How hard can it be? Estimating the difficulty of visual search in an image

Radu Tudor Ionescu[1], Bogdan Alexe[1,4], Marius Leordeanu[3], Marius Popescu[1],
Dim P. Papadopoulos[2], Vittorio Ferrari[2]

[1]University of Bucharest, [2]University of Edinburgh,
[3]Institute of Mathematics of the Romanian Academy,
[4]Institute of Mathematical Statistics and Applied Mathematics of the Romanian Academy

Abstract

We address the problem of estimating image difficulty defined as the human response time for solving a visual search task. We collect human annotations of image difficulty for the PASCAL VOC 2012 data set through a crowd-sourcing platform. We then analyze what human interpretable image properties can have an impact on visual search difficulty, and how accurate are those properties for predicting difficulty. Next, we build a regression model based on deep features learned with state of the art convolutional neural networks and show better results for predicting the ground-truth visual search difficulty scores produced by human annotators. Our model is able to correctly rank about 75% image pairs according to their difficulty score. We also show that our difficulty predictor generalizes well to new classes not seen during training. Finally, we demonstrate that our predicted difficulty scores are useful for weakly supervised object localization (8% improvement) and semi-supervised object classification (1% improvement).

1. Introduction

Humans can naturally understand the content of images quite easily. The visual human perception system works by first recognizing the 'gist' of the image almost instantaneously [32, 33], just from a single glance (200 ms) and, then, in a second stage, by recognizing the individual objects in the image [33] as a result of visual search. Cognitive studies [3, 43, 48] show evidence that, for the task of searching for a pattern in an image, the user response time is proportional to the visual search difficulty, which could vary from one image to another. Images are not equal in their difficulty: some images are easy to search and objects are found fast while others are harder, requiring intensive visual processing by humans. The measure of visual search difficulty could be related to several factors such as background clutter, complexity of the scene, number of objects, whether they are partially occluded or not, and so on.

In this paper, we address the problem of estimating visual search difficulty. This topic is little explored in the computer vision literature with no data sets assessing the difficulty of an image being available. We approach our study by collecting annotations on the PASCAL VOC 2012 data set [15] as human response times during a visual search task and convert them into difficulty scores (Section 2). While measuring visual search difficulty by human observations might be subject to some user variability, we believe that there are intrinsic image properties that constitute the ingredients in the unknown underlying recipe of making an image difficult (Figure 1). We use the PASCAL VOC 2012 images annotated with difficulty scores to investigate in depth how different image properties correlate with the ground-truth difficulty scores. We find that higher level features, such as the ones learned with convolutional neural networks (CNN) [25] are the most effective, suggesting that visual search difficulty is indeed a measure that relates to higher level cognitive processing. Using such features, we train models to automatically predict the human assessment of visual search difficulty in an image (Section 3). We release the human difficulty scores we collected on PASCAL VOC 2012, as well as our code to predict the difficulty of any image at http://image-difficulty.herokuapp.com.

Measuring image difficulty could have many potential applications that use the primary information that some images are harder to analyze than others. In Section 4, we demonstrate the usefulness of our difficulty measure in two object recognition applications. For the task of weakly supervised object localization, we show how to enhance standard methods based on multiple instance learning [5, 8, 10, 37, 39, 40, 41] with our measure and obtain an 8% improvement. Similarly, for the task of semi-supervised object classification, we use our measure to improve the accuracy of a classifier based on CNN features [38] by 1%.

Related work. There are many computer vision works analyzing global image properties such as saliency [17, 19, 26, 30, 31], memorability [20, 21], photo quality [29] and

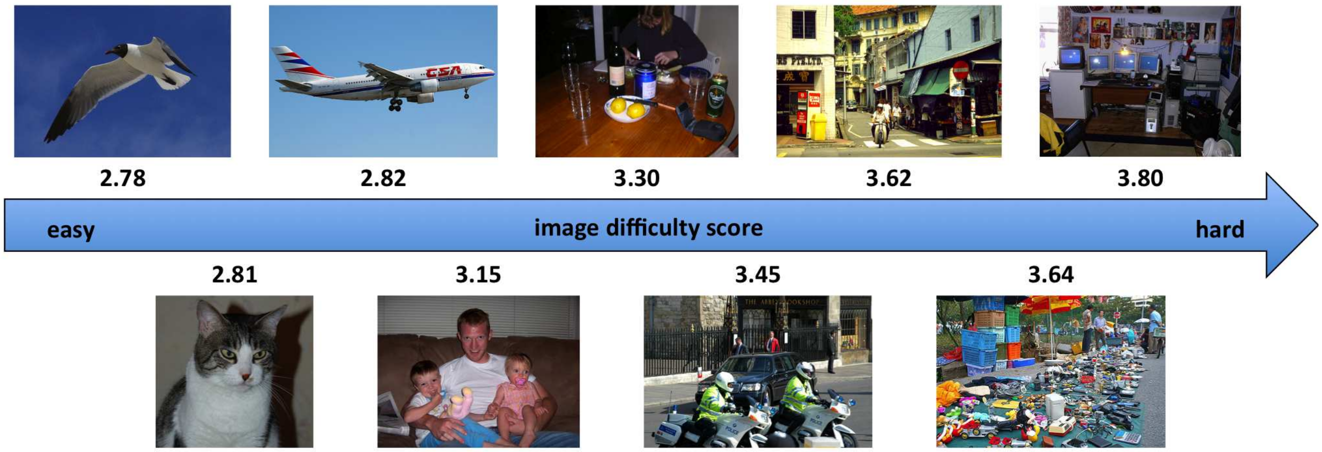

Figure 1. Images with difficulty scores predicted by our system in increasing order of their difficulty.

objects' importance [42]. However, there is little work on the topic of image difficulty [28, 34, 46]. Russakovsky et al. [34] measure difficulty as the rank of an object's bounding-box in the order of image windows induced by the objectness measure [1, 2]. This basically measures image clutter. However, it needs ground-truth bounding-boxes in order to quantify difficulty (even at test time). Liu et al. [28] predict the performance of a segmentation algorithm to be applied to an image, based on various features including gray tone, color, gradient and texture (on just 100 images). More closely related to our idea, Vijayanarasimhan and Grauman [46] try to predict the difficulty of an image in terms of the time needed by a human to segment it, with the specific goal of reducing manual annotation effort. They select candidate low-level features and train multiple kernel learning models to predict easy versus hard images. However, the image segmentation task [46] is conceptually different from our visual search task. For example, it might be very easy to find a tree in a particular image, although it can be very hard to segment, while a truncated car can be easily segmented but difficult to find and recognize. Jain and Grauman [22] predict what level of human annotation will be sufficient for interactive segmentation to succeed. Their approach learns the image properties that indicate how successful a given form of user input will be.

In contrast to these previous works, we approach the problem from a higher level of image interpretation, for the general task of visual search and collect annotations for a much larger data set of over 10K images.

2. Image difficulty from a human perspective

Supported by cognitive studies [3, 43, 48], we consider that the difficulty of an image is related to how hard it is for a human to decide the presence or absence of a given object class in an image. We quantify the difficulty as the time needed by a human to solve this visual search task. This value could depend on several factors such as the amount of irrelevant clutter in the image, the number of objects, their scale and position, their class type, the relevant contextual relationships among them, occlusions and other kinds of noise. We thoroughly investigate how these properties correlate with the visual search difficulty in Section 2.2.

First, we designed a visual search protocol for collecting human response times on a crowd-sourcing platform, namely CrowdFlower[1]. We collected ground-truth difficulty annotations by human evaluators on a per image basis for all $11,540$ *train* and *validation* images in PASCAL VOC 2012 data set [15]. This data set contains images with object instances from 20 classes (*aeroplane*, *boat*, *cat*, *dog*, *person* and so on) annotated with bounding-boxes. The images vary in their difficulty: objects appear against a variety of backgrounds, ranging from uniform to heavily cluttered, and vary greatly in their number, location, size, appearance, viewpoint and illumination. This variety makes this data set very suitable for collecting ground-truth difficulty annotations. We next describe the protocol and present informative statistics about the collected data.

2.1. Can we measure visual search difficulty?

Collecting response times. We collected ground-truth difficulty annotations by human evaluators using the following protocol: (i) we ask each annotator a question of the type "Is there an $\{$*object class*$\}$ in the next image?", where $\{$*object class*$\}$ is one of the 20 classes included in the PASCAL VOC 2012; (ii) we show the image to the annotator; (iii) we record the time spent by the annotator to answer the question by "Yes" or "No". Finally, we use this response time to estimate the visual search difficulty.

To make sure the measured time is representative, the annotator has to signal that he or she is ready to see the image by clicking a button (after reading the question first). After

[1] http://www.crowdflower.com/

	Mean	Minimum	Maximum
Kendall τ	0.562 ± 0.127	0.182	0.818

Table 1. Kendall's τ rank correlation coefficient among 58 trusted annotators, on a subset of 56 images. The response time of each annotator is compared to the mean response time of all annotators.

seeing the image and analyzing it, the annotator has to signal when he or she made up his mind on the answer by clicking another button. At this moment we hide the image to prevent cheating on the time. Moreover, we made sure the annotation task is not trivial by associating two questions for each image, such that the ground-truth answer for one question is positive (the object class specified in the question *is present* in the image) and the ground-truth answer for the other question is negative (the object class specified in the question *is not present* in the image). In this way we prevented a bias in obtaining answers uncorrelated with the image content, constraining the annotator to be focused during the entire task. Each answer ("Yes" or "No") has a 50% chance of being the right choice. Naturally, an annotator could memorize an image and answer more quickly if the image would be presented several times, so we made sure that a person did not get to annotate the same image twice. Each question was answered by three human annotators. Given that we used $11,540$ images and we associated two questions per image, we obtained $69,240$ annotations. The annotations come from 736 trusted contributors. A *trusted* contributor has an accuracy (percentage of answers that match the ground-truth answers) higher than 90%.

Data post-processing and cleanup. When the annotation task was finished, we had 6 annotations per image (3 for each of the two questions) with the associated response times. We removed all the response times longer than 20 seconds, and then, we normalized each annotator's response times by subtracting the annotator's mean time and by dividing the resulted times by the standard deviation. We removed all the annotators with less than 3 annotations since their mean time is not representative. We also excluded all the annotators with less than 10 annotations with an average response time higher than 10 seconds. After removing all the outliers, the difficulty score per image is computed as the geometric mean of the remaining times. It is worth mentioning that by adjusting the accuracy threshold for trusted annotators to 90%, we allow some wrong annotations in the collected data. Wrong annotations provide the ultimate evidence of a difficult image, showing also that the problem of estimating image difficulty is not trivial. We determined the images containing wrong annotations (based on the ground-truth labels from PASCAL VOC 2012) and added a penalty to increase the difficulty scores of these images.

Human agreement. We report the inter-human correlations on a subset of 56 images that we used to spot untrusted annotators in CrowdFlower. We consider only the 58 trusted annotators who annotated all these 56 images. In this setting, we compute the correlation following a one-versus-all scheme, comparing the response time of an annotator to the mean response time of all annotators. For this, we use the Kendall's τ rank correlation coefficient [24, 44]. Kendall's τ is a correlation measure for ordinal data based on the difference between the number of concordant pairs and the number of discordant pairs among two variables, divided by the total number of pairs. The mean Kendall's τ correlation is reported in Table 1, along with the standard deviation, the minimum and the maximum correlations obtained. The mean value of 0.562 means that the average human ranks about 80% image pairs in the same order as given by the mean response time of all annotators. This high level of agreement among humans demonstrates that visual search difficulty can indeed be consistently measured.

Image property		Kendall τ
(i)	number of objects	0.32
(ii)	mean area covered by objects	-0.28
(iii)	non-centeredness	0.29
(iv)	number of different classes	0.33
(v)	number of truncated objects	0.22
(vi)	number of occluded objects	0.26
(vii)	number of difficult objects	0.20
(viii)	combine (i) to (vii) with ν-SVR	0.36

Table 2. Kendall's τ rank correlations for various image properties.

2.2. What makes an image difficult?

Images are not equal in their difficulty. In order to gain an understanding of what makes an image more difficult than another, we consider several human interpretable image properties and analyze their correlation with the visual search difficulty assessed by humans. The image properties are derived from the human manual annotations provided for each image in PASCAL VOC 2012 [15]. All object instances of the 20 classes are annotated with bounding boxes and other several details (viewpoint, truncation, occlusion, difficult flags) regarding the annotated object (for more details see [15]). In our analysis, we consider the following image properties: (i) number of annotated objects; (ii) mean area covered by objects normalized by the image size; (iii) non-centeredness, defined as the mean distance of the center of all objects' bounding boxes to image center normalized by the square root of image area; (iv) number of different classes; (v) number of objects marked as truncated; (vi) number of objects marked as occluded; (vii) number of objects marked as difficult.

It is important to remark that these image properties are not available at test time. We only use them in our analysis to study how human interpretable properties correlate with visual search difficulty and also how well these properties could predict difficulty.

We quantify the correlation between image properties and visual search difficulty assessed by humans (Section 2.1) by measuring how well image properties scores can predict ground-truth human difficulty scores. More precisely, we compute the Kendall's τ correlation between the rankings of the images when ranked either by the image properties scores or by the ground-truth human difficulty scores. Each image property assigns visual search difficulty scores in a range that is different from the range of the ground-truth scores. Kendall's τ is suitable for our analysis because it is invariant to different ranges of the various measurements.

In all our experiments on visual search difficulty prediction throughout this paper, we divided the $11,540$ samples included in the official training and validation sets of PASCAL VOC 2012 into three subsets. We used 50% of the samples for training, 25% for validation and another 25% for testing. Table 2 shows the Kendall's τ rank correlations between the difficulty scores based on the image properties and the ground-truth difficulty scores on our test set. The results confirm that human interpretable properties are informative for predicting visual search difficulty. The top three most correlated image properties with the ground-truth difficulty score specify some of the ingredients that make an image difficult: the image should contain many instances of different classes scattered all over the image (not just in the center). The next most informative property is the mean area covered by objects. It shows a negative correlation with the ground-truth difficulty score suggesting that, on average, small objects are more difficult to find. Interestingly, difficulty could also be predicted to some degree based on the number of objects marked as truncated, occluded or difficult. However, as most objects appear normally, without being truncated or occluded, these markers are rarely used, which reduces their predictive power. As each image property captures a different characteristic, combining them appears to be promising. We trained a Support Vector Regression (ν-SVR) model [36] to combine all seven image properties. In our evaluation, we used the ν-SVR implementation provided in [6]. The combination yields the highest Kendall's τ correlation (0.36). In Section 3, we show that we can learn an even better predictor capable of automatically assessing visual search difficulty based on CNN features, without information derived from image properties.

2.3. Visual search difficulty at the class level

We can produce some interesting statistics based on our collection of difficulty scores. Perhaps one of the most interesting aspects is to study the difficulty scores at the class level. We compute a difficulty score per object class by averaging the score for the images that contain at least one instance of that class. The difficulty scores for all the 20 classes in PASCAL VOC 2012 are presented in Table 3. It appears that *bird*, *cat* and *aeroplane* are the easiest object classes in PASCAL that can be found in images by humans. We believe that birds and aeroplanes are easy to find as they usually appear in a simple, uniform background, for example on the sky. On the other hand, cats can appear in various contexts (simple or complex), but their distinctive shape, eyes and other body features are probably very easy to recognize. The most difficult classes in PASCAL, from a human perspective, appear to be *potted plant*, *chair*, *dining table* and *tv monitor*. We believe that potted plants and chairs are hard to find due to high (intra-class) variability in their appearance. For instance, chairs come in different shapes and sizes, such as stools, armchairs, and so on. Furthermore, all the difficult classes usually appear in complex contexts, such indoor scenes with many objects and varying illumination conditions. Interestingly, the difficulty scores presented in Table 3 indicate that the human perspective is not very different from the results achieved by state of the art computer vision systems [7, 25]. Table 3 includes the mean Average Precision (mAP) performance of the best CNN classifier presented in [7]. It can be observed that the lowest performance is obtained for the *bottle*, *potted plant* and *chair* classes. These are also among the top 5 most difficult classes for humans according to our findings. Moreover, *aeroplane* and *bird* are among the top 4 easiest classes for both humans and machines.

Class	Score	mAP	Class	Score	mAP
bird	3.081	92.5%	bicycle	3.414	90.4%
cat	3.133	91.9%	boat	3.441	89.6%
aeroplane	3.155	95.3%	car	3.463	91.5%
dog	3.208	89.7%	bus	3.504	81.9%
horse	3.244	92.2%	sofa	3.542	68.0%
sheep	3.245	82.9%	bottle	3.550	54.4%
cow	3.282	76.3%	tv monitor	3.570	74.4%
motorbike	3.355	86.9%	dining table	3.571	74.9%
train	3.360	95.5%	chair	3.583	64.1%
person	3.398	95.2%	potted plant	3.641	60.7%

Table 3. Average difficulty scores per class produced by humans versus the classification mean Average Precision (mAP) performance of the best model presented in [7] for the 20 classes available in PASCAL VOC 2012. Classes are sorted by human scores.

3. Learning to predict visual search difficulty

So far, we obtained a set of ground-truth difficulty scores based on human annotations. We now go a step further and train a model to predict the difficulty of an input image. We compare our supervised model with a handful of baseline models. We first describe our supervised model and the baseline models and then present experimental results.

3.1. Our regression model

We build our predictive model based on CNN features and linear regression with ν-SVR [36] or Kernel Ridge Regression (KRR) [36]. We considered two pre-trained CNN architectures provided in [45], namely VGG-f [7] and

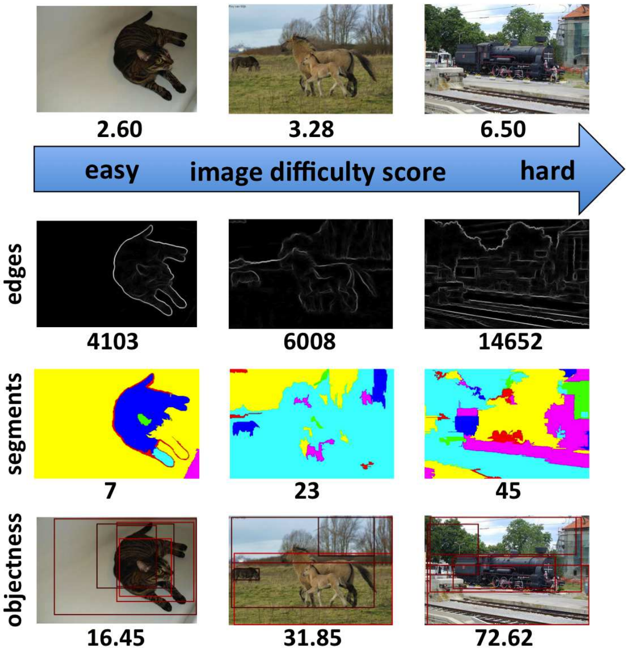

Figure 2. Visual search difficulty assessed by baselines. We show the global image features used by each baseline for computing a difficulty score. Top row: input images with predicted scores by our method. Following rows: image edge maps [13], image segmentation [16], top 5 highest scoring objectness windows (colored red to black from highest to lowest).

VGG-verydeep-16 [38]. These CNN models are trained on the ILSVRC benchmark [34].

We removed the last layer of the CNN models and used them to extract deep features as follows. The input image is divided into 1×1, 2×2 and 3×3 bins in order to obtain a pyramid representation for increased performance. The input image is also horizontally flipped and the same pyramid is applied over the flipped image. Finally, the 4096 CNN features extracted from each bin are concatenated into a single feature vector for the original input image. The final feature vectors are normalized using the L_2-norm. The normalized feature vectors are then used to train either a ν-SVR or a KRR model to regress to the ground-truth difficulty scores. We use our learned models as a continuous measure to automatically predict visual search difficulty.

3.2. Baselines

We try out several baselines. Each baseline can assess the visual search difficulty based on some specific feature: image area, file size, objectness [2], edge strengths [13], number of segments [16]. Unlike the image properties analyzed in Section 2.2, these features can be computed at test time (without manual annotations).

Random scores. We assign random scores for each image.

Image area. Without any prior information about the image content, the visual search task should be more difficult on larger images than on smaller ones. Based on this intuition, without analyzing the pixels inside an image, we quantify the difficulty of an image by its area.

File size. A similar feature for quantifying visual search difficulty without looking at the pixels is the image file size. The images in PASCAL VOC 2012 are all compressed, in JPEG format. In this context, we tried recovering the compression rate induced to the original image by normalizing the file size with the image area, but it did not provide better results than the image file size alone.

Objectness. The objectness measure [2] quantifies how likely it is for an image window to contain an object of any class. It is trained to distinguish windows containing an object with a well defined boundary and center, such as cows, cars and telephones, from windows covering amorphous background such as grass, road and sky. We used the official objectness code and obtained, for each image, a difficulty score by summing all the objectness scores of sampled windows. This difficulty score quantifies the image clutter through the objectness distribution in the $4D$ space of image windows. An easy image (Figure 2, first column) should have a small score as it contains only a small number of windows with high objectness (red colored windows) covering the dominant object. All other windows, not covering objects, have small objectness (black colored windows). Conversely, a harder image (Figure 2, last column) would have several peaks in the objectness distribution in the $4D$ space of image windows corresponding to objects' positions in the image. We tried several variants for obtaining a difficulty measure by using objectness: (i) entropy of the objectness distribution estimated with kernel density in the $4D$ space of all image windows; (ii) mean value of the objectness heat map obtained by accumulating objectness scores at each pixel for all windows containing the respective pixel; (iii) entropy of the sampled objectness windows; (iv) sum of all (usually 1000 samples obtained via the NMS sampling procedure [2]) objectness windows scores. We found out that all variants are essentially the same in terms of performance (Kendall's τ correlations between 0.20 and 0.24), with (iv) being marginally better.

Edge strengths. Humans can easily find objects in cluttered scenes by detecting their contours [35]. We use this idea to provide a measure of difficulty based on edges. Intuitively, an image with a smaller density of edges should be easier to search than another image with higher density. We use the fast edge detector of [13] to compute the edge map of an image and characterize its visual search difficulty by the sum of edge strengths.

Segments. A different way of measuring difficulty rests on using segments as features. Segments divide an image into regions of uniform texture and color. Ideally, each segment should correspond to an object or to a background region. We quantify the complexity of an image by counting

Model	MSE	Kendall τ
Random scores	0.458	0.002
Image area	-	0.052
Image file size	-	0.106
Objectness [1, 2]	-	0.238
Edge strengths [13]	-	0.240
Number of segments [16]	-	0.271
Combination with ν-SVR	0.264	0.299
VGG-f + KRR	0.259	0.345
VGG-f + ν-SVR	0.236	0.440
VGG-f + pyramid + ν-SVR	0.234	0.458
VGG-f + pyramid + flip + ν-SVR	0.233	0.459
VGG-vd + ν-SVR	0.235	0.442
VGG-vd + pyramid + ν-SVR	0.232	0.467
VGG-vd + pyramid + flip + ν-SVR	0.231	0.468
VGG-f + VGG-vd + pyramid + flip + ν-SVR	**0.231**	**0.472**

Table 4. Visual search difficulty prediction results of baseline models versus our regression models based on deep features extracted by VGG-f [7] and VGG-verydeep-16 (VGG-vd) [38]. KRR and ν-SVR are alternatively used for training our model on $5,770$ samples from PASCAL VOC 2012. The mean squared error (MSE) and the Kendall's τ correlation are computed on a test set of $2,885$ samples. The best results are highlighted in bold.

the number of segments. While turbo-pixels [27] segment the image in regular small regions, essentially providing the same number of superpixels per image, the method of [16] divides the image into irregular segments covering objects and larger portions of uniform background with fewer superpixels (Figure 2). We use the available segmenter tool of [16] with the default parameters for segmenting an image and characterize the difficulty by the number of segments.

3.3. Experimental Analysis

Evaluation measures. In order to evaluate the proposed regression model for predicting visual search difficulty, we report both the mean squared error (MSE) and the Kendall's τ rank correlation coefficient [44]. We report only the Kendall's τ correlation coefficient for the baseline models that do not involve regression, since the scores predicted by the baseline models are on a different range compared to the ground-truth difficulty scores and the MSE is a quantitative measure of performance unsuitable in this case.

Evaluation protocol. We use the same split of the data set as described in Section 2.2. The validation set is used for tuning the regularization parameters of ν-SVR and KRR.

Results. Table 4 shows the results of different methods for predicting the ground-truth difficulty. Using random scores to assess difficulty leads to almost zero accuracy, showing that visual search difficulty estimation is not a trivial problem. Baselines that do not analyze image pixels perform a little bit better but are far away from accurately predicting the order of the images based on their difficulty. The methods based on mid-level features offer an increase in accuracy. Objectness and edge strengths perform essentially the same, achieving a correlation rank around 0.24. Using segments further improves the performance to around 0.27.

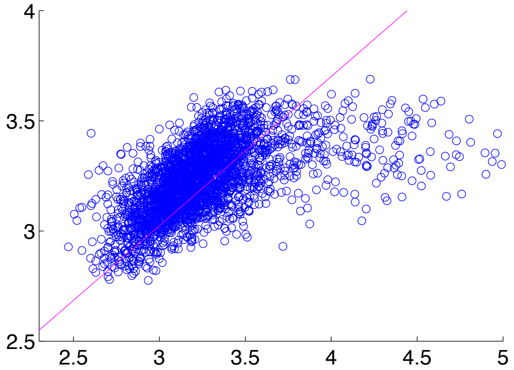

Figure 3. Correlation between ground-truth (x-axis) and predicted (y-axis) difficulty scores. The least squares regression line is almost diagonal suggesting a strong correlation.

Combining all these baselines with the ν-SVR framework, we obtain a predictor that achieves a Kendall's τ rank correlation of about 0.30. Based on the Kendall's τ definition, this translates in ranking about 65% image pairs correctly.

In training our regression models, we tested out several configurations, including two neural network architectures (VGG-f and VGG-verydeep-16), various ways of extracting features (standard, pyramid, horizontal flip), and finally, two different regression methods, namely Kernel Ridge Regression and Support Vector Regression. The least accurate configuration (VGG-f + KRR) gives already better performance compared to the baselines, reaching a rank correlation coefficient of 0.345. Changing the regression method, ν-SVR instead of KRR, we obtain a substantial increase to 0.440. The best approach is to combine the pyramid features from both CNN architectures and to train the model using ν-SVR. This combination outperforms by far all the baselines and their combination, and it remarkably achieves better performance than the image properties investigated in Section 2.2, which require knowledge of the number objects, classes, bounding boxes (unavailable at test time). The best approach based on linear regression reaches a Kendall's τ correlation coefficient of 0.472, which means that it correctly ranks about 75% image pairs. We consider the best regression model as our difficulty predictor and use it in two applications in Section 4.

Figure 3 shows the correlation between the ground-truth and the predicted difficulty scores. The cloud of points forms a slanted Gaussian with the principal component oriented almost diagonally, indicating a strong correlation between the predicted and ground-truth scores.

The examples presented in Figure 1 visually confirm the performance of our model: images with small number of objects and uniform backgrounds are ranked lower in difficulty than cluttered images with many objects and complex backgrounds. We explain the high accuracy of our model through the powerful features that capture visual abstrac-

Model	Iteration 1	Iteration 2	Iteration 3	Iteration 4	Iteration 5	Iteration 6	Iteration 7	Iteration 8	Iteration 9
Standard MIL	26.5%	29.9%	31.8%	32.7%	33.3%	33.6%	33.9%	34.3%	34.4%
Easy-to-Hard MIL	31.1%	36.1%	36.8%	38.9%	40.1%	40.8%	42.1%	42.4%	42.8%

Table 5. CorLoc results for standard MIL versus *Easy-to-Hard* MIL.

tions at a higher level, close to the level of object class recognition. Since we define difficulty based on human response times for a visual search task that involves object detection and recognition, the fact that the best features are the higher level ones makes perfect sense. Analyzing the image content at lower levels (edge strengths, objectness, segmentation) is not good enough, showing that a higher level of interpretation is needed in order to assess difficulty.

Machine versus human performance. Interestingly, when our best difficulty predictor is evaluated on the same 56 images used for computing human agreement (0.562) in Section 2.1, we obtain a Kendall's τ correlation of 0.434. Notably, our best difficulty predictor correctly ranks about 72% image pairs, which is just a little lower than the average human performance of 80% image pairs correctly ranked.

Generalization across classes. To demonstrate that our difficulty measure generalizes to classes not seen during training, we consider the setting where we train and test on disjoint PASCAL VOC 2012 classes. We train on 10 classes (*bicycle*, *bottle*, *car*, *chair*, *dining table*, *dog*, *horse*, *motorbike*, *person*, *TV monitor*) and test on the remaining 10 classes. We remove images containing both training and testing classes. The classes are split in order to exclude a minimal number of images (1601). In this setting, our ν-SVR model based on CNN features obtains a Kendall's τ correlation of 0.427, compared to 0.270 for the ν-SVR model that combines all the baselines. This result is rather close to that obtained without separating classes (0.472). Hence, this shows that our system *generalizes well across classes*.

4. Applications

We demonstrate the usefulness of our difficulty measure in two applications: weakly supervised object localization and semi-supervised object classification.

4.1. Weakly supervised object localization

In a weakly supervised object localization (WSOL) scenario, we are given a set of images known to contain instances of a certain object class. In contrast to the standard full supervision, the location of the objects is unknown. The task is to localize the objects in the input images and to learn a model that can detect new class instances in a test image. Often, WSOL is addressed as a Multiple Instance Learning (MIL) problem [5, 8, 10, 11, 37, 39, 40, 41]. In the MIL paradigm, images are treated as bags of windows (instances). A negative image contains only negative windows, while a positive image contains at least one positive window, mixed in with a majority of negative ones. The goal is to find the true positives instances from which to learn a window classifier for the object class. This typically happens by iteratively alternating two steps: (i) select instances in the positive images based on the current window classifier; (ii) update the window classifier given the current selection of positive instances and all windows from negative images.

Learning protocol. We employ our measure of difficulty as an additional cue in the standard MIL scheme for WSOL. We design a simple learning protocol that integrates the difficulty measure: rank input images by their estimated difficulty and pass them in this order to the standard MIL. We call this *Easy-to-Hard* MIL.

Evaluation protocol. We perform experiments on the training and validation sets of PASCAL VOC 2007 [14]. The main goal of WSOL is to localize the object instances in the training set. Following the standard evaluation protocol in the WSOL literature, we quantify this with the Correct Localization (CorLoc) measure [8, 9, 37, 39, 47]. For a given target class, a WSOL method outputs one window in each positive training image. CorLoc is the percentage of images where the returned window correctly localizes an object of the target class according to the PASCAL VOC criterion (intersection-over-union > 0.5 [15]).

Implementation details. We represent each image as a bag of windows extracted using the state-of-the-art object proposal method of [12]. This produces about $2,000$ windows per image. Following [5, 18, 40, 41, 47], we describe windows by the output of the second-last layer of the CNN model [25], pre-trained for whole-image classification on ILSVRC [34], using the Caffe implementation [23]. This results in 4096-dimensional features. We employ linear SVM classifiers that we train with a hard-mining procedure at each iteration. For our *Easy-to-Hard* MIL we split the images in k batches according to their difficulty. We use the easiest images (easiest batch) first, in order to update the window classifier, and progressively use more and more difficult batches. We used $k = 3$ batches and 3 iterations per batch, for a total of 9 iterations. The standard MIL baseline instead uses all images in every iteration.

Results. In Table 5, we compare the performance of our *Easy-to-Hard* MIL with the standard MIL, in terms of average CorLoc over all 20 classes. From the first iteration the improvement is already noticeable: almost $+5\%$ CorLoc. Easier images lead to a better initial class model as the MIL has a higher chance to detect class specific patterns and localize objects correctly. The improvement increases as we add more batches: $+7\%$ after the second batch and $+8.4\%$ after the third. This increase demonstrates that the order in which images are processed is important in WSOL. Processing easier images in the initial stages results in better class models that in turn improve later stages. Remarkably, our difficulty measure is trained on PASCAL VOC 2012, while here, we used it to quantify difficulty on images from PASCAL VOC 2007. As these two datasets have no images in common, the results show that our measure can *generalize across different data sets*. Finally, we point out that better CorLoc performance results have been reported on PASCAL VOC 2007 by other works using different WSOL algorithms [8, 47].

4.2. Semi-supervised object classification

Here we use our difficulty measure in a second application, namely in predicting whether an image contains a certain object class (without localizing it).

Learning protocol. We consider three sets of samples: a set L of labeled training samples, a set U of unlabeled training samples and a set T of unlabeled test samples. Our learning procedure operates iteratively, by training at each iteration a classifier on an enlarged training set L. We enlarge the training set at each iteration by moving k samples from U to L as follows: we select k samples from U based on some heuristic, we label them (positive or negative) using the current classifier and move them from U to L. We stop the learning when L reaches a certain number of samples. The final trained classifier is tested on the test set T.

Selection heuristics. To select the k samples from U at each iteration, we use one of the following heuristics: (i) select the samples randomly (RAND); (ii) select the easiest k samples based on the ground-truth difficulty scores (GTdifficulty); (iii) select the easiest k samples based on the predicted difficulty scores (PRdifficulty); (iv) select the most confident (farthest from the hyperplane) k examples from U according to the current classifier confidence score (HIconfidence); (v) select the least confident (closest to the hyperplane) k examples from U according to the current classifier (LOconfidence); (vi) select the least confident K examples from U according to the current classifier, and from these K, take the easiest k examples based on our predicted difficulty score (LOconfidence+PRdifficulty).

Evaluation protocol. We evaluate the classification performance of several models on PASCAL VOC 2012. All models are linear SVM classifiers based on CNN features [38]. We use as test set T the official PASCAL *validation* set, and we partition the PASCAL *train* set into L and U. We stopped the learning process when L reached 3 times more samples than the initial training set. We choose the initial L to have 500 labeled images randomly selected and repeat each run for 20 times to reduce the amount of variation in the results. We report the mean Average Precision (mAP) performance. We set k to 50 and K to 2000. In addition to the 6 models given by the above heuristics, we include a baseline model (*BASIC*) trained only on the initial set L. We evaluate all models on the 7 classes (*aeroplane*, *bird*, *car*, *cat*, *chair*, *dog* and *person*) from PASCAL VOC 2012 that include more than 5% positive samples. If the number of positive samples is not large enough, our semi-supervised learning protocol has trouble capturing feature patterns of the class.

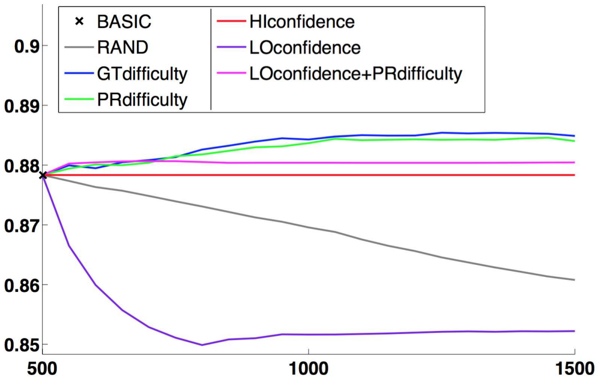

Figure 4. The mAP performance (y-axis) as the size of the training set (x-axis) grows by adding automatically labeled samples using different heuristics (compared to the BASIC baseline).

Results. Figure 4 shows the evolution of mAP for the proposed heuristics and the baseline. Randomly choosing 50 examples leads to a decrease in performance (86.1% ± 1.0%) compared to the BASIC method (87.8% ± 0.6%). Adding the most confident examples from U (HIconfidence) does not influence the results because the support vectors remain essentially the same. Using the least confident examples from U (LOconfidence) in order to change the support vectors decreases performance (85.2% ± 1.1%). The only useful information is provided by the difficulty scores, either predicted (88.4% ± 0.6%) or ground-truth (88.5% ± 0.7%), although it improves performance by less than 1%. Interestingly, by taking the least confident $2,000$ examples from U, and the easiest 50 from these examples based on our predicted difficult score (LOconfidence+PRdifficulty), we can also improve performance (88.1% ± 0.7%) by a little margin.

5. Future work

Curriculum learning [4] can help to optimize the training of deep learning models. We believe that our difficulty measure can be used in a curriculum learning setting to optimize the training of CNN models for various vision tasks.

Acknowledgments

Vittorio Ferrari was supported by the ERC Starting Grant VisCul. Radu Tudor Ionescu was supported by the POSDRU/159/1.5/S/137750 Grant. Marius Leordeanu was supported by project number PNII PCE-2012-4-0581.

References

[1] B. Alexe, T. Deselaers, and V. Ferrari. What is an object? In *Proceedings of CVPR*, pages 73–80, San Francisco, CA, USA, June 2010. IEEE.

[2] B. Alexe, T. Deselaers, and V. Ferrari. Measuring the objectness of image windows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2189–2202, 2012.

[3] S. Arun. Turning visual search time on its head. *Vision Research*, 74:86–92, 2012.

[4] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *Proceedings of ICML*, pages 41–48, New York, NY, USA, 2009. ACM.

[5] H. Bilen, M. Pedersoli, and T. Tuytelaars. Weakly supervised object detection with posterior regularization. In *Proceedings of BMVC*, 2014.

[6] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[7] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *Proceedings of BMVC*, 2014.

[8] R. Cinbis, J. Verbeek, and C. Schmid. Multi-fold mil training for weakly supervised object localization. In *Proceedings of CVPR*, 2014.

[9] T. Deselaers, B. Alexe, and V. Ferrari. Localizing Objects while Learning Their Appearance. In *Proceedings of ECCV*, 2010.

[10] T. Deselaers, B. Alexe, and V. Ferrari. Weakly Supervised Localization and Learning with Generic Knowledge. *International Journal of Computer Vision*, 100(3):275–293, September 2012.

[11] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Perez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1), 1997.

[12] P. Dollár and C. L. Zitnick. Edge boxes: Locating object proposals from edges. In *Proceedings of ECCV*, 2014.

[13] P. Dollár and C. L. Zitnick. Fast edge detection using structured forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015.

[14] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2007 Results, 2007.

[15] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 Results, 2012.

[16] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. *Internation Journal of Computer Vision*, 59(2), 2004.

[17] D. Gao and N. Vasconcelos. Bottom-up saliency is a discriminant process. In *Proceedings of ICCV*, 2007.

[18] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of CVPR*, 2014.

[19] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. In *Proceedings of CVPR*, pages 1–8, 2007.

[20] P. Isola, J. Xiao, D. Parikh, A. Torralba, and A. Oliva. What makes a photograph memorable? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1469–1482, 2014.

[21] P. Isola, J. Xiao, A. Torralba, and A. Oliva. What makes an image memorable? In *Proceedings of CVPR*, pages 145–152, 2011.

[22] S. D. Jain and K. Grauman. Predicting sufficient annotation strength for interactive foreground segmentation. In *Proceedings of ICCV*, December 2013.

[23] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional Architecture for Fast Feature Embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[24] M. G. Kendall. *Rank correlation methods*. Griffin, London, 1948.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Proceedings of NIPS*, pages 1106–1114, 2012.

[26] I. Laurent, K. Christof, and N. Ernst. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, 1998.

[27] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi. TurboPixels: Fast Superpixels Using Geometric Flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12), 2009.

[28] D. Liu, Y. Xiong, K. Pulli, and L. Shapiro. Estimating image segmentation difficulty. In *Proceedings of MLDM*, pages 484–495, 2011.

[29] Y. Luo and X. Tang. Photo and video quality evaluation: Focusing on the subject. In *Proceedings of ECCV*, 2008.

[30] L. Marchesotti, C. Cifarelli, and G. Csurka. A framework for visual saliency detection with applications to image thumbnailing. In *Proceedings of ICCV*, 2009.

[31] F. Moosmann, D. Larlus, and F. Jurie. Learning saliency maps for object categorization. In *Proceedings of ECCV*, 2006.

[32] A. Oliva. Gist of the scene. *Neurobiology of Attnetion, Elsevier*, pages 251–256, 2005.

[33] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *International Journal on Computer Vision*, 42:145–175, 2001.

[34] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, K. A., A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 2015.

[35] R. M. Shapley and D. J. Tolhurst. Edge detectors in human vision. *Journal of Physiology*, 229(1):165–183, 1973.

[36] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[37] Z. Shi, P. Siva, and T. Xiang. Transfer learning by ranking for weakly supervised object annotation. In *Proceedings of BMVC*, 2012.

[38] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[39] P. Siva and T. Xiang. Weakly Supervised Object Detector Learning with Model Drift Detection. In *Proceedings of ICCV*, 2011.

[40] H. Song, R. Girshick, S. Jegelka, J. Mairal, Z. Harchaoui, and T. Darell. On learning to localize objects with minimal supervision. In *Proceedings of ICML*, 2014.

[41] H. Song, Y. Lee, S. Jegelka, and T. Darell. Weakly-supervised discovery of visual pattern configurations. In *Proceedings of ICML*, 2014.

[42] M. Spain and P. Perona. Some objects are more equal than others: Measuring and predicting importance. In *Proceedings of ECCV*, 2008.

[43] L. M. Trick and J. T. Enns. Lifespan changes in attention: The visual search task. *Cognitive Development*, 13(3):369–386, 1998.

[44] G. Upton and I. Cook. *A Dictionary of Statistics*. Oxford University Press, Oxford, 2004.

[45] A. Vedaldi and K. Lenc. MatConvNet – Convolutional Neural Networks for MATLAB. In *Proceeding of ACMMM*, 2015.

[46] S. Vijayanarasimhan and K. Grauman. Whats It Going to Cost You?: Predicting Effort vs. Informativeness for Multi-Label Image Annotations. In *Proceedings of CVPR*, 2009.

[47] C. Wang, W. Ren, J. Zhang, K. Huang, and S. Maybank. Large-scale weakly supervised object localization via latent category learning. *IEEE Transactions on Image Processing*, 2015.

[48] J. M. Wolfe, E. M. Palmer, and T. S. Horowitz. Reaction time distributions constrain models of visual search. *Vision Research*, 50(14):1304–1311, 2010.