645 (2010)
17. Friedman, J., Hastie, T., Tibshirani, R., et al.: Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). The annals of statistics **28**(2), 337–407 (2000)

18. Geman, S., Graffigne, C.: Markov random field image models and their applications to computer vision. In: Proceedings of the international congress of mathematicians. vol. 1, p. 2 (1986)
19. Girshick, R.B.: Fast R-CNN. In: ICCV. pp. 1440–1448 (2015). https://doi.org/10.1109/ICCV.2015.169, http://dx.doi.org/10.1109/ICCV.2015.169
20. Girshick, R.B., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR. pp. 580–587 (2014). https://doi.org/10.1109/CVPR.2014.81, http://dx.doi.org/10.1109/CVPR.2014.81
21. He, K., Zhang, X., Ren, S., Sun, J.: Spatial pyramid pooling in deep convolutional networks for visual recognition. In: ECCV. pp. 346–361 (2014)
22. Hosang, J., Omran, M., Benenson, R., Schiele, B.: Taking a deeper look at pedestrians. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4073–4082 (2015)
23. Hu, P., Ramanan, D.: Finding tiny faces. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1522–1530. IEEE (2017)
24. Jain, V., Learned-Miller, E.: FDDB: A benchmark for face detection in unconstrained settings. Tech. Rep. UM-CS-2010-009, University of Massachusetts, Amherst (2010)
25. Jiang, H., Learned-Miller, E.: Face detection with the faster r-cnn. In: Automatic Face & Gesture Recognition (FG 2017), 2017 12th IEEE International Conference on. pp. 650–657. IEEE (2017)
26. Jin, S., Su, H., Stauffer, C., Learned-Miller, E.: End-to-end face detection and cast grouping in movies using erdos-renyi clustering. In: ICCV (2017)
27. Kalal, Z., Matas, J., Mikolajczyk, K.: Pn learning: Bootstrapping binary classifiers by structural constraints. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. pp. 49–56. IEEE (2010)
28. Kläser, A., Marszałek, M., Schmid, C., Zisserman, A.: Human focused action localization in video. In: European Conference on Computer Vision. pp. 219–233. Springer (2010)
29. Li, H., Lin, Z., Shen, X., Brandt, J., Hua, G.: A convolutional neural network cascade for face detection. In: CVPR. pp. 5325–5334 (2015). https://doi.org/10.1109/CVPR.2015.7299170, http://dx.doi.org/10.1109/CVPR.2015.7299170
30. Li, J., Liang, X., Shen, S., Xu, T., Feng, J., Yan, S.: Scale-aware fast r-cnn for pedestrian detection. IEEE Transactions on Multimedia (2017)
31. Li, Y., Sun, B., Wu, T., Wang, Y., Gao, W.: Face detection with end-to-end integration of a convnet and a 3d model. ECCV **abs/1606.00850** (2016), http://dblp.uni-trier.de/db/journals/corr/corr1606.html#LiSWW016
32. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: CVPR. vol. 1, p. 4 (2017)
33. Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P.: Focal loss for dense object detection. arXiv preprint arXiv:1708.02002 (2017)
34. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: European conference on computer vision. pp. 21–37. Springer (2016)
35. Loshchilov, I., Hutter, F.: Online batch selection for faster training of neural networks. arXiv preprint arXiv:1511.06343 (2015)

36. Lu, J., Sibai, H., Fabry, E., Forsyth, D.: No need to worry about adversarial examples in object detection in autonomous vehicles. arXiv preprint arXiv:1707.03501 (2017)
37. Luo, Y., Boix, X., Roig, G., Poggio, T., Zhao, Q.: Foveation-based mechanisms alleviate adversarial examples. arXiv preprint arXiv:1511.06292 (2015)
38. Ozerov, A., Vigouroux, J.R., Chevallier, L., Pérez, P.: On evaluating face tracks in movies. In: Image Processing (ICIP), 2013 20th IEEE International Conference on. pp. 3003–3007. IEEE (2013)
39. Ranjan, R., Patel, V.M., Chellappa, R.: A deep pyramid deformable part model for face detection. In: BTAS. pp. 1–8. IEEE (2015), http://dblp.uni-trier.de/db/conf/btas/btas2015.html#RanjanPC15
40. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)
41. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Trans. Pattern Anal. Mach. Intell. (2016)
42. Ren, S., He, K., Girshick, R.B., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: NIPS. pp. 91–99 (2015), http://papers.nips.cc/paper/5638-faster-r-cnn-towards-real-time-object-detection-with-region-proposal-networks
43. Rosenberg, C., Hebert, M., Schneiderman, H.: Semi-supervised self-training of object detection models (2005)
44. Rowley, H.A., Baluja, S., Kanade, T.: Neural network-based face detection. IEEE Transactions on pattern analysis and machine intelligence **20**(1), 23–38 (1998)
45. Schölkopf, B., Smola, A.J.: Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT press (2002)
46. Shrivastava, A., Gupta, A., Girshick, R.: Training region-based object detectors with online hard example mining. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 761–769 (2016)
47. Simo-Serra, E., Trulls, E., Ferraz, L., Kokkinos, I., Moreno-Noguer, F.: Fracking deep convolutional image descriptors. CoRR, abs/1412.6537 **2** (2014)
48. Singh, K.K., Xiao, F., Lee, Y.J.: Track and transfer: Watching videos to simulate strong human supervision for weakly-supervised object detection. In: CVPR. vol. 1, p. 2 (2016)
49. Sonntag, D., Barz, M., Zacharias, J., Stauden, S., Rahmani, V., Fóthi, Á., Lőrincz, A.: Fine-tuning deep cnn models on specific ms coco categories. arXiv preprint arXiv:1709.01476 (2017)
50. Stalder, S., Grabner, H., Van Gool, L.: Cascaded confidence filtering for improved tracking-by-detection. In: European Conference on Computer Vision. pp. 369–382. Springer (2010)
51. Sun, X., Wu, P., Hoi, S.C.: Face detection using deep learning: An improved faster rcnn approach. arXiv preprint arXiv:1701.08289 (2017)
52. Sung, K.K., Poggio, T.: Learning and example selection for object and pattern detection (1994)
53. Sutton, C., McCallum, A.: An introduction to conditional random fields for relational learning, vol. 2. Introduction to statistical relational learning. MIT Press (2006)
54. Tang, K., Ramanathan, V., Fei-Fei, L., Koller, D.: Shifting weights: Adapting object detectors from image to video. In: Advances in Neural Information Processing Systems. pp. 638–646 (2012)

55. Wan, S., Chen, Z., Zhang, T., Zhang, B., Wong, K.k.: Bootstrapping face detection with hard negative examples. arXiv preprint arXiv:1608.02236 (2016)
56. Wang, X., Shrivastava, A., Gupta, A.: A-fast-rcnn: Hard positive generation via adversary for object detection (2017)
57. Wang, Y., Ji, X., Zhou, Z., Wang, H., Li, Z.: Detecting faces using region-based fully convolutional networks. arXiv preprint arXiv:1709.05256 (2017)
58. WESTON, J.: Large-scale semi-supervised learning
59. Yang, B., Nevatia, R.: An online learned crf model for multi-target tracking. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. pp. 2034–2041. IEEE (2012)
60. Yang, S., Luo, P., Loy, C.C., Tang, X.: From facial parts responses to face detection: A deep learning approach. In: ICCV. pp. 3676–3684 (2015). https://doi.org/10.1109/ICCV.2015.419, http://dx.doi.org/10.1109/ICCV.2015.419
61. Yang, S., Luo, P., Loy, C.C., Tang, X.: WIDER FACE: A face detection benchmark. In: CVPR (2016)
62. Yu, J., Jiang, Y., Wang, Z., Cao, Z., Huang, T.: Unitbox: An advanced object detection network. In: Proceedings of the 2016 ACM on Multimedia Conference. pp. 516–520. ACM (2016)
63. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters **23**(10), 1499–1503 (2016)
64. Zhang, L., Lin, L., Liang, X., He, K.: Is faster r-cnn doing well for pedestrian detection? In: European Conference on Computer Vision. pp. 443–457. Springer (2016)
65. Zhang, S., Benenson, R., Omran, M., Hosang, J., Schiele, B.: How far are we from solving pedestrian detection? In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1259–1267 (2016)
66. Zhang, S., Zhu, X., Lei, Z., Shi, H., Wang, X., Li, S.Z.: S3fd: Single shot scale-invariant face detector. arXiv preprint arXiv:1708.05237 (2017)
67. Zitnick, C.L., Dollár, P.: Edge boxes: Locating object proposals from edges. In: European Conference on Computer Vision. pp. 391–405. Springer (2014)