From Virtual to Reality: Fast Adaptation of Virtual Object Detectors to Real Domains

Baochen Sun
http://www.cs.uml.edu/~bsun
Kate Saenko
http://www.cs.uml.edu/~saenko

Computer Science Department
University of Massachusetts Lowell
Lowell, Massachusetts, US

Abstract

The most successful 2D object detection methods require a large number of images annotated with object bounding boxes to be collected for training. We present an alternative approach that trains on virtual data rendered from 3D models, avoiding the need for manual labeling. Growing demand for virtual reality applications is quickly bringing about an abundance of available 3D models for a large variety of object categories. While mainstream use of 3D models in vision has focused on predicting the 3D pose of objects, we investigate the use of such freely available 3D models for multicategory 2D object detection. To address the issue of dataset bias that arises from training on virtual data and testing on real images, we propose a simple and fast adaptation approach based on decorrelated features. We also compare two kinds of virtual data, one rendered with real-image textures and one without. Evaluation on a benchmark domain adaptation dataset demonstrates that our method performs comparably to existing methods trained on large-scale real image domains.

1 Introduction

Recent success of multicategory object detection relies heavily on the availability of large amounts of labeled training data. For example, the PASCAL VOC dataset [4] uses a total of over 10,000 labeled images to train detectors for 20 common object categories. Unfortunately, such data collection is extremely time-consuming and does not scale to tens or even hundreds of thousands of object categories and subcategories used by humans. What is worse, the web–the primary source of large-scale training data–is biased in ways that make finding images that match the statistics of realistic test images problematic [20, 24].

In this paper, we propose to bypass expensive 2D image collection and annotation and instead use freely available 3D models of objects to generate training data. Previous work generated synthetic data from 3D models for 2D detection, but has been limited to a few categories [15]. We capitalize on increasing availability of a large variety of 3D object models on websites such as Google 3D Warehouse[1]. The number of models available for some categories is staggering (a search for *table* returns 39,815 results) and projects using them range from kitchen designs to recreations of entire downtowns.

One barrier to tapping into this rich data source is that the vast majority of freely available models are not *photorealistic*, i.e. they generate 2D object images that do not look real,

[1] https://3dwarehouse.sketchup.com

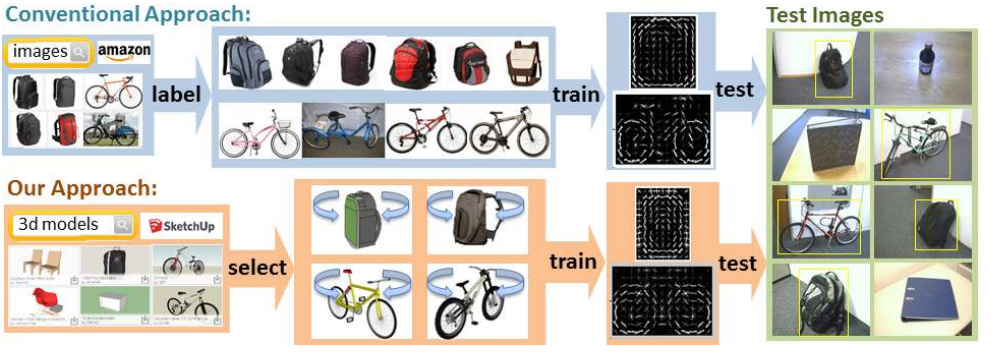

Figure 1: Conventional object detection requires a high-cost manual annotation effort to collect and label a large number of training images. Our approach bypasses labor-intensive labeling and generates models directly from 3D models downloaded from the web. With only 2 models per object category, it performs comparably to the coventional approach when detecting a variety of object categories in real test images.

despite being recognizable. Unrealistic backgrounds, surface textures and lighting all result in image statistics that differ significantly from those of real images. Thus training on virtual images and testing on real images brings us right back to the problem of dataset bias.

[27] addressed the bias issue using supervised domain adaptation techniques. They obtained state-of-the-art pedestrian detection results on the KITTI autonomous driving benchmark [6] by training on auxiliary data generated from a video game. However, their domain adaptation method requires a substantial amount of labeled target domain images, as well as access to a video game engine's 3D model. While some open source games may provide access to their rendering engines, we believe the web to be a much better source or arbitrary category models. Also, the web models are created with accessible tools like Google Sketchup and can be crowdsourced on a large scale.

We present an approach to quickly train and adapt multicategory virtual detectors to real image domains. Perhaps counter-intuitively, we show that photorealistic training data is not required to train virtual models that generalize to real domains. The reason is that modern discriminative detectors based on gradient features throw away the "background statistics" and retain only the average shape and category-specific texture. For example, the backpack model trained on a large number of real images shown in Figure 1 learns the round shape of a backpack but ignores textures specific to individual backpacks.

Hariharan *et al*. [10] showed that such a discriminative model can be learned efficiently by decorrelating the input features, and then simply computing the mean of the positive class. They proposed to compute the local covariance and mean negative class (background) statistics of real images once, and use them for all future detection tasks. We build on this idea, however, we show that its assumption of a single one-fits-all set of real image statistics breaks down in transfer learning scenarios such as virtual to real domain adaptation. The key to our approach is utilizing background statistics that match the source and target domains.

We evaluate our method on a benchmark aimed to simulate home robotics tasks [20]. It contains real images of everyday object categories found in a typical home or office, some of which are shown as the "test images" in Figure 1. Experiments show that our approach performs comparably to training on real-image web domains. Furthermore, when a small number of labeled images is available in the target domain, we propose efficient supervised

adaptation following the method of [7]. In this case, our method using auxiliary virtual data slightly outpeforms the approach using real-image web domains reported in [7] on the benchmark.

To summarize, our paper makes the following major contributions: (1) we show that freely available non-photorealistic 3D models can be used to train 2D object detectors, in a first such study that evaluates across a large variety of categories; (2) we eliminate the need to generate images that match real-image statistics by utilizing domain-specific image statistics; (3) we present a supervised adaptation approach, and show improved results on a multi-domain dataset.

2 Related work

3D Models for Detection. Most previous work utilizing 3D models focuses on predicting the 3D pose or viewpoint of an object in 2D test images [14, 18, 22, 23, 25], or categorizing an object from an arbitrary novel viewpoint [21]. While these are important computer vision tasks, in this paper, we focus solely on 2D object detection, where the goal is to output a 2D bounding box and category label.

Some 3D pose prediction methods have also been applied to 2D detection. For example, [15] show competitive performance with state-of-the-art 2D object detectors on car and motorcycle classes in PASCAL. They extract a set of discriminant features from synthetic 3D object models suitable for matching to real image data and represent them by their appearance and 3D position. However the vast majority of methods use annotated 2D images in some form, either to build 3D models via complex structure-from-motion techniques, or to combine 3D models with natural image appearance [15]. [25] collect a large dataset of real images annotated with 3D pose for 12 rigid categories of PASCAL, and show results of both 2D detection and pose estimation, training multi-view detectors for each distinct object pose on labeled real images. In contrast, we do not require labeled 2D images and train solely on a small number of 3D models (2 per category) downloaded from the web. Our method requires no tedius manual part or bounding box annotation, nor matching of real appearance with virtual models.

The idea of using 3D models of objects as the only source of information was first introduced in the early days of computer vision, see e.g., [17]. [22] go "back to the future" of the field and use 3D CAD models of cars as the only source of labeled data. They detect different viewpoints of cars and show performance comparable to approaches using real images. However, their approach involves 41 extremely detailed CAD models of cars. It is unclear how easily such models can be obtained for arbitrary categories. Also, a major drawback of existing methods is that they have been demonstrated only on a handful of object categories, namely, car [14, 23], bicycle [14] and motorcycle[15]. We show the feasibility of our approach for a varied set of 20 object categories.

Domain Adaptation Domain adaptation is the standard approach to alleviate dataset bias caused by a difference in the statistical distributions between training and test data. In computer vision, several domain adaptation models have been proposed for object categorization. These can be divided into supervised methods that use a small number of labeled examples in the target domain [11, 12, 13], and unsupervised methods that use only unlabeled target examples [8, 9].

For detection tasks, a class of supervised models that adjust the parameters of a classifier trained on source data have been proposed for linear support vector machines [1, 28], as well as for variants of SVMs, e.g., adaptive latent SVM [27] or adaptive exemplar SVM [2].

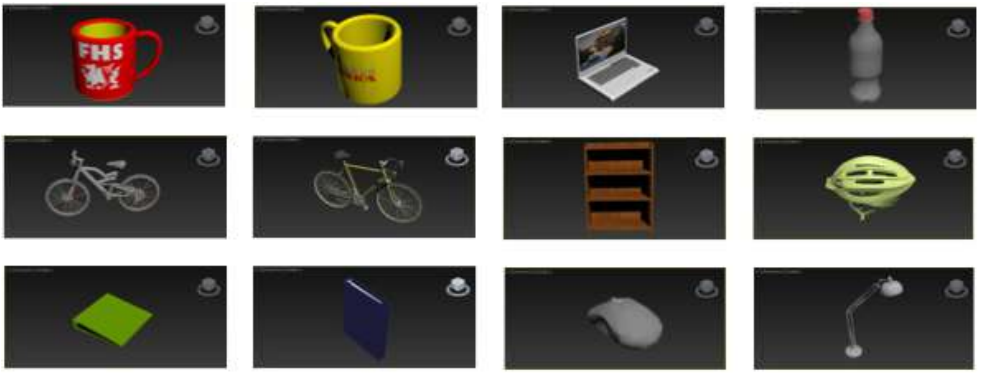

Figure 2: Some of the 3D models used in this paper.

Recently there has been considerable interest in scalable training of many object detectors. The supervised domain adaptation method of [7] proposes a fast adaptation technique that builds on a recent efficient approach to learn detectors based on whitened HOG (WHO) features [10]. It can learn and adapt a model in a few seconds, compared to hours of training an SVM, but, like [10], assumes that the same whitening procedure works for all domains. We argue that this is not the case, and present an improved method that achieves better results. A related method that used exemplar LDA to train pedestrian classifiers and performed supervised adaptation on them using a boosting approach was shown in [26].

Unsupervised adaptation of detectors was explored in [19] for the pedestrian class. To the best of our knowledge, ours is the first unsupervised method applied to a large variety of object categories.

3 Virtual Data

3.1 3D Models

The 3D models used in this paper were downloaded from Google 3D Warehouse by searching for the names of 20 object categories in the Office dataset [20]. The returned models vary considerably, from very simple ones to complicated ones. For each category, two models with similar shapes as the objects in the real world were selected from the first one or two pages of returned results. Some of the selected 3D models are shown in Figure 2.

3.2 2D Data Generation

The 3D models were then rendered in 3ds Max to generate 2D virtual training data. In order to investigate the role of image statistics, two sets of virtual data were generated. The first, referred to as *Virtual*, attempts to create images that better match real image statistics. It does so by using random image selected from ImageNet [3] as background and texturemap on the object. The second, referred to as *Virtual-Gray*, forgoes photorealism altogether and uses a uniform gray texturemap and white background.

For each model, 15 random poses were generated by rotating the original 3D model by a random angle from 0 to 20 degrees in each of the three axes. Since the angle of rotation was generated randomly, the poses are different for each model. In this paper we do not attempt to model multiview components (we leave that to future work), so the goal was to

(a) (b)

Figure 3: Two sets of virtual images were genereated: (a) *Virtual*: background and texturemap from a random real ImageNet image; (b) *Virtual-Gray*: uniform gray texturemap and white background.

generate some small variations in pose and learn a single template from all images. The random background and texture from ImageNet or gray texture and white background was then applied to each pose to get the final 2D virtual data. The default lighting was used. Thus, for each category, there were 30 different rendering settings and 30 corresponding 2D virtual images were generated. The rendering process above was fully automated in MAXScript.

To get the annotations, a parallel set of images was also generated which share the same rendering setting as the virtual image generated above except that the background is always white. These images were only used in automatically calculating the bounding box for the corresponding virtual training images, by calculating the largest bounding box of non-white pixels. Note that full segmentation masks could also be easily obtained.

4 Adapting from Virtual to Real Domains

4.1 Discriminative Decorrelation: Background

We begin by describing the decorrelation-based approach to detection proposed in [10]. Given an image I, it follows the sliding-window paradigm, extracting a d-dimensional feature vector $\phi(I,b)$ at each window b across all locations and at multiple scales. It then scores the windows using a scoring function

$$f_{\mathbf{w}}(I,b) = \mathbf{w}^T \phi(I,b). \tag{1}$$

In practice, all windows with values of $f_{\mathbf{w}}$ above a predetermined threshold are considered positive detections.

In recent years, use of the linear SVM as the scoring function $f_{\mathbf{w}}$, usually with Histogram of Gradients (HOG) as the features ϕ, has emerged as the predominant object detection paradigm. Yet, as observed by Hariharan *et al.* [10], training SVMs can be expensive, especially because it usually involves costly rounds of hard negative mining. Furthermore, the training must be repeated for each object category, which makes it scale poorly with the number of categories.

Hariharan et al. proposed a much more efficient alternative, learning $f_{\mathbf{w}}$ with Linear Discriminant Analysis (LDA). LDA is a well-known linear classifier that models the training set of examples $\mathbf{x}$ with labels $y \in \{0,1\}$ as being generated by $p(\mathbf{x},y) = p(\mathbf{x}|y)p(y)$. $p(y)$ is

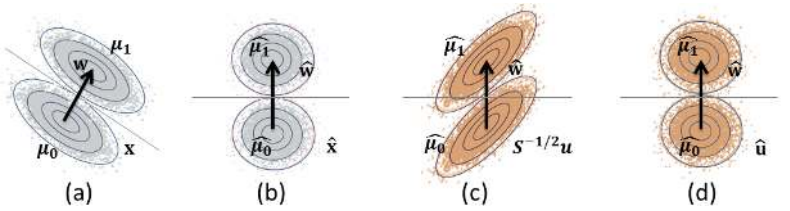

Figure 4: (a) Applying a linear classifier $\mathbf{w}$ learned by LDA to source data $\mathbf{x}$ is equivalent to (b) applying classifier $\hat{\mathbf{w}} = \mathbf{S}^{-1/2}\mathbf{w}$ to decorrelated points $\mathbf{S}^{-1/2}\mathbf{x}$. (c) However, target points $\mathbf{u}$ may still be correlated after $\mathbf{S}^{-1/2}\mathbf{u}$, hurting performance. (d) Our method uses target-specific covariance to obtain properly decorrelated $\hat{\mathbf{u}}$.

the prior on class labels and the class-conditional densities are normal distributions

$$p(\mathbf{x}|y) = N(\mathbf{x}; \mu^y, \mathbf{S}), \tag{2}$$

where the feature vector covariance $\mathbf{S}$ is assumed to be the same for both positive and negative (background) classes. In our case, the feature is represented by $\mathbf{x} = \phi(I, b)$. The resulting classifier is given by

$$\mathbf{w} = \mathbf{S}^{-1}(\mu_\mathbf{1} - \mu_\mathbf{0}) \tag{3}$$

The innovation in [10] was to re-use $\mathbf{S}$ and $\mu_\mathbf{0}$, the background mean, for all categories, reducing the task of learning a new category model to computing the average positive feature, $\mu_\mathbf{1}$. This was accomplished by calculating $\mathbf{S}$ and $\mu_\mathbf{0}$ for the largest possible window and subsampling to estimate all other smaller window sizes. Also, $\mathbf{S}$ was shown to have a sparse local structure, with correlation falling off sharply beyond a few nearby image locations.

Like other classifiers, LDA learns to suppress non-discriminative structures and enhance the contours of the object. However it does so by learning the global covariance statistics once for all natural images, and then using the inverse covariance matrix to remove the non-discriminative correlations, and the negative mean to remove the average feature. LDA was shown in [10] to have competitive performance to SVM, and can be implemented both as an exemplar-based [16] or as deformable parts model (DPM) [5].

4.2 Discriminative Decorrelation for Unsupervised Adaptation

We observe that estimating global statistics $\mathbf{S}$ and $\mu_\mathbf{0}$ once and re-using them for all tasks may work when training and testing in the same domain, but in our case, the virtual training data is likely to have different statistics from the target real data. Figure 5 illustrates the effect of centering and decorrelating a positive mean using global statistics from the wrong domain. The effect is clear: important discriminative information is removed while irrelevant structures are not.

Based on this observation, we propose an adaptive decorrelation approach to detection. Assume that we are given labeled training data $\{\mathbf{x}, y\}$ in the source domain (*e.g.* virtual images rendered from 3D models), and unlabeled examples $\mathbf{u}$ in the target domain (*e.g.* real images collected in an office environment). Evaluating the scoring function $f_\mathbf{w}(\mathbf{x})$ in the source domain is equivalent to first decorrelating the training features $\hat{\mathbf{x}} = \mathbf{S}^{-1/2}\mathbf{x}$, computing their positive and negative class means $\hat{\mu_\mathbf{1}} = \mathbf{S}^{-1/2}\mu_\mathbf{1}$ and $\hat{\mu_\mathbf{0}} = \mathbf{S}^{-1/2}\mu_\mathbf{0}$ and then projecting the decorrelated feature onto the decorrelated difference between means, $f_\mathbf{w}(\mathbf{x}) = \hat{\mathbf{w}}^T\hat{\mathbf{x}}$, where $\hat{\mathbf{w}} = (\hat{\mu_\mathbf{1}} - \hat{\mu_\mathbf{0}})$. This is illustrated in Figure 4(a-b).

However, as we saw in Figure 5, the assumption that the input is properly decorrelated does not hold if the input comes from a target domain with a different covariance structure. Figure 4(c) illustrates this case, showing that $\mathbf{S}^{-1/2}\mathbf{u}$ does not have isotropic covariance. Therefore, $\mathbf{w}$ cannot be used directly.

We may be able to compute the covariance of the target domain on the unlabeled target points $\mathbf{u}$, but not the positive class mean. Therefore, we would like to re-use the decorrelated mean difference $\hat{\mathbf{w}}$, but adapt to the covariance of the target domain. In the rest of the paper, we make the assumption that the difference between positive and negative means is the same in the source and target. This may or may not hold in practice, and we discuss this further in Section 5.

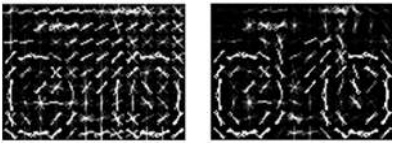

Figure 5: Mean bicycle decorrelated with mismatched-domain covariance (left) vs. with same-domain covariance (right).

Let the estimated target covariance be $\mathbf{T}$. We first decorrelate the target input feature with its inverse square root, and then apply $\hat{\mathbf{w}}$ directly, as shown in Figure 4(d). The resulting scoring function is

$$f_{\hat{\mathbf{w}}}(\mathbf{u}) = \hat{\mathbf{w}}^T\hat{\mathbf{u}} \tag{4}$$

$$= (\mathbf{S}^{-1/2}(\mu_1 - \mu_0))^T(\mathbf{T}^{-1/2}\mathbf{u}) \tag{5}$$

$$= ((\mathbf{T}^{-1/2})^T\mathbf{S}^{-1/2}(\mu_1 - \mu_0))^T\mathbf{u} \tag{6}$$

This corresponds to a transformation $(\mathbf{T}^{-1/2})^T(\mathbf{S}^{-1/2})$ instead of the original whitening $\mathbf{S}^{-1}$ being applied to the difference between means to compute $\mathbf{w}$. Note that if source and target domains are the same, then $(\mathbf{T}^{-1/2})^T(\mathbf{S}^{-1/2})$ equals to $\mathbf{S}^{-1}$ since $\mathbf{S}$ is positive definite.

In practice, either the source or the target component of the above transformation may also work, or even statitstics from similar domains. However, as we will see in Section 5, dissimilar domain statistics can significantly hurt performance. Furthermore, if either source or target has only images of the positive category available, and cannot be used to properly compute background statistics, the other domain can still be used.

4.3 Supervised Adaptation

We extend our approach to supervised adaptation when a few labeled examples are available in the target domain. Following [7], a simple adaptation method is used whereby the template learned on source positives is combined with a template learned on target positives, using a weighted linear combination. The key difference with our approach is that the target template uses target-specific statistics.

In [7], the author uses the same background statistics as [10] which were estimated on 10,000 natural images from the PASCAL VOC 2010 dataset. Based on our analysis above, even though these background statistics were estimated from a very large amount of real image data, it will not work for all domains. In section 5, our results confirms this claim.

5 Evaluation

Datasets and Setup. We evaluate our method on the Office dataset [20], a standard benchmark for domain adaptation. We use the same setting as [7], performing detection on the *Webcam* domain as the target (test) domain, and evaluating on same 783 image test set of 20 categories (out of 31). As source (training) domains, we use: our *Virtual* and *Virtual-Gray*

	Virtual	Virtual-Gray	Amazon	DSLR	PASCAL
Virtual	30.8 (*0.1*)	16.5 (*1.0*)	24.1 (*0.6*)	28.3 (*0.2*)	10.7 (*0.5*)
Virtual-Gray	32.3 (*0.6*)	32.3 (*0.5*)	27.3 (*0.8*)	32.7 (*0.6*)	17.9 (*0.7*)
Amazon	39.9 (*0.4*)	30.0 (*1.0*)	39.2 (*0.4*)	37.9 (*0.4*)	18.6 (*0.6*)
DSLR	68.2 (*0.2*)	62.1 (*1.0*)	68.1 (*0.6*)	66.5 (*0.1*)	37.7 (*0.5*)

Table 2: MAP of detectors trained on positive examples from each row's source domain and background statistics from each column's domain. The average distance between each set of background statistics(each column) to the true source(each row) and target(webcam) statistics is shown in parentheses.

data, and the two remaining real-image domains in Office, *Amazon* and *DSLR*. Examples of *Amazon* and *Webcam* are shown in Figure 1. We also compare to [7] who use corresponding ImageNet[3] synsets as the source. Thus, there are four potential source domains (two synthetic and two real) and one (real) target domain. The number of positive training images per category in each domain is shown in Table 1.

Domain	No. Train
Amazon	20
DSLR	8
Virtual(-Gray)	30
ImageNet	150-2000

Table 1: Source domains.

Effect of Mismatched Image Statistics. First, we explore the effect of mismatched precomputed image statistics on detection performance. For each source domain, we train LDA detectors using the positive mean from the source, and pair it with the covariance and negative mean of other domains. The virtual and the Office domains are used as sources, and the test domain is always Webcam. The statistics for each of the four domains were calculated using all of the training data, following the same approach as [10]. The pre-computed statistics of 10,000 real images from PASCAL, as proposed in [7, 10], are also evaluated.

Detection performance, measured in Mean Average Precision (MAP), is shown in Table 2. We also calculate the normalized Euclidean distance between pairs of domains as $(\|\mathbf{S^1} - \mathbf{S^2}\|)/(\|\mathbf{S^1}\| + \|\mathbf{S^2}\|) + (\|\mu_0^1 - \mu_0^2\|)/(\|\mu_0^1\| + \|\mu_0^2\|)$, and show the average distance to source and target in parentheses in Table 2. From these results we can see a trend that larger domain difference leads to poorer performance. Note that larger difference to the target domain also leads to lower performance, confirming our hypothesis that both source and target statistics matter. Some of the variation could also stem from our assumption about the difference of means being the same not quite holding true. Finally, the PASCAL statistics from [10] perform the worst. Thus, in practice, statistics from either source domain or target domain or domains close to them could be used. However, unrelated statistics will not work even thought they might be estimated from a very large amount of data as [10].

Unsupervised and Supervised Adaptation. Next, we report the results of our unsupervised and supervised adaptation technique. We use the same setting as [7], in which three positive and nine negative labelled images per category were used for supervised adaptation. Target covariance in Equation 4 is estimated from 305 unlabeled training examples. We also followed the same approach to learn a linear combination between the unsupervised and supervised model via cross-validation. The results are presented in Table 3. Please note that our target-only MAP is 52.9 compared to 36.6 in [7]. This also confirms our conclusion that the statistics should come from a related domain. It is clear that both of our unsupervised and supervised adaptation techniques outperform the method in [7]. Furthermore, *Virtual-Gray* data outperforms *Virtual*, and *DSLR* does best, as it is very close to the target domain (the main difference is in the camera used to capture images).

Source	Source-only [10]	UnsupAdapt-Ours	SupAdapt [7]	SupAdapt-Ours
Virtual	10.7	27.9	30.7	45.2
Virtual-Gray	17.9	33.0	35.0	54.7
Amazon	18.6	38.9	35.8	53.0
DSLR	37.7	67.1	42.9	71.4

Table 3: Top: Comparison of the source-only [10] and supervised-adapted model of [7] with our unsupervised-adapted and supervised adapted models. Mean AP across categories is reported on the webcam test data, using different source domains for training. Bottom: Sample detections of the DSLR-UnsupAdapt-Ours detectors.

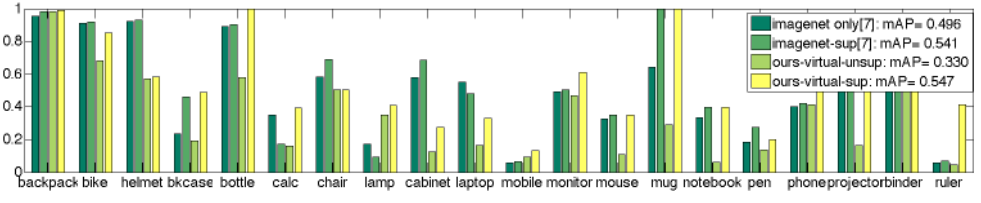

Figure 6: Comparison of unsupervised and supervised adaptation of virtual detectors using our method with the results of training on ImageNet and supervised adaptation from Imagenet reported in [7]. Our supervised-adapted detectors achieve comparable performance despite not using any real source training data, and using only 3 positive images for adaptation, and even outperform ImageNet significanly for several categories (c.f. *ruler*).

Finally, we compare our method trained on *Virtual-Gray* to the results of adapting from ImageNet reported by [7], in Figure 6. While their unsupervised models are learned from 150-2000 real ImageNet images per category and the background statistics are estimated from 10,000 PASCAL images, we only have 30 virtual images per category and the background statistics is learned from about 1,000 images. What's more, all the virtual images used are with uniform gray texturemap and white background. This clearly demonstrates the importance of domain-specific decorrelation, and shows that the there is no need to collect a large amount of real images to train a good classifier.

6 Conclusion

This paper demonstrates that virtual data rendered from freely available 3D models could be a promising new way to train object detectors on a large scale. In our experiments, detectors trained on virtual data and adapted to real-image statistics perform comparably to detectors trained on real image datasets, including ImageNet. Interestingly, our results showed that non-photorealistic data works just as well as attempts to render more realistic images. The objects in our evaluation were mostly rigid man-made objects; in future work we plan to include more non-rigid objects and more categories.

7 Acknowledgments

The authors would like to thank Judy Hoffman and Bharath Hariharan for sharing their code and answering so many questions, Karim Ali for reading the draft, and the anonymous reviewers for their valuable comments and suggestions. This research was supported by NSF award 1212928 and by DARPA.

References

[1] Y. Aytar and A. Zisserman. Tabula rasa: Model transfer for object category detection. In *IEEE International Conference on Computer Vision*, 2011.

[2] Y. Aytar and A. Zisserman. Enhancing exemplar svms using part level transfer regularization. In *British Machine Vision Conference*, 2012.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[4] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[5] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645, 2010.

[6] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[7] Daniel Goehring, Judy Hoffman, Erik Rodner, Kate Saenko, and Trevor Darrell. Interactive adaptation of real-time object detectors. In *International Conference on Robotics and Automation (ICRA)*, 2014.

[8] B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *Proc. CVPR*, 2012.

[9] R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *Proc. ICCV*, 2011.

[10] Bharath Hariharan, Jitendra Malik, and Deva Ramanan. Discriminative decorrelation for clustering and classification. In *Computer Vision–ECCV 2012*, pages 459–472. Springer, 2012.

[11] Judy Hoffman, Erik Rodner, Jeff Donahue, Kate Saenko, and Trevor Darrell. Efficient learning of domain-invariant image representations. In *International Conference on Representation Learning, arXiv:1301.3224*, 2013.

[12] A. Khosla, T. Zhou, T. Malisiewicz, A. Efros, and A. Torralba. Undoing the damage of dataset bias. In *Proceedings of the 12th European conference on Computer Vision*, 2012.

[13] Brian Kulis, Kate Saenko, and Trevor Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1785–1792. IEEE, 2011.

[14] J. Liebelt and C. Schmid. Multi-view object class detection with a 3d geometric model. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010.

[15] J. Liebelt, C. Schmid, and Klaus Schertler. Viewpoint-independent object class detection using 3d feature maps. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, 2008.

[16] Tomasz Malisiewicz, Abhinav Gupta, and Alexei A Efros. Ensemble of exemplar-svms for object detection and beyond. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 89–96. IEEE, 2011.

[17] Ramakant Nevatia and Thomas O. Binford. Description and recognition of curved objects. *Artificial Intelligence*, 8(1):77 – 98, 1977.

[18] B. Pepik, M. Stark, P. Gehler, and B. Schiele. Teaching 3d geometry to deformable part models. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012.

[19] Daniel Ponsa, David Geronimo, Antonio M. Lopez, Javier Marin, and David Vazquez. Virtual and real world adaptationfor pedestrian detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(4):797–809, 2014. ISSN 0162-8828.

[20] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *Computer Vision–ECCV 2010*, pages 213–226. Springer, 2010.

[21] Silvio Savarese and Li Fei-Fei. View synthesis for recognizing unseen poses of object classes. In David Forsyth, Philip Torr, and Andrew Zisserman, editors, *Computer Vision Ű ECCV 2008*, volume 5304 of *Lecture Notes in Computer Science*, pages 602–615. Springer Berlin Heidelberg, 2008.

[22] Michael Stark, Michael Goesele, and Bernt Schiele. Back to the future: Learning shape models from 3d cad data. In *Proc. BMVC*, pages 106.1–11, 2010. ISBN 1-901725-40-5. doi:10.5244/C.24.106.

[23] Min Sun, Hao Su, S. Savarese, and Li Fei-Fei. A multi-view probabilistic model for 3d object classes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 2009.

[24] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1521–1528. IEEE, 2011.

[25] Yu Xiang, Roozbeh Mottaghi, and Silvio Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2014.

[26] Jiaolong Xu, D. Vazquez, S. Ramos, A.M. Lopez, and D. Ponsa. Adapting a pedestrian detector by boosting lda exemplar classifiers. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on*, June 2013.

[27] Jiaolong Xu, Sebastian Ramos, David Vazquez, and Antonio Lopez. Domain adaptation of deformable part-based models. *in submission*, 2014. URL http://refbase.cvc.uab.es/files/xrv2013.pdf.

[28] J. Yang, R. Yan, and A. G. Hauptmann. Cross-domain video concept detection using adaptive svms. *ACM Multimedia*, 2007.