Building the View Graph of a Category by Exploiting Image Realism

Attila Szabó
University of Bern
Switzerland
szabo@inf.unibe.ch

Andrea Vedaldi
University of Oxford
United Kingdom
vedaldi@robots.ox.ac.uk

Paolo Favaro
University of Bern
Switzerland
paolo.favaro@inf.unibe.ch

Abstract

We propose a weakly supervised method to arrange images of a given category based on the relative pose between the camera and the object in the scene. Relative poses are points on a sphere centered at the object in a given canonical pose, which we call object viewpoints. Our method builds a graph on this sphere by assigning images with similar viewpoint to the same node and by connecting nodes if they are related by a small rotation. The key idea is to exploit a large unlabeled dataset to validate the likelihood of dominant 3D planes of the object geometry. A number of 3D plane hypotheses are evaluated by applying small 3D rotations to each hypothesis and by measuring how well the deformed images match other images in the dataset. Correct hypotheses will result in deformed images that correspond to plausible views of the object, and thus will likely match well other images in the same category. The identified 3D planes are then used to compute affinities between images related by a change of viewpoint. We then use the affinities to build a view graph via a greedy method and the maximum spanning tree.

1. Introduction

Image understanding is progressing at a rapid pace. In particular, with the introduction of the latest generation of deep Convolutional Neural Networks (CNN), problems such as object category classification, segmentation, and detection have progressed tremendously. However, these problems are still addressed by means of view-based models, reducing object understanding to matching 2D patterns. However, understanding the 3D nature of objects is essential in many advanced applications of vision which require a detailed understanding of the physical space, including navigation, manipulation, and understanding of activities. This explains the growing attention of the community to the problem of modeling 3D object categories [36, 26, 39, 21, 2, 12, 32, 23]. However, the problem remains largely open.

In this work, we propose new and effective tools to progress in the understanding of object categories in 3D. Rather than committing to a specific model of 3D object categories, we focus on the problem of *identifying the viewpoint of objects* in very large unlabelled image collections. The viewpoint of an object is an important attribute that, when known, could make the classification task much simpler, more reliable and more efficient by simplifying the task of learning 3D-aware object models. The problem of determining the viewpoint of objects requires establishing a relationship between different instances (viewpoints and identity) of the same object. This task boils down to finding correspondences and these can be very challenging with typical intraclass variations. Moreover, due to occlusions and the 3D shape of the object, correspondences between far viewpoints may be very unreliable. We argue that the dramatic change in appearance due to viewpoint changes is best handled by using large datasets, so that there exist smooth viewpoint transitions between pairs of images and there are many examples with small intraclass variability. While large collections of images can be easily obtained through Flickr, all of these images are unlabeled, so that an unsupervised approach is required.

Once labelled with viewpoint, the resulting images can support learning deep CNN models, which usually require large supervised datasets for their training. Unfortunately, current algorithms to estimate the viewpoint of object categories are too slow and/or require some form of supervision (section 2), and therefore do not satisfy our requirements of operating on *very large unlabelled image collections*.

Here we propose a simple but efficient algorithm for labelling large image collections with their 3D viewpoint. Each image is associated to a small number of 3D planes (shape hypotheses) approximating the shape of the object contained in it. Each shape hypothesis allows to synthesize out-of-plane object rotations as image homographies. Provided that the rotation is small and that the selected shape hypothesis is correct, the synthesized image often looks realistic; otherwise, this procedure usually results in noticeable distortions of the 3D object. To measure the degree of realism of each hypothesis, the synthesized images are di-

rectly compared to other images in the collection and their pairwise similarity is computed. Since we expect all viewpoints to be covered densely, we can use the degree of similarity of the best matches as a metric for the image realism, which translates into a likelihood for the shape hypothesis. The identified shape is then used to establish approximate correspondences between images. To find if two images are related by a small rotation, one can use the estimated plane to synthesize another view of one image and then compute the similarity (affinity) between the other image and the synthesized one. These affinities can then be used to build the view graph by using a greedy approach or a maximum spanning tree. The algorithm is described in full detail in section 3.

2. Related work

Current methods for viewpoint estimation of object categories can be divided into three groups: supervised, semi/weakly-supervised and unsupervised. As discussed next, the vast majority of algorithms are supervised. We conclude the section with an overview of several techniques used in our method.

2.1. Supervised methods

Supervised methods learn to recognize the viewpoint of a 3D object category from example images that are annotated with that information. A simple method to do so is to train a mixture of 2D object detectors, from samples of similar viewpoints. Xiang *et al*. [36] does this by using standard deformable parts models (DPM) [11], where each mixture component corresponds to a different viewpoint. The authors of [26, 39] extend DPMs to model 3D categories directly and use synthetic data for training. Training is simplified by leveraging high quality renderings of CAD models and the availability of virtually exact viewpoint information for each generated image. 3D DPMs also enable continuos pose estimation. A recent work [6] also exploits 3D CAD models, but with a method for rendering and training exemplar detectors at run time. A less structured approach is the one of [21] that uses Exemplar SVMs to transfer viewpoint and other attributes from training images to test images. Aubry *et al*. [2] uses instead rendered CAD models to train a large collection of exemplars of viewpoint-sensitive DPM detectors. At test time, the viewpoint of an object is transferred from the best-matching DPM detector. Exemplar-based methods and 2D and 3D DPMs result in similar viewpoint-estimation performance. The main disadvantage of the exemplar classifiers is their high computational cost at test time. Xiang *et al*. [37] learns shape from CAD models as a set of planes. They find the parts on rectified images and estimate the viewpoint based on the image transformation used for rectification.

Viewpoint estimation can also be performed separately from object detection in a two-stages pipeline. Ghodrati *et al*. [12] use standard DPMs for object detection and estimate the pose from features computed in the detected bounding box. They show that modern CNN features [8] or encodings [27] provide state-of-the-art performance for this problem. Recently CNNs [17, 9] had great success in classification and detection tasks. They turned out to be very useful in pose estimation too. Tulsiani *et al*. [33] train viewpoint estimation separately from the object class detection. CNNs learn shared representation for the image categories and the viewpoints, which makes them scalable for many categories. They also provide the best performance on standard 3D benchmark datasets [36].

2.2. Semi-supervised methods

Work on semi-supervised learning of 3D viewpoint is relatively limited. Tulsiani *et al*. [32] show that a CNN trained using strong supervision is able to infer the pose of unlabelled categories, provided the training set contained a similar category. They also show that the viewpoint estimation quality can be improved by jointly labelling a large image collections of the target category.

2.3. Unsupervised methods

In the paper of Liang *et al*. [23] the pose of objects is learned from short video sequences using statistical manifold learning techniques. To the best of our knowledge their method is the only one that does not require any viewpoint annotations. However, we are interested in learning the viewpoints only from images, as images are more abundantly available than videos.

2.4. Other related work

Most work is based on a common set of algorithms or representations, so that it is useful to present prior work focused on specific choices. One popular choice is to build a graph connecting images based on their viewpoint, as we do in this paper. Another popular choice is to establish part correspondences between image pairs and to analyze the transformation between images via techniques reminiscent of structure from motion.

View graph. Many 3D estimation techniques such as structure from motion start by comparing pairs of images. When images are drawn from a very large dataset, however, considering all possible pairs is infeasible. Thus, in large scale structure from motion images are only compared to their nearest neighbors [1], discovered via a fast bag of words technique [7] using vocabulary trees [25]. Cho *et al*. [5] use a large image set for object discovery; they start by proposing pairs of nearest neighbor images that are likely to contain the same class. They use GIST features [31] for this task. Grauman *et al*. [13] use spectral clustering and

normalized cuts [28] to group objects from the same class in an unsupervised way. To avoid the comparison between all image pairs, they estimate the affinity matrix by using the Nyström method [35]. Their affinity however does not involve any similarity based on viewpoint. In our paper we use nearest neighbor image retrieval to group similar viewpoints together and to reduce the number of image-pairs that need to be evaluated.

Part correspondences. Establishing point correspondences in image pairs is often the first step of structure from motion methods. Such correspondences are often obtained by using local feature detectors and descriptors like SIFT [20]. ASIFT [24] improves on SIFT by simulating a number of affine deformation of the features allowing to establish correspondences between wider baselines. Optical flow [16, 3] provides dense correspondences between frames of a video. Its main advantage is speed, which enables real-time processing of videos.

When the images show two different instances of objects from the same category matching is much more difficult. An approach is to learn the appearance of parts and use the part detections to establish correspondences [33, 19]. Such methods often rely on extensive annotation of keypoints in images; if these are not available, methods such as DPMs [11] can be used to learn parts in a weakly-supervised manner. [33] relies instead on part detectors implicitly learned by deep CNNs. SIFT Flow [18] combines rich feature descriptors and optical flow methods and provides correspondences between semantically related images without any supervised training. FlowWeb [38] provides correspondences between images in a large collection by improving the initial flow field with cycle consistency. In our work we obtain global correspondence between 2 images by using a finite set of simple geometric primitives (3D planes). This choice is dictated by the need to deal with a high intraclass variability and the lack of labeling.

Structure from Motion. Standard structure from motion can recover the shape of a scene and the motion between cameras from images of a single object *instance* [30, 15]. In the case of categories, structure from motion needs to deal with intra-class variance, *i.e.*, when images contain different instances of the same object category. Vicente *et al*. [34] reconstruct the objects in the PASCAL VOC [10] dataset starting from a small set of key-point annotations and ground truth segmentation of objects. Carreira *et al*. [4] reconstruct an object from a single viewpoint using images of the same class to establish correspondences between the test image and the dataset images and use SFM to recover the 3D. Both methods assume orthographic cameras and use the rigid shape model of Marques *et al*. [22]. They show that this simple model is robust against intra-class shape variations. Both methods rely on manual labeling of keypoint correspondences.

3. Construction of the View Graph

In this section we describe our method step by step. The core technique consists in postulating a number of 3D geometry hypotheses for the object in the scene and then to validate these hypotheses by rotating the object according to the given geometry. When the correct hypothesis is made, the deformation will be realistic and thus match well several images in the complete dataset. However, these calculations might be overkill if applied to all the images. Moreover, many images in the dataset should be discarded as the object might appear too occluded or too small. Therefore, we devise a procedure to select a subset of the complete dataset suitable to build the view graph with a feasible computational effort.

3.1. Global Feature Analysis

One key component of our method is the ability to determine if two images depict an object from the same category and with the same viewpoint. We follow the common practice of performing clustering via global features [5], extracting them from every image, and using them to compare all image pairs.

By following prior work on supervised learning for viewpoint estimation [12], we restrict the options for global features to GIST [31], VGG CNNs [29] and BoW Fisher encodings [27]. In order to compare these alternatives, we perform several tests on the Pascal 3D dataset [36, 10]. Because this dataset contains viewpoint annotations, it can be used to verify the agreement between the viewpoint of different images. We use a discrete set (1, 4 and 24) of viewpoints, uniformly placed along the view circle. Each image is then used to recall the top 20 nearest neighbors using one of the global features. A retrieved result is considered correct if it is of the same class of the query and if it has the same discretized viewpoint. Both the cases in which the database contains background images or not are considered. The results are shown in Fig. 1 and clearly show that VGG CNN is the preferred representation.

3.2. Preprocessing

In order to limit the number of calculations later on, we select a subset of the 100,000 images which will contribute to building the view graph. We calculate all the pairwise affinities between the images based on the inner product of their global features. We denote with I_i the i-th image in the dataset and with ϕ the VGG CNN feature mapping including the L_2 normalization, so that

$$\phi(I) = \frac{\text{VGG CNN}(I)}{\|\text{VGG CNN}(I)\|_2}. \quad (1)$$

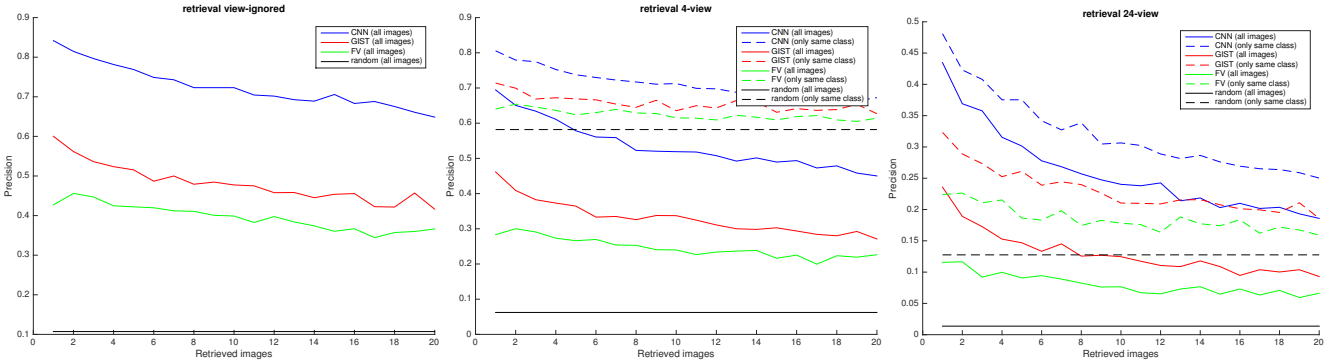

Figure 1. Evaluation of global features for viewpoint classification on the Pascal 3D dataset [36, 10]. The evaluation compares VGG CNNs [29], GIST [31] and the BoW Fisher encodings [27]. From the left to the right: Evaluation of class detection (any view), evaluation of viewpoint estimation with 4 views, and evaluation of viewpoint estimation with 24 viewpoints. As can be seen, the VGG CNNs are consistently better than the other 2 global features (the same ranking applies to the evaluation with 8 and 16 viewpoints).

For every image I_i we select its top 20 matches, and sum up their matching scores, resulting in m_i. We rank the images according to their m_i scores. We discard the top 1000 to eliminate near duplicates. We choose images randomly between the images of rank 1001 and 10001. From this process we obtain a set S of 1000 images per category. The selected images are the representative images of the full dataset. Each image corresponds to a node in the view graph and represents a (not necessarily unique) viewpoint. The other images in the dataset can be assigned to their nearest neighbor in S. In this paper we focus on the relative viewpoint difference between the nodes. The global viewpoint assignment is still an open problem.

3.3. 3D Geometry Identification

As mentioned in the introduction, we formulate 3D geometry hypotheses for a given object. The objective is to fit 3D primitives to the objects in the scene. Several prior works have looked at the same problem (see, for instance [14] and references therein), but always with a supervised learning approach. Our approach is instead radically different as we illustrate here below.

To limit the computational complexity we approximate objects with 3D planes. Then, changes of viewpoint of these planes result in a homography transformation (see Fig. 2). In our experiments we make a total of 81 such hypotheses, which are all combinations of 9 changes of orientation of the plane along the horizontal axis with 9 changes of orientation of the plane along the vertical axis. We denote each hypothesis with θ.

We validate a hypothesis θ by transforming each image in S via a rotation to the left and a rotation to the right with 30 degrees (we do not consider other changes of viewpoint because of limited viewpoint coverage in the dataset). The transformed image $J_i^{\theta,\text{left}}$ then denotes the i-th image I_i rotated to the left under the hypothesis of a 3D plane θ. The image $J_i^{\theta,\text{right}}$ is defined in an analogous manner.

The rotated views are matched against all images in the complete dataset via the following inner product

$$s_{i,j}^{\theta,\text{left/right}} \doteq \langle \phi(J_i^{\theta,\text{left/right}}), \phi(I_j) \rangle. \tag{2}$$

In Fig. 3 we show matches obtained for the 3 manually identified plane hypotheses in Fig. 2. Notice how the proposed score finds suitable matches (the matches are sorted from the best to the worst and the illustration shows the top 13 matches).

These scores are then sorted from the highest to the smallest, and the sorted list is denoted with $\mathcal{K}$. A *realism* score ρ_i^θ is assigned to each plane hypothesis θ for the i-th image by adding the top 20 matches in the dataset, *i.e.*,

$$\rho_i^\theta = \sum_{\text{dir} \in \text{left,right}} \sum_{j=1}^{20} s_{i,\mathcal{K}_j}^{\theta,\text{dir}} \tag{3}$$

i.e., by adding the top 20 best matches for both left and right rotations in the dataset. In Fig. 4 we show the resulting best matches for our 81 hypotheses. Finally, we denote the selected plane hypothesis of the image I_i with θ_i^*

$$\theta_i^* = \arg\max_\theta \rho_i^\theta. \tag{4}$$

3.4. Extracting the View Graph

Once a plane has been assigned to all 1000 images, we can define the affinities between pairs of images that are related by a rotation to the left, to the right or no rotation. Thus, we compute all pairwise affinities $A_{i,j}$ between image I_i and image I_j as

$$A_{i,j} = \max_{\text{dir} \in \{\text{left,none,right}\}} s_{i,j}^{\theta_i^*,\text{dir}} \tag{5}$$

Figure 2. In the diagonal we show the original images (to make all images of the same size smoothed padding has been added at the boundaries). In each column all images are aligned to the same viewpoint. In the case of a bicycle a plane is sufficient to model most viewpoints. When the correct plane hypothesis is used, other viewpoints can be realistically synthesized even though the initial viewpoint was very different.

and also store the corresponding rotation

$$\psi_{i,j} = \arg \max_{\text{dir} \in \{\text{left,right}\}} s_{i,j}^{\theta_i^*,\text{dir}}. \quad (6)$$

We then consider two ways to build the graph: a greedy approach and via the maximum spanning tree. Let us choose as initial node the image I_i. To build the view graph from the initial node, the greedy approach simply picks the left node as the image I_{j^*} where j^* has the highest affinity A_{i,j^*}, the rotation ψ_{i,j^*} is a left rotation and $\psi_{j^*,i}$ is a right rotation. We pick the right node similarly but with reversed direction for ψ_{i,j^*} and $\psi_{j^*,i}$. The same procedure is then repeated on the two new nodes (only one direction per node).

The second approach instead computes the maximum weight spanning tree on the graph, where the nodes are the images and the edge weights are the affinities $A_{i,j}$. Then, we calculate the shortest path between all pairs of images using the tree. This way we can connect the images through a path, where the neighboring images are likely to have the same or similar viewpoints. We find sections of these paths where the object always turns in the same direction. That is, in any of these paths the sequence indices $\mathcal{P}$ will satisfy $\psi_{\mathcal{P}_i,\mathcal{P}_{i+1}} = \text{left}$ for $i = 1,\ldots,|\mathcal{P}|$ or $\psi_{\mathcal{P}_i,\mathcal{P}_{i+1}} = \text{right}$ for $i = 1,\ldots,|\mathcal{P}|$. We visualize the obtained paths with both methods in Fig. 5.

4. Visualization and Evaluation of the View Graphs

We test our method by building datasets for 5 categories: airplane, bicycle, chair, bus and car. For each category we collect 100,000 images via Flickr just by simple index search. Most images match the correct category, but some images might be completely incorrect or repeated multiple times. The selection strategy presented in subsection 3.2 takes care of these cases.

A view graph is a collection of paths obtained by the greedy or the spanning tree method. In Fig. 5 we show results on the airplane, bicycle, chair, bus and car categories obtained from both approaches. In the case of the maximum spanning tree we show one among the longest paths that have a consistent direction ψ (left or right turn). The paths might be shorter than 10 images depending on whether the algorithm is able to find good matches between image pairs or not. The greedy approach tends to have longer paths, but with smoother rotations. Notice that the paths show mostly consistent viewpoint changes and in cases where the available images densely sample the viewpoint space, such as with the airplane category, the ordering can include drastic changes of viewpoint.

Finally, we estimate the quality of the extracted view graphs quantitatively. Because the view graph does not contain the exact viewpoint changes between the images, we

Figure 3. Evaluation of image matches with 3 manually chosen planes. The first column shows the overlap between the identity homography and the homography needed to transform the original image into one of the other 2 examples. Second column: the transformed originals. Columns 3 to 16 are the top matches with decreasing score. Notice how the correct 3D model leads to accurate matches in the complete dataset. The first three rows can be easily interpreted, as the object in the original image (first row) lies on a fronto-parallel plane, which matches exactly the identity homography (blue grid). The rotations in the second and third rows are shown with a red grid and can be easily associated to the correct change of pose. The following 6 rows are also showing the correct pose changes, but may be more difficult to evaluate visually as the identity homography is assigned to a non frontal plane (fifth and ninth rows).

employ evaluation criteria that is based on the relative order of the viewpoints. The paths in the view graph result in an ordering of the dataset images. Let $I_{p,i}$ denote the i-th image in path p. The order can be expressed by the binary labels $y_{p,i} \in \{+1, -1\}$ where $y_{p,i} = +1$ if, and only if, the viewpoint of $I_{p,i+1}$ is consistent with a left turn (with an angle between 0 and π) from the viewpoint of $I_{p,i}$. If $y_{p,j}$ is the ground-truth order and $\hat{y}_{p,j}$ the one estimated using the view graph construction, then a measure of the performance is the fraction of correctly-ordered image pairs:

$$\text{Acc} = \frac{1}{N} \sum_p \sum_i \frac{1 + y_{p,i}\hat{y}_{p,i}}{2}, \tag{7}$$

where N is the number of terms in the summation. While the ground-truth $y_{p,i}$ order is unknown for our unlabelled datasets, we estimate it by label transfer from Pascal 3D. We match each dataset image to the closest image in Pascal 3D. Then we use the viewpoints of the matched images to estimate $y_{p,i}$. The resulting performance is summarized in the following table:

Acc (%)	airplane	bicycle	chair	bus	car
tree	86.2	87.5	61.5	77.8	74.1
greedy	81.8	84.9	70.6	77.2	74.5

5. Conclusion

We have presented a weakly supervised method for building a view graph for a given category. At the core of our algorithm is the identification of simple geometrical structures in the scene (3D planes) without relying on any prior labeling. The method uses images to validate the hypotheses with large unlabeled datasets. When the geometrical structure hypothesis is incorrect, the synthesis of an image after applying a rotation may result in unrealistic deformations. The identified 3D structures are fundamental to determine affinities between images that are related by a small rotation. These affinities are then used to establish

Figure 4. Evaluation of image matches with 81 plane hypotheses. Rows 3, 6, and 9 show the original image with the corresponding best matches from the dataset. Rows 1, 4, and 7 show the rotation to the right of the best plane hypothesis (based on the realism score ρ). Rows 2, 5, and 8 show the rotation to the left of the best plane hypothesis (based on the realism score ρ). In the first column the best plane hypothesis for the original image is shown with a blue grid (the same is shown for left and right rotations) and its corresponding rotated version with a red grid. Notice that the identified plane hypotheses may correspond to different planes in the scene. In particular, in the case of the bus, there are two large planar surfaces (the front and the side of the bus). Which one of these two surfaces becomes dominant depends on the rotation as it might expose one surface more than the other. The first 6 rows favor the frontal plane while the last 3 rows favor the side plane.

links between images so that it is possible to make an object "spin". Although we limited this investigation to only left/right rotations, the procedure is applicable to a wider range of directions. Similarly, it is possible to explore other geometric primitives beyond planes. These further extensions will be subject of future work.

Acknowledgements

This work was supported by the Swiss National Science Foundation (SNSF) grant number 149227.

References

[1] S. Agarwal, Y. Furukawa, N. Snavely, I. Simon, B. Curless, S. M. Seitz, and R. Szeliski. Building rome in a day. *Commun. ACM*, 54(10):105–112, Oct. 2011. 2

[2] M. Aubry, D. Maturana, A. Efros, B. Russell, and J. Sivic. Seeing 3d chairs: Exemplar part-based 2d-3d alignment using a large dataset of cad models. In *CVPR*, 2014. 1, 2

[3] T. Brox and J. Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *PAMI*, 33(3):500–513, 2011. 3

[4] J. Carreira, A. Kar, S. Tulsiani, and J. Malik. Virtual view networks for object reconstruction. 2015. 3

[5] M. Cho, S. Kwak, C. Schmid, and J. Ponce. Unsupervised object discovery and localization in the wild: Part-based matching with bottom-up region proposals. *CoRR*, abs/1501.06170, 2015. 2, 3

[6] C. B. Choy, M. Stark, S. Corbett-Davies, and S. Savarese. Enriching object detection with 2d-3d registration and continuous viewpoint estimation. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, 2015. 2

[7] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, pages 1–2. Prague, 2004. 2

Figure 5. Visualization of the view graphs for the categories airplane, bicycle, chair, bus, and car. The top 6 rows show the results of the maximum spanning tree method, while the bottom 6 rows show the greedy method. Paths produced by the maximum spanning tree may be short due to the lack of good matches. For the greedy method we only show 10 images out of a 20 long path. We skip images because the object turns little otherwise.

[8] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *CoRR*, abs/1310.1531, 2013. 2

[9] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. In *CVPR*, 2014. 2

[10] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–

338, 2010. 3, 4

[11] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *PAMI*, 32(9):1627–1645, 2010. 2, 3

[12] A. Ghodrati, M. Pedersoli, and T. Tuytelaars. Is 2d information enough for viewpoint estimation. In *BMVC 2014*, volume 2, page 6. 1, 2, 3

[13] K. Grauman and T. Darrell. Unsupervised learning of categories from sets of partially matching image features. In *CVPR*, 2006. 2

[14] O. Haines and A. Calway. Recognising planes in a single image. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 37(9):1849–1861, 2015. 4

[15] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 3

[16] B. K. Horn and B. G. Schunck. Determining optical flow. In *1981 Technical symposium east*, pages 319–331. International Society for Optics and Photonics, 1981. 3

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pages 1097–1105. 2012. 2

[18] C. Liu, J. Yuen, and A. Torralba. Sift flow: Dense correspondence across scenes and its applications. *PAMI*, 33(5):978–994, 2011. 3

[19] J. L. Long, N. Zhang, and T. Darrell. Do convnets learn correspondence? In *Advances in Neural Information Processing Systems*, pages 1601–1609, 2014. 3

[20] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 3

[21] T. Malisiewicz, A. Gupta, and A. Efros. Ensemble of exemplar-svms for object detection and beyond. In *ICCV*, 2011. 1, 2

[22] M. Marques and J. Costeira. Estimating 3d shape from degenerate sequences with missing data. *Computer Vision and Image Understanding*, 113(2):261–272, 2009. 3

[23] L. Mei, M. Sun, K. M. Carter, A. O. Hero III, and S. Savarese. Unsupervised object pose classification from short video sequences. Technical report, DTIC Document, 2009. 1, 2

[24] J.-M. Morel and G. Yu. Asift: A new framework for fully affine invariant image comparison. *SIAM Journal on Imaging Sciences*, 2(2):438–469, 2009. 3

[25] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR*, 2006. 2

[26] B. Pepik, M. Stark, P. Gehler, and B. Schiele. Teaching 3d geometry to deformable part models. In *CVPR*, 2012. 1, 2

[27] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*. 2010. 2, 3, 4

[28] J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 22(8):888–905, 2000. 3

[29] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015. 3, 4

[30] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9(2):137–154, 1992. 3

[31] A. Torralba, R. Fergus, and Y. Weiss. Small codes and large image databases for recognition. In *CVPR*, 2008. 2, 3, 4

[32] S. Tulsiani, J. Carreira, and J. Malik. Pose induction for novel object categories. *CoRR*, abs/1505.00066, 2015. 1, 2

[33] S. Tulsiani and J. Malik. Viewpoints and keypoints. *CoRR*, abs/1411.6067, 2014. 2, 3

[34] S. Vicente, J. Carreira, L. Agapito, and J. Batista. Reconstructing pascal voc. In *CVPR*, 2014. 3

[35] C. Williams and M. Seeger. Using the nyström method to speed up kernel machines. In *Proceedings of the 14th Annual Conference on Neural Information Processing Systems*, number EPFL-CONF-161322, pages 682–688, 2001. 3

[36] Y. Xiang, R. Mottaghi, and S. Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *WACV 2014*, pages 75–82. 1, 2, 3, 4

[37] Y. Xiang and S. Savarese. Estimating the aspect layout of object categories. In *CVPR 2012*, pages 3410–3417, June. 2

[38] T. Zhou, Y. Jae Lee, S. X. Yu, and A. A. Efros. Flowweb: Joint image set alignment by weaving consistent, pixel-wise correspondences. In *CVPR*, 2015. 3

[39] M. Zia, M. Stark, B. Schiele, and K. Schindler. Detailed 3d representations for object recognition and modeling. *Pattern Analysis and Machine Intelligence*, 35(11):2608–2623, 2013. 1, 2