Unbiased Look at Dataset Bias

Antonio Torralba
Massachusetts Institute of Technology
torralba@csail.mit.edu

Alexei A. Efros
Carnegie Mellon University
efros@cs.cmu.edu

Abstract

Datasets are an integral part of contemporary object recognition research. They have been the chief reason for the considerable progress in the field, not just as source of large amounts of training data, but also as means of measuring and comparing performance of competing algorithms. At the same time, datasets have often been blamed for narrowing the focus of object recognition research, reducing it to a single benchmark performance number. Indeed, some datasets, that started out as data capture efforts aimed at representing the visual world, have become closed worlds unto themselves (e.g. the Corel world, the Caltech-101 world, the PASCAL VOC world). With the focus on beating the latest benchmark numbers on the latest dataset, have we perhaps lost sight of the original purpose?

The goal of this paper is to take stock of the current state of recognition datasets. We present a comparison study using a set of popular datasets, evaluated based on a number of criteria including: relative data bias, cross-dataset generalization, effects of closed-world assumption, and sample value. The experimental results, some rather surprising, suggest directions that can improve dataset collection as well as algorithm evaluation protocols. But more broadly, the hope is to stimulate discussion in the community regarding this very important, but largely neglected issue.

1. Introduction

It is a capital mistake to theorize before one has data.

SHERLOCK HOLMES

Let's play a game we call *Name That Dataset!* Shown in Figure 1 are three most discriminable (to be explained in a moment) images from twelve popular recognition datasets. The goal is to guess which images came from which dataset (go ahead, try it, we will wait... finished? Now check your answers below[1]). In theory, this should be a very difficult task, considering that the datasets contain thousands to millions of images. Moreover, most of these datasets were collected with the expressed goal of being as varied and rich as possible, aiming to sample the visual world "in the wild". Yet in practice, this task turns out to be relatively easy for anyone who has worked in object and scene recognition (in our labs, most people got more than 75% correct).

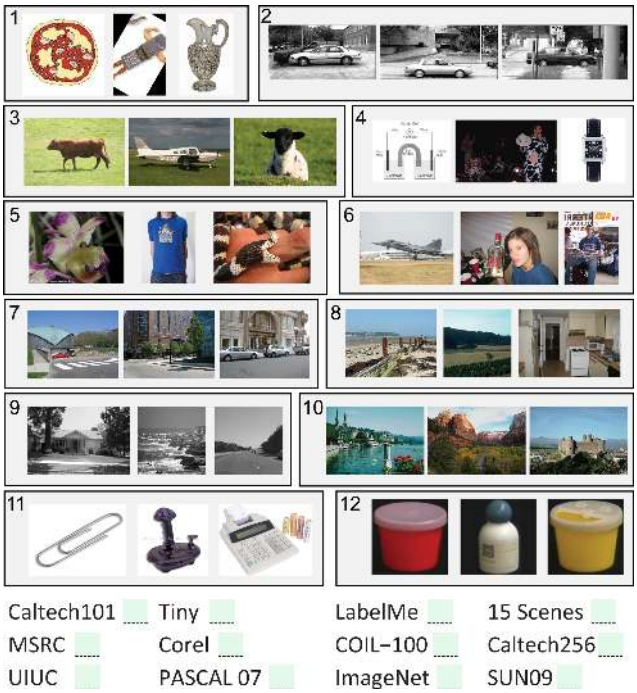

Figure 1. Name That Dataset: Given three images from twelve popular object recognition datasets, can you match the images with the dataset? (answer key below)

Intrigued, we decided to perform a toy experiment: to train a classifier to play *Name That Dataset*. We randomly sampled 1000 images from the training portions of each of the 12 datasets, and trained a 12-way linear SVM classifier. The classifier was tested on 300 random images from each of the test sets, repeated 20 times. Figure 2(left) shows classifier performance for four popular image descriptors (32x32 thumbnail, both grayscale and color [18], gist [13], and bag of HOG [1] visual words) as a function of training set size (log scale). Curiously, the best classifier performs rather well at 39% (chance is $1/12 = 8\%$), and what is even more intriguing – there is no evidence of saturation as more training data is added.

[1] Answer key: 1) Caltech-101, 2) UIUC, 3) MSRC, 4) Tiny Images, 5) ImageNet, 6) PASCAL VOC, 7) LabelMe, 8) SUNS-09, 9) 15 Scenes, 10) Corel, 11) Caltech-256, 12) COIL-100.

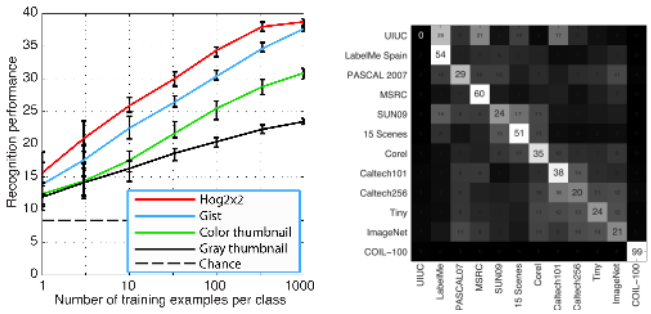

Figure 2. Computer plays *Name That Dataset*. Left: classification performance as a function of dataset size (log scale) for different descriptors (notice that performance does not appear to saturate). Right: confusion matrix.

Figure 2(right) shows the confusion matrix, grouped by similarity. Apart from two outliers (UIUC test set is not the same as its training set and COIL is a lab-based dataset), there is strong grouping of scene-centric datasets vs. object-centric datasets (in the latter, Caltech101 and Caltech256 are predictably confused with each other). Still, despite the small sample size, there is a clearly pronounced diagonal, suggesting that each dataset possesses a unique, identifiable "signature". We can try to visualize this signature by looking at the most discriminable images within each dataset, i.e. the images placed furthest from the decision boundary by the SVM. That is, in fact, what we did for Figure 1. We could also do the opposite: for a given dataset, look at the images placed closest to the decision boundary separating it from another dataset (Figure 3). This shows how one dataset can "impersonate" a different dataset.

The lesson from this toy experiment is that, despite the best efforts of their creators, the datasets appear to have a strong build-in bias. Of course, much of the bias can be accounted for by the divergent goals of the different datasets: some captured more urban scenes, others more rural landscapes; some collected professional photographs, others the amateur snapshots from the Internet; some focused on entire scenes, others on single objects, etc. Yet, even if we try to control for these capture biases by isolating specific objects of interest, we find that the biases are still present in some form. As a demonstration, we applied the same analysis that we did for full images to object crops of cars from five datasets where car bounding boxes have been provided (PASCAL, ImageNet, SUN09, LabelMe, Caltech101). Interestingly, the classifier was still quite good at telling the different datasets apart, giving 61% performance (at 20% chance). Visually examining the most discriminable cars (Figure 4), we observe some subtle but significant differences: Caltech has a strong preference for side views, while ImageNet is into racing cars; PASCAL have cars at non-canonical view-points; SUNS and LabelMe cars appear to be similar, except LabelMe cars are often occluded by small objects, etc. Clearly, whatever we, as a community, are trying to do to get rid of dataset bias is not quite working.

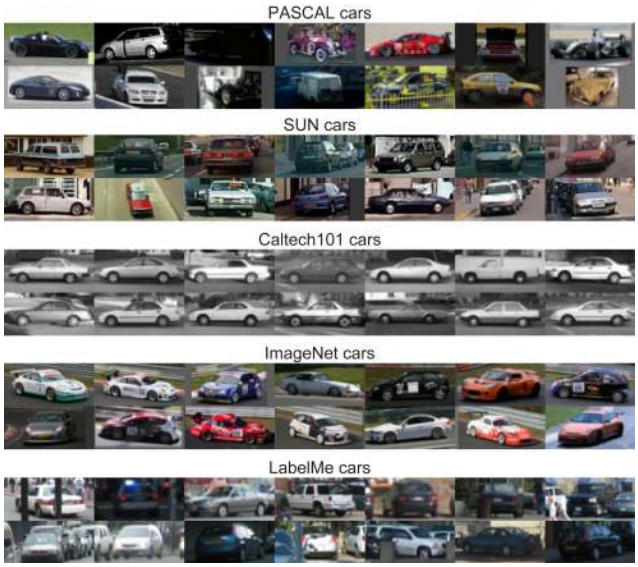

Figure 4. Most discriminative cars from 5 datasets

Hence the aim of this paper is two-fold. First, to try to understand some of the subtle ways in which bias sneaks into our datasets and affects detection and classification performance. Second, to raise awareness in the recognition community about this important issue that is, sadly, not getting the attention it deserves.

2. Prologue: The Promise and Perils of Visual Datasets

We are in the midst of a data revolution. Ubiquitous access to image datasets has been responsible for much of the recent progress in object recognition [14] after decades of proverbial wandering in the desert. For instance, it was the availability of face training data (both positive *and* negative), more than the perceived advances in machine learning, that produced the first breakthrough in face detection [15] (while neural networks were central to [15], subsequent approaches showed similar performance with very different techniques). And it is the dataset of millions of photographs of consumer products, as much as the clever feature matching, that allowed visual search engines like GOOGLE GOGGLES to become reality. Datasets have also played the leading role in making object recognition research look less like a black art and more like an experimental science. The fact that today, unlike just a few years ago, it is virtually impossible to publish a CVPR paper in recognition without a quantitative evaluation attests to the sea-change brought forward by data.

Alas, like any proper revolution, this one has brought with it new problems to replace the old ones. Many people are worried that the field is now getting *too* obsessed with evaluation, spending more time staring at precision-recall curves than at pixels. There is concern that research is becoming too incremental, since a completely new approach will initially have a hard time competing against estab-

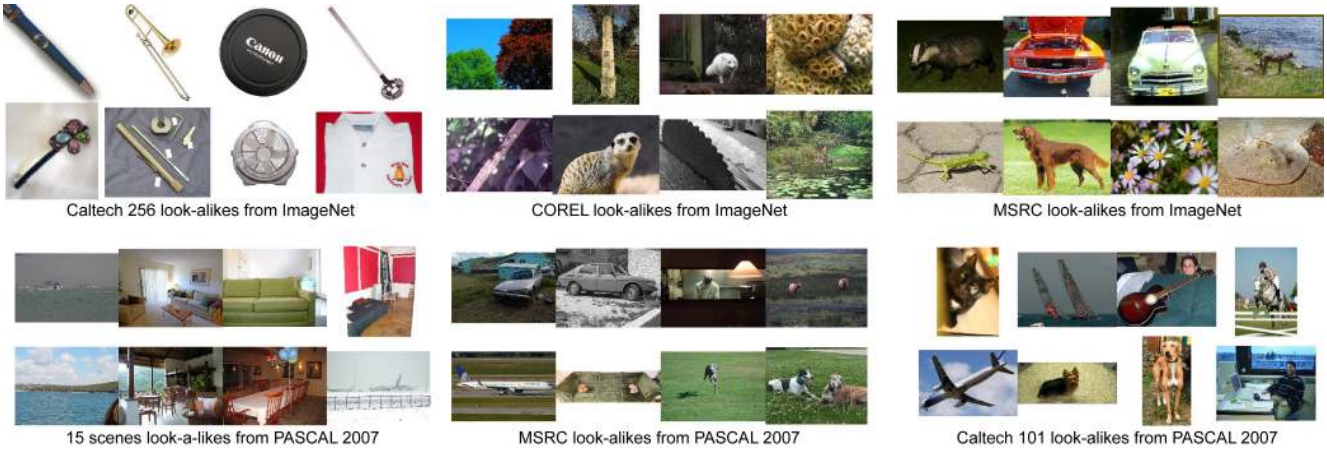

Figure 3. Dataset Look-alikes: Above, ImageNet is trying to impersonate three different datasets. Here, the samples from ImageNet that are closest to the decision boundaries of the three datasets are displayed. Look-alikes using PASCAL VOC are shown below.

lished, carefully fine-tuned methods. For instance, one major issue for many popular dataset competitions is "creeping overfitting", as algorithms over time become too adapted to the dataset, essentially memorizing all its idiosyncrasies, and losing ability to generalize [14]. Fortunately, this problem can be greatly alleviated by either changing the dataset regularly (as done in PASCAL VOC 2005-2007), or withholding the test set and limiting the number of times a team can request evaluation on it (as done in PASCAL VOC 2008+ and Caltech Pedestrian benchmark [4]).

Another concern is that our community gives too much value to "winning" a particular dataset competition, regardless of whether the improvement over other methods is statistically significant. For PASCAL VOC, Everingham et al [6] use the Friedman/Nemenyi test, which, for example, showed no statistically significant difference between the eight top-ranked algorithms in the 2010 competition. More fundamentally, it may be that the right way to treat dataset performance numbers is not as a competition for the top place, but rather as a sanity check for new algorithms and an efficient way of comparing against multiple baselines. This way, fundamentally new approaches will not be forced to compete for top performance right away, but will have a chance to develop and mature.

Luckily, the above issues are more behavioral rather than scientific, and should be alleviated as our field develops benchmarking best practices similar to those in other fields. However, there is a more fundamental question: *are the datasets measuring the right thing*, that is, the expected performance on some real-world task? Unlike datasets in machine learning, where the dataset *is* the world, computer vision datasets are supposed to be a *representation* of the world. Yet, what we have been witnessing is that our datasets, instead of helping us train models that work in the real open world, have become closed worlds unto themselves, e.g. the Corel world, the Caltech101 world, the PASCAL VOC world, etc. This is particularly unfortunate since, historically, the development of visual datasets has been driven, in no small part, by the desire to be a better, more authentic representation of the visual world.

2.1. The Rise of the Modern Dataset

Any good revolution needs a narrative of struggle against perceived unfairness and bias, and the history of dataset development certainly provides that. From the very beginning, every new dataset was, in a way, a reaction against the biases and inadequacies of the previous datasets in explaining the visual world. The famous single-image-dataset *Lena*, one of the first "real" images (digitized in 1972 from a PLAYBOY centerfold) was a reaction against all the carefully controlled lab stock images, the "dull stuff dating back to television standards work" [10]. In the same spirit, the COIL-100 dataset [12] (a hundred household objects on a black background) was a reaction against model-based thinking of the time (which focused mostly on staplers), and an embrace of data-driven appearance models that could capture textured objects like Tylenol bottles. Professional collections like Corel Stock Photos and 15 Scenes [13] were a reaction against the simple COIL-like backgrounds and an embrace of visual complexity. Caltech-101 [7] (101 objects mined using Google and cleaned by hand) was partially a reaction against the professionalism of Corel's photos, and an embrace of the wilderness of the Internet. MSRC [19] and LabelMe [16] (both researcher-collected sets), in their turn, were a reaction against the Caltech-like single-object-in-the-center mentality, with the embrace of complex scenes with many objects [14]. PASCAL Visual Object Classes (VOC) [6] was a reaction against the lax training and testing standards of previous datasets [14]. Finally the batch of very-large-scale, Internet-mined datasets – Tiny Images [18], ImageNet [3], and SUN09 [20] – can be considered a reaction against the inadequacies of training and testing on datasets that are just too small for the complexity of the real world.

On the one hand, this evolution in the development of datasets is perhaps a sign of progress. But on the other hand,

one could also detect a bit of a vicious cycle. Time and again, we as a community reject the current datasets due to their perceived biases. Yet time and again, we create new datasets that turn out to suffer from much the same biases, though differently manifested. What seems missing, then, is a clear understanding of the types and sources of bias, without which, we are doomed to repeat our mistakes.

3. Measuring Dataset Bias

Granted, it makes little sense to talk about a bias-free representation of the visual world without specifying the observer and the task (e.g. a wandering human perceives a completely different visual reality than, say, a wandering bird). Still, for the purposes of object recognition, most existing datasets assume roughly the same general task: given the typical visual environments encountered by people, to detect commonly occurring objects. Using that as the definition of our visual world, can we evaluate how well does a particular dataset represent it? Alas, to correctly measure a dataset's bias would require comparing it to the real visual world, which would have to be in form of a dataset, which could also be biased... so, not a viable option. So, here we will settle for a few standard checks, a diagnostic of dataset health if you will.

3.1. Cross-dataset generalization

The biggest warning sign that something is rotten in the state of today's datasets is that there are virtually no papers demonstrating cross-dataset generalization, e.g. training on ImageNet, while testing on PASCAL VOC (but see [4] for an encouraging study). Surely, if our datasets were truly representative of the real world, this would be a very easy thing to do, and would give access to more of the much needed labelled data. To be sure, there are methods for transferring a model learned on one dataset onto another [21, 5, 17], where the target dataset is considered to be in a different "domain". But from our perspective, all the datasets are really trying to represent the same domain – our visual world – and we would like to measure how well or badly they do it.

So, we would like to ask the following question: how well does a typical object detector trained on one dataset generalize when tested on a representative set of other datasets, compared with its performances on the "native" test set? To answer this question, we picked a set of six representative datasets that are: 1) in active research use today, and 2) have some annotated objects in common: SUN09 [20], LabelMe [16], PASCAL VOC 2007 [6], ImageNet [3], Caltech-101 [7], and MSRC [19]. Since each of the datasets has objects labeled with bounding boxes, two testing regimes are possible: a) classification – find all images containing the desired object; and b) detection – in all images, find all bounding boxes containing the desired object. Notice that the detection task is basically the same as classification if you think of bounding boxes as images – those that contain the object are positives, these that don't are negatives. Importantly, for detection the number of negatives is naturally much larger and more diverse.

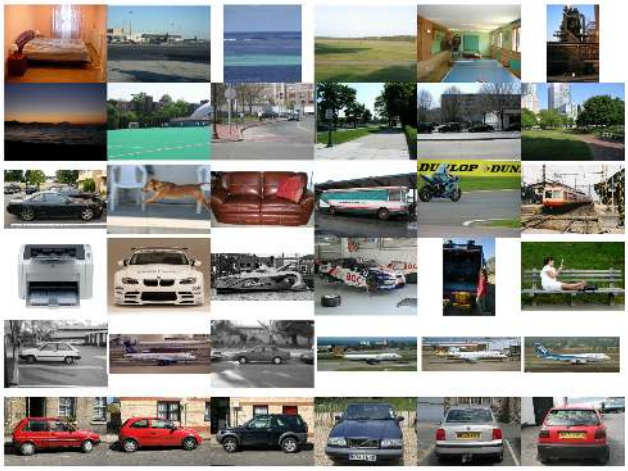

Figure 5. Cross-dataset generalization for "car" classification (full image) task, trained on MSRC and tested on (one per row): SUN, LabelMe, PASCAL, ImageNet, Caltech-101, and MSRC.

For the object detection task, we use the most standard, off-the-shelf approach of Dalal&Triggs [1] (HOG detector followed by a linear SVM), that has been quite popular in recent years, and is the basis of the currently best-performing detector of Felzenszwalb et al [8]. Likewise, for the classification task, we used the most standard and popular bag-of-words approach with a non-linear SVM (Gaussian kernel). We picked two objects that were common among all the datasets and are also popular with various algorithms: "car" and "person". Each classifier was trained with 500 positive and 2000 negative for the classification task and 100 positive and 1000 negative examples for the detection task for each dataset (these numbers were the maximum ones possible because some of the datasets are quite small). The test was performed with 50 positive and 1000 negative examples for classification and 10 positive and 20000 negative for detection. For testing, each classifier was run 20 times and the results averaged.

Table 1 shows a summary of results. Each column corresponds to the performance obtained when testing on one dataset and training on all datasets. Each row corresponds to training on one dataset and testing on all the others. Note that since our training and testing protocol is necessarily different from the ones traditionally used for each of the datasets, the actual performance numbers will not be too meaningful; rather it's the differences in performance which are telling. The first observation is that, as expected, the best results are typically when training and testing on the same dataset. By looking at the values across one row, we can evaluate how good is one dataset at generalizing over the others. By looking at the values across each column, we can evaluate how easy is one dataset for the other datasets. As

Table 1. Cross-dataset generalization. Object detection and classification performance (AP) for "car" and "person" when training on one dataset (rows) and testing on another (columns), i.e. each row is: training on one dataset and testing on all the others. "Self" refers to training and testing on the same dataset (same as diagonal), and "Mean Others" refers to averaging performance on all except self.

task	Train on: \ Test on:	SUN09	LabelMe	PASCAL	ImageNet	Caltech101	MSRC	Self	Mean others	Percent drop
"car" classification	SUN09	**28.2**	29.5	16.3	14.6	16.9	21.9	28.2	19.8	**30%**
	LabelMe	14.7	**34.0**	16.7	22.9	43.6	24.5	34.0	24.5	**28%**
	PASCAL	10.1	25.5	**35.2**	43.9	44.2	39.4	35.2	32.6	**7%**
	ImageNet	11.4	29.6	36.0	**57.4**	52.3	42.7	57.4	34.4	**40%**
	Caltech101	7.5	31.1	19.5	33.1	**96.9**	42.1	96.9	26.7	**73%**
	MSRC	9.3	27.0	24.9	32.6	40.3	**68.4**	68.4	26.8	**61%**
	Mean others	10.6	28.5	22.7	29.4	39.4	34.1	53.4	27.5	48%
"car" detection	SUN09	**69.8**	50.7	42.2	42.6	54.7	69.4	69.8	51.9	**26%**
	LabelMe	61.8	**67.6**	40.8	38.5	53.4	67.0	67.6	52.3	**23%**
	PASCAL	55.8	55.2	**62.1**	56.8	54.2	74.8	62.1	59.4	**4%**
	ImageNet	43.9	31.8	46.9	**60.7**	59.3	67.8	60.7	49.9	**18%**
	Caltech101	20.2	18.8	11.0	31.4	**100**	29.3	100	22.2	**78%**
	MSRC	28.6	17.1	32.3	21.5	67.7	**74.3**	74.3	33.4	**55%**
	Mean others	42.0	34.7	34.6	38.2	57.9	61.7	72.4	44.8	**48%**
"person" classification	SUN09	**16.1**	11.8	14.0	7.9	6.8	23.5	16.1	12.8	**20%**
	LabelMe	11.0	**26.6**	7.5	6.3	8.4	24.3	26.6	11.5	**57%**
	PASCAL	11.9	11.1	**20.7**	13.6	48.3	50.5	20.7	27.1	**-31%**
	ImageNet	8.9	11.1	11.8	**20.7**	76.7	61.0	20.7	33.9	**-63%**
	Caltech101	7.6	11.8	17.3	22.5	**99.6**	65.8	99.6	25.0	**75%**
	MSRC	9.4	15.5	15.3	15.3	93.4	**78.4**	78.4	29.8	**62%**
	Mean others	9.8	12.3	13.2	13.1	46.7	45.0	43.7	23.4	**47%**
"person" detection	SUN09	**69.6**	56.8	37.9	45.7	52.1	72.7	69.6	53.0	**24%**
	LabelMe	58.9	**66.6**	38.4	43.1	57.9	68.9	66.6	53.4	**20%**
	PASCAL	56.0	55.6	**56.3**	55.6	56.8	74.8	56.3	59.8	**-6%**
	ImageNet	48.8	39.0	40.1	**59.6**	53.2	70.7	59.6	50.4	**15%**
	Caltech101	24.6	18.1	12.4	26.6	**100**	31.6	100	22.7	**77%**
	MSRC	33.8	18.2	30.9	20.8	69.5	**74.7**	74.7	34.6	**54%**
	Mean others	44.4	37.5	31.9	38.4	57.9	63.7	71.1	45.6	**36%**

one could expect, both Caltech 101 and MSRC are the easiest datasets (column averages) across all tasks. PASCAL and ImageNet are, most of the time, the datasets that generalize the best (row averages), although they score higher in object-centric datasets such as Caltech 101 and MSRC, than in scene-centric datasets such as SUN09 and LabelMe. In general there is a dramatic drop of performance in all tasks and classes when testing on a different test set. For instance, for the "car" classification task the average performance obtained when training and testing on the same dataset is 53.4% which drops to 27.5%. This is a very significant drop that would, for instance, make a method ranking first in the PASCAL competition become one of the worst. Figure 5 shows a typical example of car classification gone bad. A classifier trained on MSRC "cars" has been applied to six datasets, but it can only find cars in one – MSRC itself.

Overall the results look rather depressing, as little generalization appears to be happening beyond the given dataset. This is particularly surprising given that most datasets are collected from the same source – the Internet. Why is this happening? There are likely several culprits. First, there is clearly some **selection bias**, as we've shown in Section 1 – datasets often prefer particular kinds of images (e.g. street scenes, or nature scenes, or images retrieved via Internet keyword searches). Second, there is probably some **capture bias** – photographers tending to take pictures of objects in similar ways (although this bias might be similar across the different datasets). Third, there is **category or label bias**. This comes from the fact that semantic categories are often poorly defined, and different labellers may assign differing labels to the same type of object [11] (e.g. "grass" vs. "lawn", "painting" vs. "picture"). Finally, there is the **negative set bias**. The negative set defines what the dataset considers to be "the rest of the world". If that set is not representative, or unbalanced, that could produce classifiers that are overconfident and not very discriminative. Of all the above, the negative set bias seems to receive the least attention, so in the next section we will investigate it in more detail.

3.2. Negative Set Bias

Datasets define a visual phenomenon (e.g. object, scene, event) not just by what it is (positive instances), but also by *what it is not* (negative instances). Alas, the space of *all* possible negatives in the visual world is astronomically large, so datasets are forced to rely on only a small sample.

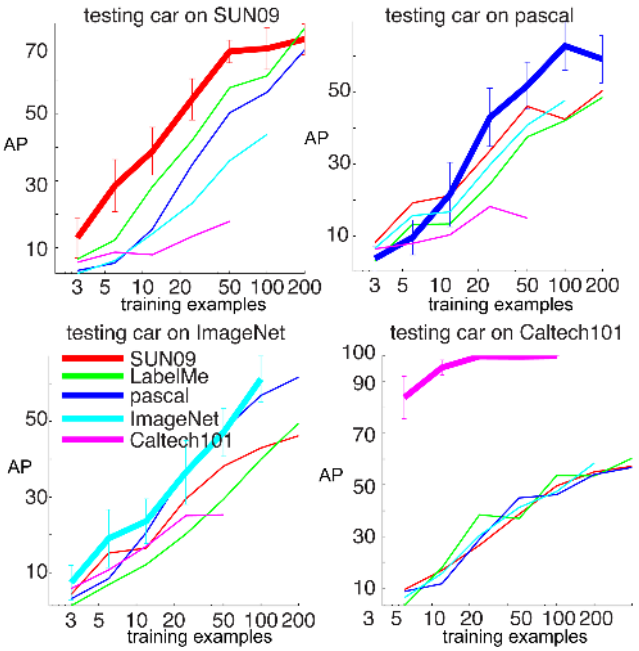

Figure 6. Cross-dataset generalization for "car" detection as function of training data

But is this negative sample representative or even sufficient?

To answer the first question, we designed an experiment to evaluate the relative bias in the negative sets of different datasets (e.g. is a "not car" in PASCAL different from "not car" in MSRC?). The idea is to approximate the real-world negative set by a super-set of dataset negatives via combining the negative sets of each of the 6 datasets in the evaluation pool. First, for each dataset, we train a classifier on its own set of positive and negative instances. Then, during testing, the positives come from that dataset, but the negatives come from all datasets combined. The number of negatives is kept the same as the number of negatives of the original test, to keep chance performance at the same level. We ran a detection task with 100 positives and 1000 negatives. For testing, we did multiple runs of 10 positive examples for 20,000 negatives.

The results for "car" and "person" detection are shown in Table 2. Each column shows performance obtained when training on each of the 6 datasets evaluated, and testing on 1) the original test set and 2) on a new negative test super-set. For three popular datasets (SUN09, LabelMe and PASCAL) we observe a significant (20%) decrease in performance, suggesting that some of the new negative examples coming from other datasets are confounded with positive examples. On the other hand, ImageNet, Caltech 101 and MSRC do not show a drop. The reasons for this lack of change are likely different for each dataset. ImageNet benefits from a large variability of negative examples and does not seem to be affected by a new external negative set, whereas Caltech and MSRC appear to be just too easy.

A much harder question is whether the negative data sample is *sufficient* to allow a classifier to tease apart the important bits of the visual experience. This is particularly important for classification tasks, where the number of negatives is only a few orders of magnitude larger than the number of positives for each class. For example, if we want to find all images of "boats" in a PASCAL VOC-like classification task setting, how can we make sure that the classifier focuses on the boat itself, and not on the water below, or shore in the distance (after all, all boats are depicted in water)? This is where a large negative set (including rivers, lakes, sea, etc, without boats) is imperative to "push" the lazy classifier into doing the right thing. Unfortunately, it's not at all easy to stress-test the sufficiency of a negative set in the general case since it will require huge amounts of labelled (and unbiased) negative data. While beyond the scope of the present paper, we plan to evaluate this issue more fully, perhaps with the help of Mechanical Turk.

4. Measuring Dataset's Value

Given a particular detection task and benchmark, there are two basic ways of improving the performance. The first solution is to improve the features, the object representation and the learning algorithm for the detector. The second solution is to simply enlarge the amount of data available for training. However, increasing the amount of training data might be harder than it seems. The first issue is that to achieve a significant improvement in performance, the increase in training data must be very significant (performance has an annoying logarithmic dependency on amount of training data). The second issue is that, as discussed in the previous section, if we add training data that does not match the biases of the test data this will result in a less effective classifier.

These two problems are illustrated in Fig. 6. As shown, performance increases with increasing dataset size (note the log scaling for the horizontal axis), and the performance using data from another dataset almost always under the performance with the original training dataset. The vertical gap between two curves represents the decrease in performance resulting from training on a different dataset. The horizontal shift corresponds to the increase in amount of data needed to reach the same level of performance. One could also expect that different datasets saturate at different points, but there are no signs of saturation in the experiments we performed, with a linear relationship, in a log-log plot, between the amount of training data, $log(n)$, and the error $log(1-AP)$, with AP being average precision-recall.

Let's say we decide to increase the amount of training data available in PASCAL VOC. We will undoubtedly first check what alternative sources of data are available, e.g. can we use, say, LabelMe to improve our performance? The question is: what is the relative value of a training example from LabelMe with respect to the value of a training example from PASCAL?

Table 2. Measuring Negative Set Bias.

task	Positive Set: / Negative Set:	SUN09	LabelMe	PASCAL	ImageNet	Caltech101	MSRC	Mean
"car" detection	self	67.6	62.4	56.3	60.5	97.7	74.5	70.0
	all	53.8	51.3	47.1	65.2	97.7	70.0	64.1
	percent drop	20%	18%	16%	-8%	0%	6%	8%
"person" detection	self	67.4	68.6	53.8	60.4	100	76.7	71.1
	all	52.2	58.0	42.6	63.4	100	71.5	64.6
	percent drop	22%	15%	21%	-5%	0%	7%	9%

Table 3. "Market Value" for a "car" sample across datasets

	SUN09 market	LabelMe market	PASCAL market	ImageNet market	Caltech101 market
1 SUN09 is worth	1 SUN09	0.91 LabelMe	0.72 pascal	0.41 ImageNet	0 Caltech
1 LabelMe is worth	0.41 SUN09	1 LabelMe	0.26 pascal	0.31 ImageNet	0 Caltech
1 pascal is worth	0.29 SUN09	0.50 LabelMe	1 pascal	0.88 ImageNet	0 Caltech
1 ImageNet is worth	0.17 SUN09	0.24 LabelMe	0.40 pascal	1 ImageNet	0 Caltech
1 Caltech101 is worth	0.18 SUN09	0.23 LabelMe	0 pascal	0.28 ImageNet	1 Caltech
Basket of Currencies	0.41 SUN09	0.58 LabelMe	0.48 pascal	0.58 ImageNet	0.20 Caltech

Given the performance $AP_i^j(n)$ obtained when training on dataset i and testing on dataset j as a function of the number of training samples n, we define the sample value (α) as $Ap_j^j(n) = Ap_i^j(n/\alpha)$. In the plots of Fig. 6 this corresponds to a horizontal shift and can be estimated as the shift needed to align each pair of graphs. For instance, 1 LabelMe car sample is worth 0.26 PASCAL car samples on the PASCAL benchmark. This means that if we want to have a modest increase (maybe 10% AP) in performance on the car detector trained with 1250 PASCAL samples available on PASCAL VOC 2007, we will need $1/0.26 \times 1250 \times 10 = 50000$ LabelMe samples!

Table 3 shows the "market value" of training samples from different datasets[2]. One observation is that the sample values are always smaller than 1 – each training sample gets devalued if it is used on a different dataset. There is no theoretical reason why this should be the case and it is only due to the strong biases present in actual datasets. So, what is the value of current datasets when used to train algorithms that will be deployed in the real world? The answer that emerges can be summarized as: "better than nothing, but not by much".

5. Discussion

Is it to be expected that when training on one dataset and testing on another there is a big drop in performance? One could start by arguing that the reason is not that datasets are bad, but that our object representations and recognition algorithms are terrible and end up over-learning aspects of the visual data that relates to the dataset and not to the ultimate visual task. In fact, a human learns about vision by living in a reduced environment with many potential local biases and yet the visual system is robust enough to overcome this. However, let us not put all the blame on the algorithms, at least not yet. If a dataset defines a "car" to be the rear view of a race-car, then there is no reasonable algorithm that will say that a side view of a family sedan is also a "car".

So, how well do the currently active recognition datasets stack up overall? Unsurprisingly, our results show that Caltech-101 is extremely biased with virtually no observed generalization, and should have been retired long ago (as argued by [14] back in 2006). Likewise, MSRC has also fared very poorly. On the other hand, most modern datasets, such as PASCAL VOC, ImageNet and SUN09, have fared comparatively well, suggesting that perhaps things are starting to move in the right direction.

Should we care about the quality of our datasets? If the goal is to reduce computer vision to a set of feature vectors that can be used in some machine learning algorithm, then maybe not. But if the goal is to build algorithms that can understand the visual world, then, having the right datasets will be crucial. In next section we outline some recommendations for developing better datasets.

6. Epilogue

Is there any advice that can be offered to researchers thinking of creating a new dataset on how to detect and avoid bias? We think that a good first step would be to run any new dataset on the battery of tests that have been outlined in this paper (we will be happy to publish all code and data online). While this will not detect all potential sources of bias, it might help finding the main problematic issues quickly and early, not years after the dataset has been released. What about tips on how to avoid, or at least minimize, the effects of bias during the dataset construction itself? Here we briefly go over a few suggestions for minimizing each type of bias:

Selection Bias: As suggested by Figure 2, datasets that are gathered automatically fare better than these collected manually. However, getting images from the Internet does not in itself guarantee a fair sampling, since keyword-based searches will return only particular types of images. Obtaining data from multiple sources (e.g. multiple search engines

[2]We have also experimented with "incremental market value" – how much does data from other datasets help *after* using all the original data. We found that this quickly converges to the absolute "market value".

from multiple countries [3]) can somewhat decrease selection bias. However, it might be even better to start with a large collection of unannotated images and label them by crowd-sourcing.

Capture Bias: Professional photographs as well as photos collected using keyword search appear to suffer considerably from the capture bias. The most well-known bias is that the object is almost always in the center of the image. Searching for "mug" on Google Image Search will reveal another kind of capture bias: almost all the mugs has a right-facing handle. Beyond better data sampling strategies, one way to deal with this is to perform various data transformations to reduce this bias, such as flipping images left-right [8, 9] (but note that any text will appear the wrong way), or jittering the image [2], e.g. via small affine transformations [18]. Another fruitful direction might be generating various automatic crops of the image.

Negative Set Bias: As we have shown, having a rich and unbiased negative set is important to classifier performance. Therefore, datasets that only collect the things they are interested in might be at a disadvantage, because they are not modeling the rest of the visual world. One remedy, proposed in this paper, is to add negatives from other datasets. Another approach, suggested by Mark Everingham, is to use a few standard algorithms (e.g. bag of words) to actively mine hard negatives as part of dataset construction from a very large unlabelled set, and then manually going through them to weed out true positives. The down side is that the resulting dataset will be biased against existing algorithms.

This paper is only the start of an important conversation about datasets. We suspect that, despite the title, our own biases have probably crept into these pages, so there is clearly much more to be done. All that we hope is that our work will start a dialogue about this very important and underapprechiated issue.

Acknowledgements: The authors would like to thank the Eyjafjallajokull volcano as well as the wonderful *kirs* at the Buvette in Jardin du Luxembourg for the motivation (former) and the inspiration (latter) to write this paper. This work is part of a larger effort, joint with David Forsyth and Jay Yagnik, on understanding the benefits and pitfalls of using large data in vision. The paper was co-sponsored by ONR MURIs N000141010933 and N000141010934.

Disclaimer: No graduate students were harmed in the production of this paper. Authors are listed in order of increasing procrastination ability.

References

[1] N. Dalal and B. Triggs. Histogram of oriented gradients for human detection. 2005. 1521, 1524

[2] D. DeCoste and M. Burl. Distortion-invariant recognition via jittered queries. In *CVPR*, 2000. 1528

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. 2009. 1523, 1524, 1528

[4] P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: A benchmark. In *CVPR*, 2009. 1523, 1524

[5] L. Duan, I. W.-H. Tsang, D. Xu, and S. J. Maybank. Domain transfer svm for video concept detection. In *CVPR*, 2009. 1524

[6] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010. 1523, 1524

[7] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *CVPR Workshop of Generative Model Based Vision*, 2004. 1523, 1524

[8] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008. 1524, 1528

[9] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical report, California Institute of Technology, 2007. 1528

[10] J. Hutchison. Culture, communication, and an information age madonna. In *IEEE Professional Communication Society Newsletter*, volume 45, 2001. 1523

[11] T. Malisiewicz and A. A. Efros. Recognition by association via learning per-exemplar distances. In *CVPR*, 2008. 1525

[12] S. A. Nene, S. K. Nayar, and H. Murase. Columbia object image library (coil-100). Technical Report CUCS-006-96, Columbia Univ., 1996. 1523

[13] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *International Journal in Computer Vision*, 42:145–175, 2001. 1521, 1523

[14] J. Ponce, T. L. Berg, M. Everingham, D. Forsyth, M. Hebert, S. Lazebnik, M. Marszałek, C. Schmid, C. Russell, A. Torralba, C. Williams, J. Zhang, and A. Zisserman. Dataset issues in object recognition. In *Towards Category-Level Object Recognition*. Springer, 2006. 1522, 1523, 1527

[15] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23–38, January 1998. 1522

[16] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. LabelMe: a database and web-based tool for image annotation. 77(1-3):157–173, 2008. 1523, 1524

[17] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Transferring visual category models to new domains. In *ECCV*, 2010. 1524

[18] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: a large database for non-parametric object and scene recognition. *IEEE PAMI*, 30(11):1958–1970, November 2008. 1521, 1523, 1528

[19] J. Winn, A. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. In *ICCV*, 2005. 1523, 1524

[20] J. Xiao, J. Hays, K. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010. 1523, 1524

[21] J. Yang, R. Yan, and A. G. Hauptmann. Cross-domain video concept detection using adaptive svms. MULTIMEDIA '07, 2007. 1524