# Non Maximal Suppression in Cascaded Ranking Models

Matthew Blaschko, Juho Kannala, Esa Rahtu

## To cite this version:

Matthew Blaschko, Juho Kannala, Esa Rahtu. Non Maximal Suppression in Cascaded Ranking Models. Scandanavian Conference on Image Analysis, Jun 2013, Espoo, Finland. pp.408-419, 10.1007/978-3-642-38886-6_39 . hal-00815374

# Non Maximal Suppression in Cascaded Ranking Models

Matthew B. Blaschko[1,2,3], Juho Kannala[4], and Esa Rahtu[4]

[1] Center for Visual Computing, École Centrale Paris, France
[2] Équipe Galen, INRIA Saclay, Île-de-France, France
[3] Université Paris-Est, LIGM (UMR CNRS), École des Ponts ParisTech, France
[4] Machine Vision Group, University of Oulu, Finland

**Abstract.** Ranking models have recently been proposed for cascaded object detection, and have been shown to improve over regression or binary classification in this setting [1, 2]. Rather than train a classifier in a binary setting and interpret the function post hoc as a ranking objective, these approaches directly optimize regularized risk objectives that seek to score highest the windows that most closely match the ground truth. In this work, we evaluate the effect of non-maximal suppression (NMS) on the cascade architecture, showing that this step is essential for high performance. Furthermore, we demonstrate that non-maximal suppression has a significant effect on the tradeoff between recall different points on the overlap-recall curve. We further develop additional objectness features at low computational cost that improve performance on the category independent object detection task introduced by Alexe et al. [3]. We show empirically on the PASCAL VOC dataset that a simple and efficient NMS strategy yields better results in a typical cascaded detection architecture than the previous state of the art [4, 1]. This demonstrates that NMS, an often ignored stage in the detection pipeline, can be a dominating factor in the performance of detection systems.

## 1 Introduction

State of the art object detection typically consists of training a statistical classifier and evaluating that classifier at as many regions of an image as is computationally feasible. While such an approach is conceptually straightforward, it is often inefficient as the majority of windows in an image do not contain an instance of an object. Recognizing this imbalance, Viola and Jones proposed to improve accuracy on a fixed computational budget by designing a cascade architecture in which relatively few, inexpensive decision functions were sufficient to reject a high proportion of image locations [5]. In this work, we consider the problem of *learning* a cascade architecture rather than ordering function computations into a cascade after training is completed. In particular, we employ a ranking based objective that learns to assign high score to promising image regions using inexpensive features [1, 2]. Regions that are scored highest are then re-ranked using a more expensive function class, saving higher computational effort for regions that are most likely to contain an instance of an object.

Several broad trends have been pursued in the literature to improve the computational efficiency of object detection while maintaining the accuracy of exhaustive sliding window approaches. One such method is to train a binary classifier on a set of samples using boosting or support vector machines, and then reorder the component evaluations into a cascade architecture [6, 5]. Vedaldi et al. have recently applied a similar setting that learns a series of progressively more complex classifiers based on a hierarchy of kernel complexity [7]. Another approach is to incorporate the cascade architecture into the learning process by making assumptions about the error distribution at each cascade layer [8]. Lampert et al. proposed to use branch and bound to find the optimal window according to a scoring function [9], while Lehmann et al. extended this approach by approximating bounds by learned ranking functions [10]. Discriminative max-margin ranking was introduced by Herbrich et al. and enforces margin constraints for pairs of training samples [11]. This approach was recently extended to additionally incorporate structured output losses [12, 13, 1, 2, 14], and it is this approach that the present work extends. We evaluate our results within the category independent object detection framework proposed by [3].

All detection systems must necessarily incorporate a non-maximal suppression (NMS) step into the detection process. This is because overlapping windows will have similar scores, and the set of detections may be clustered around a single image region, ignoring other detections. Viola and Jones used the strategy of performing $k$-means to replace clusters of detections with their mean bounding box [5]. Many recent systems perform greedy optimization based on overlap or a probabilistic detection measure [15, 16, 1, 2]. NMS is in general NP-hard to solve optimally [15, 16], posing challenges to the tractable formulation of NMS aware learning strategies.

We propose in this work to address the effect of NMS on the learning problem empirically, carefully evaluating the impact of design choices on cascade performance. Of key interest is the relative performance of various design strategies for improving results. To this end, we have developed a number of improved features for the category independent detection task. These features enable us to increase the capacity of the discriminant function at low computational cost. We are therefore able to evaluate the relative improvements resulting from improved features, and non-maximal suppression strategies. The rest of the paper is structured as follows. In Section 2 we introduce precisely the family of ranking objectives considered for the cascade setting. We develop several properties of non-maximal suppression in Section 3, and discuss how the design of this step impacts the detection pipeline. Our enhanced feature pipeline is presented in Section 4. An empirical evaluation is performed in Section 5, and discussed in Section 6. We next describe the family of ranking algorithms.

## 2 Ranking Cascades

A ranking cascade is a model that uses a discriminative ranking function at each layer of the cascade [1, 2]. The discriminative ranking function is trained using an objective that enforces constraints for pairs of detections, such that the better detection is ranked higher than the worse detection. Mistakes are penalized proportional to the difference in loss between the misordered pair. The learning objective is as follows

$$\min_{w,\xi} \ \frac{1}{2}\|w\|^2 + C \sum_{(i,j)\in\mathcal{E}} \xi_{ij} \tag{1}$$

$$\text{s.t. } \langle w, \phi(x_i, y_i)\rangle - \langle w, \phi(x_j, y_j)\rangle \geq 1 - \frac{\xi_{ij}}{\Delta_j - \Delta_i}, \quad \xi_{ij} \geq 0 \quad \forall (i,j) \in \mathcal{E} \tag{2}$$

where $w$ parametrizes the scoring function, $\langle w, \phi(x_i, y_i)\rangle$, that orders image–detection pairs according to the fitness of the detection, $y_i$, in its corresponding image, $x_i$. $\Delta_i$ encodes the loss associated with making a (partially overlapping) prediction $y_i$, while $\phi$ is a feature function that encodes statistics of the image, $x_i$, at the detection location, $y_i$. $\mathcal{E}$ is the edge set of a preference graph that specifies which pairs of samples will have a constraint enforced in Equation (2). This model is the *slack rescaled* variant of structured output ranking, which has favorable properties compared to a related variant, margin rescaling [17]. $\Delta_j - \Delta_i$ is the difference in losses between a worse prediction, $j$, and a better one, $i$, while (2) pays a convex bound to this loss when a margin between such pairs is violated.

A key issue is the complexity of optimizing (1). This complexity depends on the form of $\mathcal{E}$. In the case of bipartite $\mathcal{E}$, [1] give a linear time algorithm for optimizing a related ranking objective. This makes feasible the optimization of preference graphs that contain millions of training samples. The bipartite preference graphs are designed to give performance tailored to the specific task of cascaded detection, while maintaining a high degree of scalability.

Cascaded detection works by applying an inexpensive function, in our case $w$, to a set of candidate detections, filtering these at each step. By successively increasing the complexity of the decision function, later stages can apply a more sophisticated decision function to a small fraction of potential candidate windows. Early stages are thus only tasked with making relatively easy distinctions using a less expensive function class. It is essential to apply a non-maximal suppression (NMS) strategy in order to achieve good results. NMS enforces that a diverse set of candidate windows pass through to subsequent stages by removing detections that significantly overlap with other detections. It is provably NP-hard to apply optimal non-maximal suppression in general, posing a significant challenge to learning strategies that make use of this step [15, 16]. We discuss in the next section how the careful design of non-maximal suppression strategies can approximate optimal performance while maintaining the overall tractability of the learning approach.

## 3 Non-Maximal Suppression

Non-maximal suppression ensures that the set of candidates generated at a layer of a cascade be diverse. A number of schemes have been proposed, but in general the task can be seen to trade off a unary potential measuring the quality of a given detection, with pairwise or higher order potentials that measure the degree of overlap of pairs or sets of detections. This can be interpreted as a random field model [16] resulting in a non-submodular minimization:

$$\min_{\Pi} -\sum_{i} \langle w, \phi(x_{\Pi_i}, y_{\Pi_i}) \rangle + \sum_{ij} \Omega(y_{\Pi_i}, y_{\Pi_j}) \tag{3}$$

where $\Pi$ is the set of detections selected for further processing, and $\Omega$ is a function that penalizes overlapping detections, ensuring a diverse set. For such a strategy to make sense, the unary potentials, $\langle w, \phi(x_{\Pi_i}, y_{\Pi_i}) \rangle$, must be ordered appropriately by the detection quality, i.e. the overlap with the ground truth.

Ideally, the non-maximal suppression step would be incorporated directly into the learning procedure, so that the top samples *after* non-maximal suppression be constrained to be ranked above the rest of the samples, and that the unary potentials be optimized to improve the resulting performance. However, such a strategy makes the learning architecture intractable due to the NP-hardness of the NMS step [15, 16]. Instead, a typical approach is to compose the learning step with an NMS step that makes use of the learned unary potentials, enabling NMS to be computed using an efficient approximation [15, 16, 18] to Equation (3). We explore in the Experimental Results section the composition of ranking and NMS by subsequently reranking detections after the NMS procedure, using a second ranking objective trained specifically to discriminate detections with high overlap to the ground truth.

Rahtu et al. made use of a two stage NMS strategy in a cascaded detection setting [1]. In this strategy, they did an initial approximate filter to decrease the number of windows. This was achieved by looking for local maxima in the window score landscape to select a pool of windows approximately one tenth the number of the original set, and ten times the number to be selected. Subsequently, the NMS strategy of [7] was employed on this reduced set of windows, which can be interpreted as an application of the approximation scheme of [18]. The two stage NMS strategy of [1] can be thought of as approximately optimizing higher order potentials in Equation (3) that favor windows that are local maxima. We evaluate this assumption empirically in Section 5.

Equally important are the pairwise potentials indicating the penalization allocated to overlapping windows. We consider a simple case in which windows are simply discarded if the overlap to a previous detection is too high. This can be viewed as a pairwise potential that is zero if the overlap threshold is not reached, and $\infty$ otherwise. If the threshold is set to be high, fewer windows will be discarded by non-maximal suppression, and the set of windows will be less diverse. They will, however, be more tightly localized around the areas with highest score. This will maximize recall at very high overlap, while recall at lower overlap will decrease. When the threshold is lowered, NMS is more strict, discarding a higher proportion of windows. The average score of the windows passing through NMS filtering will be lower, but the diversity of such windows will be higher. We thus expect to have fewer very precise detections, but will have higher recall at lower overlap.

The form of pairwise suppression potentials is key for trading off performance along the recall-overlap curve. While the PASCAL VOC challenge evaluates detection accuracy using a threshold of 0.5 overlap [19], it is often key to have very precisely localized candidate windows for models that are sensitive to spatial misalignment. In such a case, overall performance is likely to be optimized by tuning earlier cascade layers to trade off lower recall at 0.5 overlap for higher recall at higher overlap, with the exact parameter settings depending on the behavior of subsequent layers.

## 4 Image Features

Stronger function classes and richer features are key to improve performance in object detection [7]. Furthermore, insufficiently rich function classes may make it impossible to observe significant differences between competing learning algorithms. As early cascade layers are strongly constrained computationally, it is almost certain that the features employed will tend to underfit rather than overfit the data. Furthermore, cascade architectures are incapable of recovering from early mistakes, further reinforcing the importance of good feature design for these early stages. To this end, we develop here several novel features for category independent object detection. The feature set, $\phi(x_i, y_i)$, employed here extends the publicly available features described in [3] and [1], and shares much of the same computation. The cost of the entire pipeline is comparable to these previous works.

We have largely followed the strategy of [1] in generating the initial set of candidate windows. This consists of generating windows in two ways. The first is to use superpixel segmentations[5] to generate sets of windows that form bounding boxes around superpixel combinations, while the second is to sample windows according to a learned spatial distribution of object locations. Using these two strategies, approximately $10^5$ windows are generated. Noting that bounding boxes of superpixel combinations are more likely to describe the location of an object, we enable the learning algorithm to base its estimate of a bounding box fitness on the information of which of these two processes generated the candidate. Specifically, we include binary features that specify in which way a box was generated.

We propose a new set of computationally efficient image features which allow to characterize the generic objectness of candidate windows. The features are based on measuring the responses of a set of linear filters along the superpixel boundaries. This approach is motivated by the observations in [3] and [1], where it was noted that both superpixel boundaries and gradient magnitudes are useful objectness cues. Hence, in a sense, our aim is to combine information from both superpixels and gradients, and to do it in a way which is statistically justified so that more weight is given to rare (i.e. salient) features in each particular image.

[5] We have extended this candidate generation strategy by using four different superpixel segmentations using different color spaces: RGB, opponent color space, normalized RGB, and the hue channel from HSV.

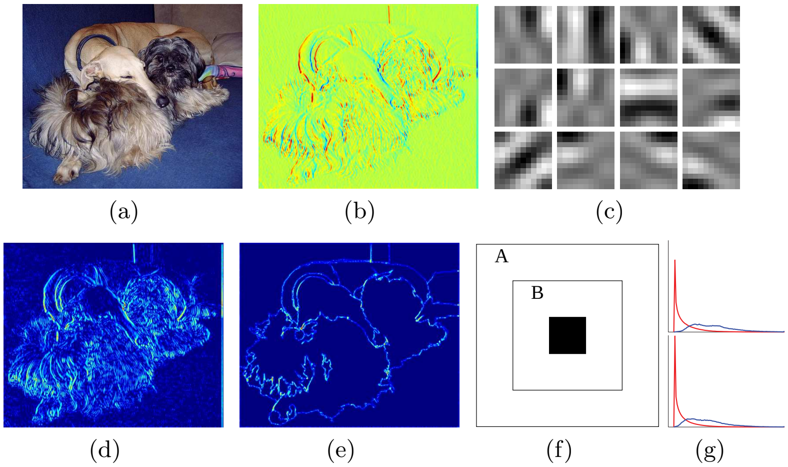


**Fig. 1.** Our image features measure the responses of 12 linear filters along the superpixel boundaries. (a) An example image. (b) Example image filtered with the first filter. (c) The set of 12 filters of size 11×11 pixels. (d) The filter response image (c) transformed to a self-information map by estimating the histogram of responses. (e) The self-information map weighted with the superpixel boundary map. (f) Given a candidate bounding box we integrate the values of (e) over two subregions (A and B) of the box, and repeat this for all the filters, which results in 24 values for the box. (g) Distribution of the summed feature values integrated over the outer (top) and inner (bottom) boundary regions (regions A and B in (f)) for boxes whose overlap score with a ground truth box is $\geq 0.5$ (blue) and $< 0.5$ (red).

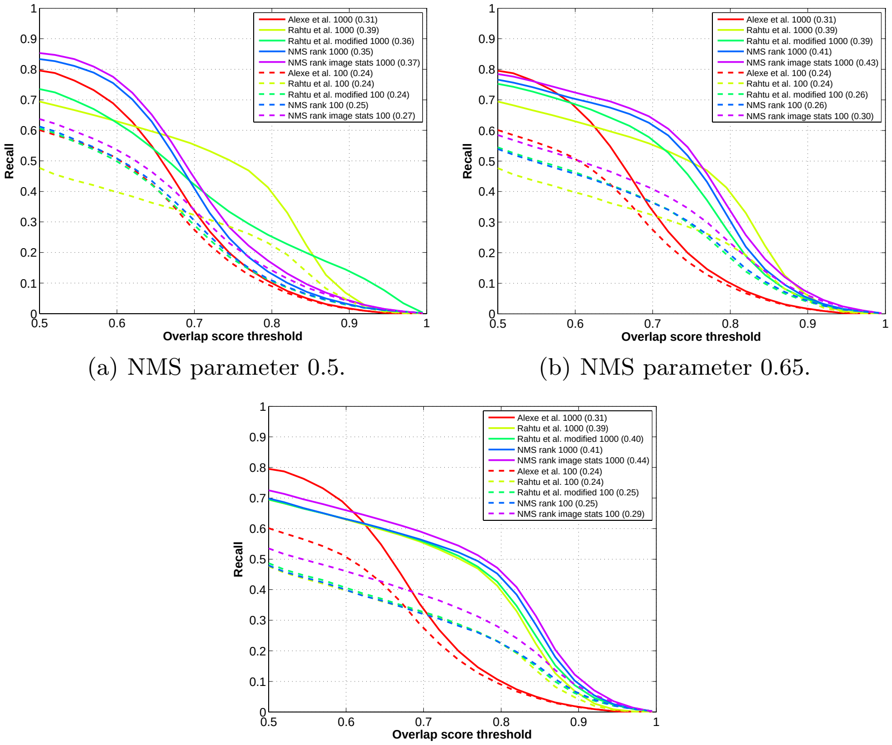
However, instead of simply using gradient filters in four different orientations as in [1], we use a richer set of features by learning a set of linear filters whose responses are as independent as possible in a set of natural training images. For this we use the independent component analysis method described in [20]. That is, given a set of image patches from natural images, represented as elements of a vector space, a linear basis is learnt for the vector space so that when the patches are projected to the basis their coordinates are as statistically independent as possible. The corresponding projection operators are linear filters. In this work, we use image patches of 11×11 pixels from a set of 13 training images provided by [20], and estimate 12 filters of size 11×11 which are shown in Fig. 1(c). The filters represent edge and corner detectors and such features typically emerge from natural images independently of the particular dataset [20]. Hence, these filters can be considered as generally informative features for natural images, and they can be fixed after once learned.

An example of a filter response image, obtained with one of the filters, is shown in Fig. 1(b). This filter response map can be seen to correspond to the oriented gradient magnitude maps which were used in [1] to build their objectness measures. In fact, some of the filters are quite similar to horizontal and vertical gradient filters. However, as we use more filters we get a richer set of features. Moreover, we do not directly integrate the filter responses along the boundary of each candidate window to measure its fitness, as in [1]. Instead, we first transform the response map to a self-information map (Fig. 1(d)), which reflects the rarity of features at the image level. That is, we estimate the histogram of filter responses in the image, normalize it to a discrete probability density function, $p_k$, and then assign the value $-log(p_k)$ to pixels assigned to histogram bin $k$. These self-information values reflect the informativeness of local image patches. Typically strongest responses are the rarest ($p_k$ is small) and their self-information $-log(p_k)$ is large. Nevertheless, if there is background clutter or dominant background texture in the image, it may be that not all strong edges are as informative as others. Hence, certain edge and corner patches may be more informative object boundary indicators than others, and this informativeness (or saliency) may vary in different images based on their content. Further, the most salient boundaries are intuitively the most promising object boundaries and our features aim to utilize this information.

Finally, since [3, 1] have observed that superpixels are powerful cues of object boundaries, we combine the information from our features with the superpixel boundaries by weighting the self-information maps (like the one in Fig. 1(d)) with a smoothed version of the superpixel boundary image. The result is illustrated in Fig. 1(d). Then, given a candidate object window, we integrate the values from the weighted self-information maps over two different subregions of the window which are illustrated in Fig. 1(f) (regions A and B). Since there are 12 self-information maps per image and two subregions per window we get in total 24 scalar feature values per window. As we need to evaluate the feature values for a large number of windows we utilize integral images for the integration. The overall computational cost is comparable to the superpixel straddling measure introduced by [3].

In principle, due to the independence assumption of our feature responses it is justified to think that the "boundary information" carried by all the self-information maps is obtained by summing the map values related to different filters. This is because the joint density of independent random variables is the product of individual densities and the logarithm of a product is the sum of logarithms. In fact, by summing the integrals of different maps in the two window subregions provides two values per window and in Fig. 1(g) we have plotted the distributions of these values for windows whose maximum overlap score with a ground-truth object box is either $\geq 0.5$ or $< 0.5$. (We used a subset of PASCAL VOC images.) This shows that our features indeed allow to distinguish the object windows from the other windows.

However, as the features related to different filters may have different performance as objectness measures and since our final scoring function for windows is simply a linear combination of feature values, we do not sum the different feature values in advance but let them be separate in a 24 dimensional vector so that the ranking approach of Section 2 may learn suitable relative weights for the different feature components. At test time the new features are linearly combined with other features (those from [3, 1]) using learnt weights.



(a) NMS parameter 0.5.

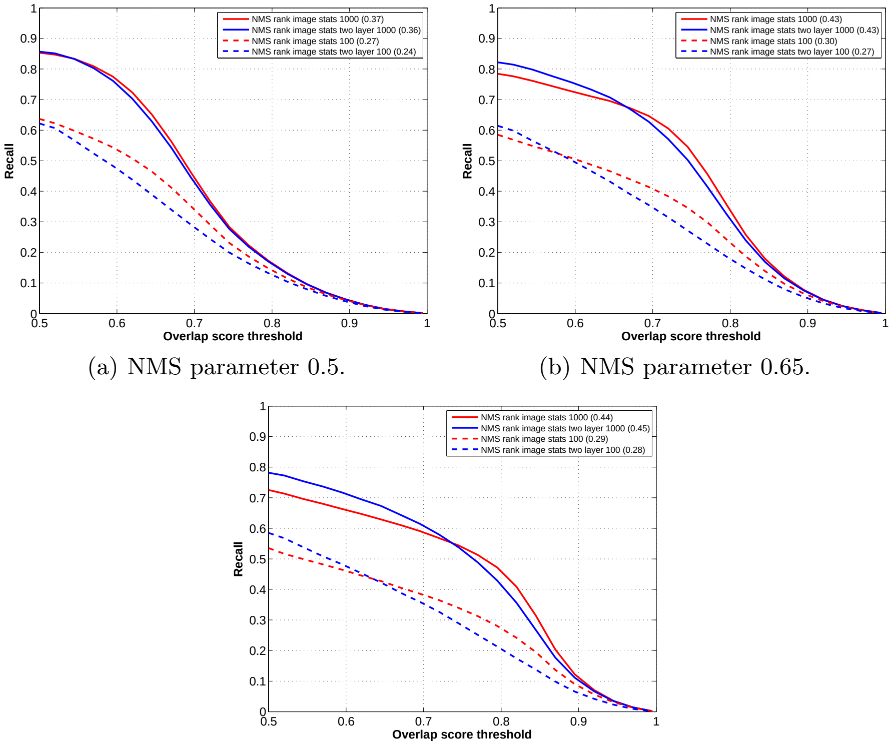
(b) NMS parameter 0.65.

(c) NMS parameter 0.75.

**Fig. 2.** A comparison of overlap-recall results across various systems with different NMS overlap parameters. Alexe et al. corresponds to the results presented in [4]. Rahtu et al. corresponds to the results presented in [1], while "Rahtu et al. modified" corresponds to our re-implementation of that system with varying NMS parameters. The results from [4, 1] are presented with fixed NMS parameters across the different plots, while the other curves have varying NMS thresholds. "NMS rank" corresponds to the system described in this paper with simplified NMS parameters while "NMS rank image stats" corresponds to the system with the new natural image statistics features incorporated.

## 5 Experimental Results

We evaluate our cascade architecture in the generic object detection task [3]. In this task, a cascade layer generates category independent object detection proposals. This has the advantage of the resulting proposals being the same for all categories, enabling the first cascade layer to be shared across multiple category dependent cascades. Provided such a problem can be solved efficiently and accurately, such a shared architecture promises to multiply the computational savings by the number of categories. Also, generic object detection enables the system to pool large numbers of training samples across multiple interrelated object categories, increasing the statistical efficiency, and enabling the system to



(a) NMS parameter 0.5.

(b) NMS parameter 0.65.

(c) NMS parameter 0.75.

**Fig. 3.** Re-ranking variant of NMS. The red curve corresponds to the NMS strategy used in previous figures, while the blue curve corresponds to the re-ranking strategy. Reranking tends to improve recall at low overlap at the expense of recall at high overlap. Measured by area under the curve (AUC), the results in Fig. 3(c) are the highest reported.

generalize to previously unseen object classes [3, 21]. As [1, 7], we present results in terms of recall-overlap curves, which measure the recall achieved in a test set for a given level of detection overlap as measured by the VOC overlap score on the 2007 dataset [19]. We have included difficult and truncated objects in this evaluation. We compare to the previous state of the art [3, 4, 1]. Like previous works, we show results for both 100 and 1000 windows returned per image.

Our first experiments compare recall-overlap curves for several baseline systems and new results with varying NMS parameters and numbers of returned windows. We have presented the results of [4], as these result from an improved version of the system described in [3], and are strictly better. We have reimplemented the system of [1] as it is the base for this work, and enables us to vary the parameters of their NMS system in concert with those of our simplified NMS strategy. We have also reported the results achieved by downloading their precomputed detections for comparison. Results for this setting are in Fig. 2. We see that our reimplementation performs slightly better, possibly due to our using different superpixel parameters or the use of slack rescaling instead of margin rescaling in the learning.

In the second set of experiments, we evaluate a variant of non-maximal suppression in which we first detect 5000 windows. Subsequent to this filtering, we apply a ranking discriminant function that was trained using the top 2000 windows per image with a complete preference graph (c.f. Section 3). We then apply this re-ranking objective to the 5000 filtered windows to obtain the top scored detection candidates. This enables the objective function to focus purely on discriminating the top candidates. Results are shown in Fig. 3.
Pre-computed detections from the experiments described here are available for download from `http://www.cse.oulu.fi/CMV/Downloads/ObjectDetection`.

## 6 Discussion

The experiments in Fig. 2 demonstrate the tradeoff inherent in the NMS threshold parameter. Lower values of this parameter result in the overlap-recall curve performing best at low overlap levels, while higher values shift the curve proportionately to the right at the expense of performance at low overlap. Alexe et al. seem to have optimized their system at 0.5 overlap, corresponding to the recall computed in the VOC challenge. Our system with NMS parameter set to 0.5 dominates the performance of [4] at all points on the curve, while our system with NMS parameter set to 0.75 dominates the performance of [1] at all points on the curve. In comparing different strategies, it is thus important to identify which part of the curve has been optimized. The green curve indicates the performance of the NMS strategy of [1]. We see that it has a bias towards optimizing recall at high overlap at the expense of performance at lower overlap. As measured by area under the curve (shown in parenthesis in the figure legend), the simplified NMS strategy employed here gives better performance.

In the second set of experiments, we use an alternate NMS strategy in which NMS is interleaved between two ranking steps (Fig. 3). This approach is designed to couple the NMS step with the optimization. As discussed in Sect. 3, computational issues prevent joint optimization of the learning objective with NMS incorporated. Instead, most works have first performed learning, and subsequently apply NMS. These experiments show that improvements can be made with a simple strategy that couples learning and NMS, but is still computationally tractable. Measured by AUC, the results in Fig. 3(c) are the highest reported so far [4, 1].

## 7 Conclusions

We have shed light on a portion of the detection architecture that is usually left unanalyzed, and is often performed using ad hoc approaches. We have shown that non-maximal suppression is critical to the performance of a system, and naturally encodes a tradeoff between recall levels at different detection qualities. The most striking observation is that the size of the changes in performance due to NMS is much larger than those due to features or window sampling. Our system has improved on the state of the art in the task of category independent object detection, dominating the performance of previous systems at all levels of overlap with a modest increase in computational cost for additional features.

NMS is typically uncoupled from statistical learning as its complexity complicates the formulation of tractable approximations. We have shown here that a loose coupling resulting from an additional ranking step can improve results while maintaining computational tractability. A combination of improved NMS and an augmented feature pipeline have enabled us to improve over the previous state of the art. We will make our code, pre-computed features, and detections available at the time of publication. We see several opportunities for the application of our work in related settings. The ranking models and NMS models developed here are likely applicable to the setting of [10]. It would also be possible to replace the use of SVMs in [7] with a ranking cascade model for category dependent object detection.

### Acknowledgements

This work is partially funded by the European Research Council under FP7/ERC Grant 259112. This research is based in part on work done at the 2012 CLSP Summer Workshop, and was partially supported by NSF Grant 1005411, ODNI via the JHU HLTCOE and Google Research. This work is also supported by the Academy of Finland Grant 127702.

## References

1. Rahtu, E., Kannala, J., Blaschko, M.B.: Learning a category independent object detection cascade. In: Proc. ICCV. (2011)
2. Zhang, Z., Warrell, J., Torr, P.H.S.: Proposal generation for object detection using cascaded ranking SVMs. In: Proc. CVPR. (2011)

3. Alexe, B., Deselaers, T., Ferrari, V.: What is an object. In: Proc. CVPR. (2010)
4. Alexe, B., Deselaers, T., Ferrari, V.: Measuring the objectness of image windows. IEEE PAMI (2012)
5. Viola, P., Jones, M.: Robust real-time object detection. In: IJCV. Volume 1. (2001)
6. Romdhani, S., Torr, P., Schölkopf, B., Blake, A.: Computationally efficient face detection. In: Proc. ICCV. (2001)
7. Vedaldi, A., Gulshan, V., Varma, M., Zisserman, A.: Multiple kernels for object detection. In: Proc. ICCV. (2009)
8. Wu, J., Brubaker, S.C., Mullin, M.D., Rehg, J.M.: Fast asymmetric learning for cascade face detection. IEEE PAMI **30** (March 2008) 369–382
9. Lampert, C.H., Blaschko, M.B., Hofmann, T.: Efficient subwindow search: A branch and bound framework for object localization. IEEE PAMI (2009)
10. Lehmann, A., Gehler, P., Van Gool, L.: Branch & rank: Non-linear object detection. In: Proc. BMVC. (2011)
11. Herbrich, R., Graepel, T., Obermayer, K.: Large margin rank boundaries for ordinal regression. In Smola, A.J., Bartlett, P.L., Schölkopf, B., Schuurmans, D., eds.: Advances in Large Margin Classifiers, MIT Press (2000) 115–132
12. Bakir, G.H., Hofmann, T., Schölkopf, B., Smola, A.J., Taskar, B., Vishwanathan, S.V.N.: Predicting structured data. MIT press (2007)
13. Blaschko, M.B., Vedaldi, A., Zisserman, A.: Simultaneous object detection and ranking with weak supervision. In: NIPS. (2010)
14. Mittal, A., Blaschko, M.B., Zisserman, A., Torr, P.H.S.: Taxonomic multi-class prediction and person layout using efficient structured ranking. In: Proc. ECCV. (2012)
15. Barinova, O., Lempitsky, V., Kohli, P.: On the detection of multiple object instances using Hough transforms. In: Proc. CVPR. (2010)
16. Blaschko, M.B.: Branch and bound strategies for non-maximal suppression in object detection. In: Proc. EMMCVPR. (2011)
17. McAllester, D.: Generalization bounds and consistency for structured labeling. In Bakır, G.H., Hofmann, T., Schölkopf, B., Smola, A.J., Taskar, B., Vishwanathan, S.V.N., eds.: Predicting Structured Data, MIT Press (2007) 247–261
18. Nemhauser, G.L., Wolsey, L.A., Fisher, M.L.: An analysis of approximations for maximizing submodular set functions. Mathematical Programming **14** (1978) 265–294
19. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes (VOC) challenge. IJCV **88**(2) (Jun 2010) 303–338
20. Hyvärinen, A., Hurri, J., Hoyer, P.: Natural Image Statistics. Springer (2009)
21. Deselaers, T., Alexe, B., Ferrari, V.: Localizing objects while learning their appearance. In: Proc. ECCV. (2010)