A Latent Model of Discriminative Aspect

Ali Farhadi [1], Mostafa Kamali Tabrizi [2], Ian Endres [1], David Forsyth [1]

[1] Computer Science Department
University of Illinois at Urbana-Champaign
{afarhad2,iendres2,daf}@uiuc.edu

[2] Institute for Research in Fundamental Sciences
kamali@ipm.ir

Abstract

Recognition using appearance features is confounded by phenomena that cause images of the same object to look different, or images of different objects to look the same. This may occur because the same object looks different from different viewing directions, or because two generally different objects have views from which they look similar. In this paper, we introduce the idea of discriminative aspect, a set of latent variables that encode these phenomena. Changes in view direction are one cause of changes in discriminative aspect, but others include changes in texture or lighting. However, images are not labelled with relevant discriminative aspect parameters. We describe a method to improve discrimination by inferring and then using latent discriminative aspect parameters. We apply our method to two parallel problems: object category recognition and human activity recognition. In each case, appearance features are powerful given appropriate training data, but traditionally fail badly under large changes in view. Our method can recognize an object quite reliably in a view for which it possesses no training example. Our method also reweights features to discount accidental similarities in appearance. We demonstrate that our method produces a significant improvement on the state of the art for both object and activity recognition.

1. Introduction

Appearance features have been very successful in both object and activity recognition. However, in each case, there are problems with **aspect** — the same thing can look very different in different aspects. We propose to shift the point of view of the aspect issue from geometrical aspect (or view) to discriminative aspect. Most multiview object recognition methods in existing literature treat aspect as a geometrical phenomenon. These models describe the geometrical relationship between an object and the camera. For example, the camera location in the room might be a rough proxy for aspect. However, we believe this notion of aspect is not well suited for recognition tasks. First, it imposes the unnatural constraint that objects in different classes be aligned into corresponding pairs of views. This is unnatural because the front of a car looks different to the front of a bicycle. Worse, the appearance change from the side to the front of a car is not comparable with that from side to front of a bicycle. Second, the geometrical notion of aspect does not identify changes that affect discrimination. We prefer a notion of aspect that is explicitly discriminative, even at the cost of geometric precision.

Our model of discrimination is mapping an appearance feature to a hash code. This is a very general model. The mapping could be achieved with many different methods. An ideal mapping would assign all instances of a given class to one bucket. Discriminative aspect parameterizes two important nuisance effects. Sometimes images that should be mapped to the same bucket are not, for example, because there was a large change in the viewing direction. Sometimes images that should be mapped to different buckets are not, for examples, because quite different objects happen to share many appearance features. An explicit representation of discriminative aspect will allow the recognizer to focus on features that are more discriminative for this case.

Generally, we expect images of the same object from different viewing directions to have different discriminative aspects. We expect images of different objects that have roughly the same appearance to have similar discriminative aspects. For example, a frontal view of a car and a side view of a computer mouse are similar in rough appearance space. They both share a strong curved contour across the top. If they share the same discriminative aspect the model can uncover the more detailed features in which they differ.

If each image was labeled with a discriminative aspect, we could use this information in a straightforward way. We would weight the feature vector by discriminative aspect parameters so as to emphasize some feature differences. For example, we weight the appearance features of the side view of the mouse and the frontal view of the car the same. This naturally suggests a bilinear model [12] which allows us to simultaneously learn appearance models and the underlying latent discriminative aspect. However, we don't know the discriminative aspects for training or testing objects. So, the discriminative aspect representation is inherently latent.

In this work, our contributions are a) introducing the notion of discriminative aspect b) showing the advantages of discriminative aspect over geometric aspect c) using a latent continuous variable to model the aspect rather than the traditional discrete camera index d) and supporting this novel model with considerable performance gains over the state of the art in two main problems of computer vision that suffer from the aspect issue, object recognition and human activity recognition.

Aspect involves a rich collection of geometric and photometric phenomena: objects can change appearance because one sees a different outline, because occlusion relations change, and because changes in viewing direction affect apparent color and texture. A core problem is to build a recognizer that can be trained from some aspects, and will work successfully on new aspects: we refer to this property as **transfer across aspect**.

Aspect in object recognition: There are three main strategies for handling aspect. One might attempt to build a **comprehensive representation of aspectual phenomena** (an **aspect graph**; review in [3], critique in [7], summary of results in [11]). This strategy usually results in unmanageably complex representations and has largely fallen into disuse. One might attempt to represent an object using **aspect-enriched models**. In the extreme, rather than build a "car" recognizer, one might build "frontal-car", "lateral-car" and "overhead-car" recognizers then drop the aspect label after classification. Usually, these multiple classes are compacted into a single model, assembled from local patches (which might have quite simple behavior), tied together by observation [25], with geometric reasoning [14], with statistical reasoning [18, 20], or with a combination [24]. This strategy is expensive in data. However, one may interpolate missing aspects [4], or interpolate models corresponding to missing aspects [23]. An important difficulty with this strategy is that it treats categories one-by-one. We do not expect to see many aspects of a new object to be able to recognize it. This is because objects tend to share aspectual properties, so that, for example, relations between "frontal-car" "lateral-car" and "overhead-car" recognizers should be similar to relations between "frontal-box", "lateral-box" and "overhead-box" recognizers. For example, the different objects ("bicycle", "car", "mouse", "toaster", "cellphone", "head", "iron", "monitor", "shoe" and "stapler") in the aspect dataset of [24] are aligned *to one another*, which wouldn't be possible if there wasn't at least a rough consensus between human observers that there is something comparable between the aspects of a mouse and the aspects of a car. Finally, one might attempt to **build aspect invariant features** (e.g. [10]). In its most direct form, this strategy usually applies only to quite specialized cases. However, there are several constructions of aspect robust features for human activity.

Activity recognition: Recognizing human activity is a core computer vision problem. Reviews appear in [9, 13]. There are rich applications in surveillance, automated interpretation of video, and search. There are two important threads: first, one might recognize activities using discriminative methods applied to image features either computed by segmenting the body (for example [1]), or from characteristic motion fields (for example [19]). Second, one might build generative models from image data (for example, [8]), or motion capture data (for example, [16]).

Aspect and activity: Procedures that infer a 3D configuration of the body can be relatively robust to viewpoint [16], but require one to infer that 3D representation (there are real difficulties here; review in [9]). The alternative is to use an appearance feature, and try to make it aspect invariant. Despite the considerable complexities of aspect phenomena related to human figure, versions of this approach have been quite successful for activity recognition. Junejo *et al.* give a direct construction of features that are robust to aspect changes [17]. An alternative is to use a two-stage strategy, where the first stage uses an estimate of aspect together with image features to produce a new set of aspect independent features. One then classifies using these feaures. Farhadi *et al.* use this approach to transfer word-spotters across aspect for ASL [5]. Farhadi and Kamali use this approach to recognize activities from a novel aspect, using a quantized aspect representation [6].

We differ from [5], [6], since a) we use a discriminative rather than geometric notion of aspect b) we treat aspect as a latent variable and learn it simultaneously with the appearance model c) our latent discriminative model of aspect allows us to encode the interactions between aspect and the appearance rather than learning the appearance for fixed viewing directions d) and therefore we aren't limited to transferring object models only between single pairs of aspects as in [6] e) Our notion of discriminative aspect has more intuitive semantics compared to the aspect indicators in [6] and [5]. Compare Figure 4 with Figure 5.

We compare our results with the state of the art recognition systems for novel aspects in both object recognition and human activity recognition. The results show a significant gain in adopting the discriminative continuous aspect model.

2. Latent Model of Aspect

After presenting a learner with many objects under many viewing directions, we want to recognize on viewing directions for which we haven't observed a particular object. Given enough training instances we can model the shared aspectual behavior between visually similar objects. This enables us to recognize an object in viewpoints for which we have no training examples for that particular object. We can do this using a hash code that shares a values for an object class under different views and also discriminates between different objects under any viewpoints.

2.1. Hashing with Discriminative Aspect

Suppose we possess the labels of discriminative aspect v for the corresponding appearance features x. If we simply form a new feature by appending v to x, we cannot use the interactions between aspect and appearance. However, a bilinear form allows the aspect vector v to influence the appearance feature x by reweighting individual components. This yields a classifier of the form $sign(v^t Ax + b)$ which is linear in the classifier matrix A. This simple bilinear model predicts hash codes using a reweighted set of features. While we use a bilinear SVM formulation in this paper, our framework can utilize any other classifier, such as decision trees.

Unfortunately, the discriminative aspect labels are not available. We encode the aspect with a latent model. Due to the close interaction between discriminative aspect and appearance we can use the geometric aspect as a rough prior for the discriminative aspect.

We want to learn to predict hash codes using (x, v) pairs. For this we need training labels to define the bits of the code. These codes should be similar for objects within a class and discriminative across classes. Each code is a clustering of object classes. To obtain discriminative codes, we randomly search the large space of possible hash codes. We generate thousands of random hash bits and choose the most discriminative ones. These selected bits are used as training labels for learning aspect invariant hash codes.

2.2. Latent Bilinear Model

We now have the following situation: for each training image we have an object label, and can search for good hash codes. We have a poor estimate of v for each training data item, supplied by the view label on the image. Learning involves simultaneously refining the estimate of v, and coupling this to appearance features to predict hash codes.

There is one v per image indexed by i, and the j^{th} bit of the hash code is predicted using the matrix of linear classifiers A_j. We use a $Regularization + Loss$ minimization framework. We penalize $\|A_j\|^2$ to avoid getting big A's. We allow discriminative aspects to vary from the given geometric aspect, while at the same time controlling this deviation. We do so by penalizing $\|v_i - v_{\mu(i)}\|^2$ where $v_{\mu(i)}$ is the average discriminative aspect of the examples of i's corresponding view. This imposes a prior on the discriminative aspect parameters requiring that, in the absence of other information, discriminative aspect should mirror the view. We write i for the index to examples, j for the index to hash code bits, ξ for the slack variable, and $s_j(i) \in \{-1, 1\}$ for the training label of the j^{th} bit of the hash code for the i^{th} image. Then we have

$$\min_{A_j, v_i} \quad \textstyle\sum_j \| A_j \|^2 + \sum_i \| v_i - v_{\mu(i)} \|^2 + C \sum_i \xi_i \tag{1}$$

$$s.t. \quad s_j(i) v_i^t A_j x_i \geq 1 - \xi_i \tag{2}$$

$$\xi_i \geq 0 \tag{3}$$

We have not found a need to regularize $\| v_{\mu_i} \|^2$. We conjecture that the $Loss$ term will not let $v_{\mu(i)}$ get big. We solve this optimization in primal, because it is straightforward to do so, and because improving the primal when we are not at the extremal point will also improve the the actual risk [2]. Therefore, we must minimize:

$$\min_{A_j, v_i} \sum_j \| A_j \|^2 + \sum_i \| v_i - v_{\mu(i)} \|^2 + \lambda \sum_{i,j} h(v_i^t A_j x_i, s_j(i)) \tag{4}$$

where h is the hinge loss. Notice that many observations share a variable here: while each image has its own v_i, these v_i's are coupled by the $v_\mu(i)$ term; and each image participates in computing many different hash code bits. Similarly, each hash code bit uses the same A_j for multiple images and aspects.

Optimization: We alternate between minimizing over A_j and v_i for fixed v_μ, and updating v_μ with a mean. The objective function is not differentiable due to the hinge loss, but can be approximated to arbitrary precision by a piecewise polynomial function. We apply limited memory BFGS [22] to the resulting problem.

Initialization: We need good initializations for v_i and A_j. We believe that a very rough representation of appearance provides a decent starting point for v. Therefore, we use projections of the appearance features to the first few principal directions. The central aspect v_μ is initialized by taking the average of v_i's of all images belonging to the same geometrical aspect. We fix v_i and solve the optimization problem above to get an initial estimate of A_j.

2.3. Recognition with Discriminative Aspect

Once we obtain the final v_i's (for training images) and A_j's from the optimization explained in the previous section, we can start recognizing objects. For that we need to have a good estimate of v_i for test frames.

We expect objects that roughly share similar appearance to also share discriminative aspect parameters. Thus, we infer a v for each test image using the average v from the nearest training neighbors in the appearance feature space. These nearest neighbors may not all belong to the same object class. This means that we can in fact share aspectual behavior across classes.

Once we have v_i, we can compute the hash values for each test example. We can now apply a classifier to predict object classes using hash codes. Since these hash codes are aspect invariant, we can recognize objects in aspects for which we haven't seen them before. We use nearest neighbors in the hash space, with examples coming from any available *training* aspects.

Improving discriminative aspects from object class hypotheses: The estimated v for a test example may be noisy. However, after predicting the hash code, we can correct our noisy estimate of v. By fixing the predicted hash code and the learned A_j, we can run the optimization 1 over v using the noisy estimate as the initial value.

2.4. Evaluations

We have tested our methods in two major problems in computer vision, object recognition and human activity recognition. We show that we recognize objects in views for which we have no training example of those objects. We compare our performance against similar methods and methods that require observing examples of all of the objects in all the views, in both object and human activity recognition literatures.

3. Results: Object Recognition

We can recognize objects in views for which we haven't observed those objects using models trained on different views. We apply our techniques to a recently released 3D object dataset. To be able to compare with [24] and [23], we run our experiments under the same conditions as in these two papers.

Object Recognition Dataset: The 3D object dataset from Savarese and Fei-Fei [24] is well suited to demonstrate our methods. The dataset contains geometric aspects varied over azimuth, elevation, and scale. Each angle and scale is aligned across objects to their corresponding canonical orientations. For example, each image of a forward facing shoe is aligned with each forward facing car figure 1.

Object Recognition Features: We generate a 72 dimensional histogram feature. To encode characteristic appearance information, we include a bag of words feature for texture, shape, edges, and color over keypoints inside the masks. Texture descriptors are computed for each pixel, and quantized to the nearest 256 kmeans centers. The texture descriptor is extracted with a texton filterbank. Shape is encoded with an HOG spatial pyramid, using 8x8 blocks, a 4 pixel step size and 2 scales per octave. Edges are found using a standard canny edge detector and their orientations are quantized into 8 unsigned bins . Finally, color descriptors are quantized to the nearest 128 kmeans centers. The color descriptor consists of the LAB values. By concatenating the silhouette and appearance features, we obtain a final 1465 dimensional feature for each instance.

Protocol: There are 8 different angles, 3 different heights, and 3 different scales. Because of the limitations in the number of objects in the dataset, we choose to follow the strategy of leaving out objects in views. This means that we never observe a single example of the held out object category in the target view, the view in which we want to recognize that object.

Procedure: We need to allocate codes for objects. To produce the hash code we generate 1000 random code bits and choose the 40 most discriminative bits. Ideally these codes should be the same for same objects and different for different objects in any aspect. We get an estimate of the discriminative aspect vector v by using PCA on the appearance feature. We pick the first 3 principal components to form an initial v. We then fix v's and use the optimization 1 to get an initial estimate of A's. Now we run the optimization 1 alternating between optimizing for v, A and updating v_μ using v. Having obtained the final v's and A's, we can now do the inference by finding out the v of a test example. For each test example we look for the 3 nearest neighbors in the appearance feature space and use their average v as a initial aspect vector for the test image. We then compute the hash codes for the test image. Finally, we predict the object class using the 3 nearest neighbors in the hash code space.

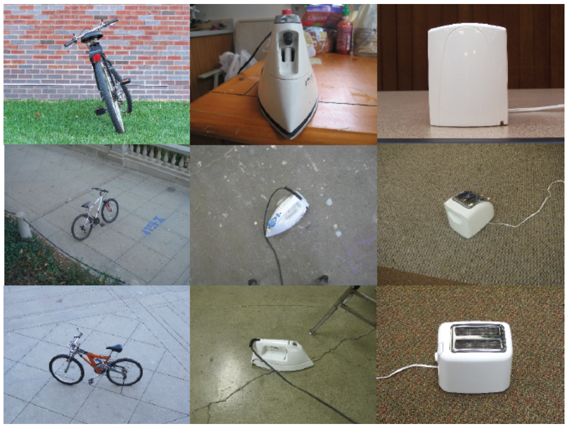


Figure 1. **Object view data examples**, taken from the dataset of Saverese and Fei-Fei [24]. Notice that there are distinct instances of distinct categories at aspects that are, rather roughly, aligned to one another (i.e. the "front" of the bike corresponds to the "front" of the iron). This demonstrates the notion of geometric aspect. Aligning objects based on their geometric aspect is unnatural, because, for example the front of the bike is dissimilar to the front of the iron. Even worse, the appearance changes between the front and side of the bike significantly differs from that of the iron.

Discriminative Aspect inference in testing: We must produce an aspect estimate to be able to evaluate features for test images, and we do so with the three nearest neighbors. The nearest neighbor estimation of v for test examples is expected to be noisy. However, since the aspectual behavior pools correctly across objects, the nearest neighbors are a good guide to the aspect even if they are not a particularly good guide to the object identity Figure 3.

Results and Comparisons: The task is to recognize objects in aspects for which we observe *no* examples of that object. We adapt the same experimental settings as [24, 23]. Figure 2 compares our performance with that of [24, 23]. Our average accuracy is **78.16%** comparing to 64.78%, and 46.80% of Savarese et al.'08 and 07. We even outperform [24] when they observe all the classes in all the aspects (where they get 75.7% performance). We adopted this dataset and the experimental settings from [24]

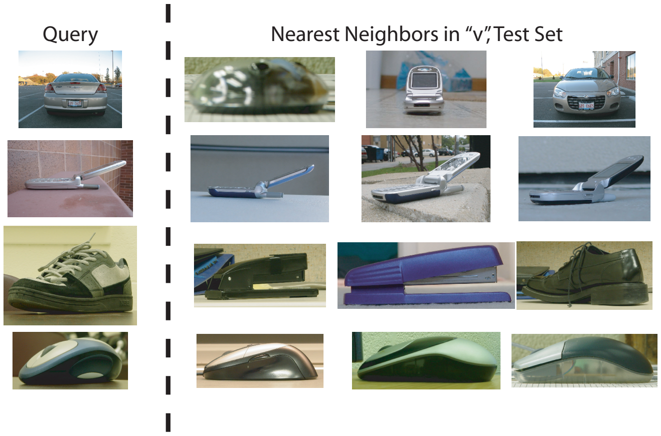


Figure 3. **Test aspect labels** are inferred using nearest neighbors. These labels are accurate, too. This figure shows four test example and their three nearest neighbors in the the v space. Now we do not know the class of the test image (that's why we're trying to estimate v), so the nearest neighbors may not even be of the same class. However, they are at the same discriminative aspect. This means that discriminative aspect estimates are pooled across comparable classes — any boxy object can serve as a cue for the aspect of another. This is a most desirable property.

and [24, 23] as they are standard. However, objects in this dataset has symmetries. To avoid this issue we evaluate our method in a better experimental design. we omit all examples of the test object in all views except one, but observe all other examples of all other objects in all other views. We test on another view of the test object. We repeat this and average the performance over all non-symmetric views of all objects. In this setting, KNN gets 12.3%, our model gets 68.6%. More, this dataset is not a challenging test set for object recognition as one may get surprisingly good results in recognizing objects by looking only at the background pixels.

Semantics of Discriminative Aspects We have shown that if we estimate discriminative aspect properly, we can improve recognition. This latent variable has intuitive semantics. In figure 7 we can see several cases where objects share discriminative aspectual similarity: a) objects within a class sharing the same geometric aspect b) objects across classes sharing geometric aspect c) objects across classes and across geometric aspects. The underlying property for each of these cases is that objects sharing discriminative aspect also share rough visual appearances.

4. Results: Human Activity Recognition

We can recognize activities in aspects for which we haven't observed those activities using models trained on different aspects. We have tested our method on the IXMAS data set (figure 4), where we outperform state of the art methods in multi-view activity recognition.

Dataset: We picked the IXMAS dataset [26] because there are 5 different views of activities, sequences are time aligned, and silhouette information is also provided. There are 11 different actions in this dataset performed by 10 different actors three times. Furthermore, we also wanted to be able to compare our results with [6] and [15] which were tested on IXMAS.

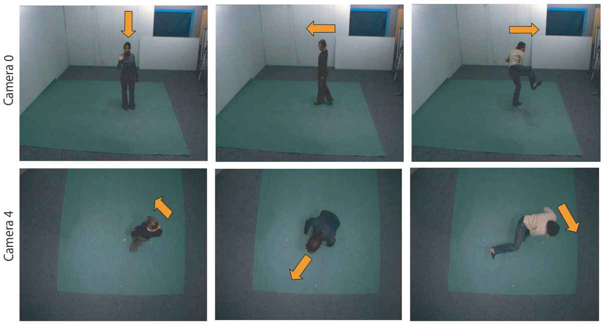
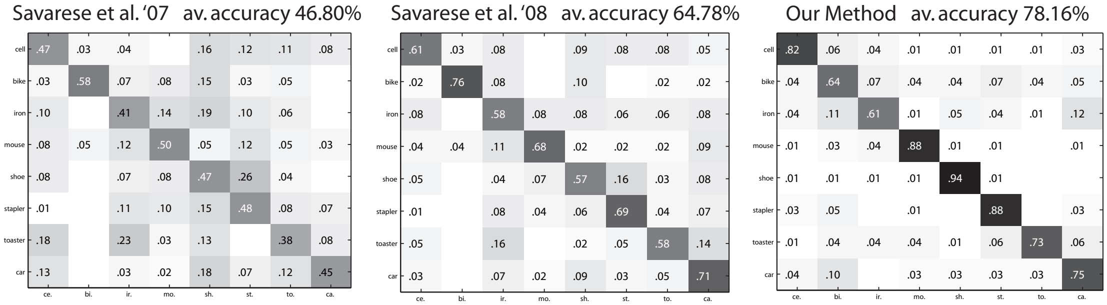


Figure 4. **Camera index as an aspect indicator in IXMAS data:** Each row shows frames in a fixed camera from IXMAS dataset. Although the camera index can be a good proxy for the aspect, when an actor moves while performing an action the aspect changes. For example in the first row, despite the fact that the actress is in frontal view in the left frame and she is in the left lateral view in the middle frame and in the right frontal view in the right frame, all of the frames have the label 0 as their aspect in [6]. This suggests that by using the camera index as an estimation of the aspect we will suppress some useful details.

Features: We employ the activity features in [6]. Their frame descriptor is a histogram of the silhouette and of the optic flow. Given the bounding box of the actor and the silhouette, the optic flow is computed using Lucas-Kanade algorithm [21]. Features consist of three channels: smoothed horizontal flow, smoothed vertical flow, and the silhouette. This results in three 72-dimensional histograms corresponding to each channel. To encode local temporal structure of the activities we consider stacking features from previous and next frames. We pick the first 50 principal components of the descriptors of a window of size 5, centered at the frame we want to describe. For further frames in both directions, previous and next frames, we pick the first 5 principal components of the windows of size 5, centered at the $(i+5)^{th}$ and $(i-5)^{th}$ frames. This gives us a 60 dimensional descriptor for each frame. We perform a clustering of the feature space to form 40 clusters. We represent activities using histograms of the frames assigned to each cluster. We assign the action label with the closest histogram of clusters to the test sequence. We use hamming distance for matching the histograms.

Protocol: We use all five aspects in the IXMAS dataset. There are 20 different possible transfer scenarios for transferring from each of these cameras to the other one. Because of the small number of actions in the dataset, we choose to follow the strategy of leaving one action out. This means that we observe all actions in the source aspect and all but the selected action in the target aspect. Therefore, we never observe a single example of the selected action in the target aspect, the aspect in which we want to recognize



Savarese et al. ‘07 av. accuracy 46.80%

	ce.	bi.	ir.	mo.	sh.	st.	to.	ca.
cell	.47	.03	.04		.16	.12	.11	.08
bike	.03	.58	.07	.08	.15	.03	.05	
iron	.10		.41	.14	.19	.10	.06	
mouse	.08	.05	.12	.50	.05	.12	.05	.03
shoe	.08		.07	.08	.47	.26	.04	
stapler	.01		.11	.10	.15	.48	.08	.07
toaster	.18		.23	.03	.13		.38	.08
car	.13		.03	.02	.18	.07	.12	.45

Savarese et al. ‘08 av. accuracy 64.78%

	ce.	bi.	ir.	mo.	sh.	st.	to.	ca.
cell	.61	.03	.08		.09	.08	.08	.05
bike	.02	.76	.08		.10		.02	.02
iron	.08		.58	.08	.08	.06	.06	.08
mouse	.04	.04	.11	.68	.02	.02	.02	.09
shoe	.05		.04	.07	.57	.16	.03	.08
stapler	.01		.08	.04	.06	.69	.04	.07
toaster	.05		.16		.02	.05	.58	.14
car	.03		.07	.02	.09	.03	.05	.71

Our Method av. accuracy 78.16%

	ce.	bi.	ir.	mo.	sh.	st.	to.	ca.
cell	.82	.06	.04	.01	.01	.01	.01	.03
bike	.04	.64	.07	.04	.04	.07	.04	.05
iron	.04	.11	.61	.01	.05	.04	.01	.12
mouse	.01	.03	.04	.88	.01	.01		.01
shoe	.01	.01	.01	.01	.94	.01		
stapler	.03	.05		.01		.88		.03
toaster	.01	.04	.04	.04	.01	.06	.73	.06
car	.04	.10		.03	.03	.03	.03	.75

Figure 2. **A comparison** of three recognition methods for recognition in the presence of strong aspectual phenomena. On the left, the class confusion matrix for the method of Savarese et al [24], where the recognizer possesses instances of each class at each aspect. In the center, the class confusion matrix for the work of Savarese et al [23], where the recognizer possesses instances of each class at most aspects, but must interpolate models to cover some aspects. On the right, the class confusion matrix for our method, where the recognizer has no example of a test image's class at the view we want to recognize. Our model of aspect offers a substantial gain.

that activity. To report the final accuracy we average over all possible combinations of leaving one action out. This strategy has been used in [6].

Procedure: The training procedure is similar to that of object recognition.

Discriminative Aspect inference in testing: This procedure is identical to aspect inference in the object recognition experiment. The nearest neighbor estimations are again reasonable Figure 6. While errors in estimating aspect do occur, if the estimate is not very bad we can still produce an action label, which is usually the case. This suggests that action labels could be used to refine aspect estimates Figure 6.

For example, given a test dataset marked with correct action labels (but no aspect information), we might be able to tag each frame with an accurate discriminative aspect (as we have shown with the training set). Because of the spatial characteristics of human activity, the discriminative aspect tends to agree with the geometric aspect figure 6. Aspect tagging with the degree of accuracy suggested by figure 6 is hard to achieve any other way, and likely to be valuable in producing datasets.

Results and Comparisons: Table 1 shows the recognition performance for all of the possibilities. We compare the results of recognizing activities learned in one aspect and tested in another aspect using our model of aspect with (a) the method in [6] using quantized and geometric aspect (b) the method in [15] which constructs features robust to change of aspect using temporal self similarity measures. All methods are tested under the same conditions. As table 1 shows we strongly outperform both methods by considering our model of aspect. On average, the quantized method of [6] gets an accuracy of 60.2%, the self similarity method of [15] gets an accuracy of 62%, and our model of discriminative aspect gets an accuracy of **76.7%**. This clearly shows that adopting a discriminative model of aspect can improve classification performance dramatically. As a baseline, one can simply train in one aspect and test in another aspect. This is known to perform poorly on this dataset; [6] gets an average accuracy of 23%.

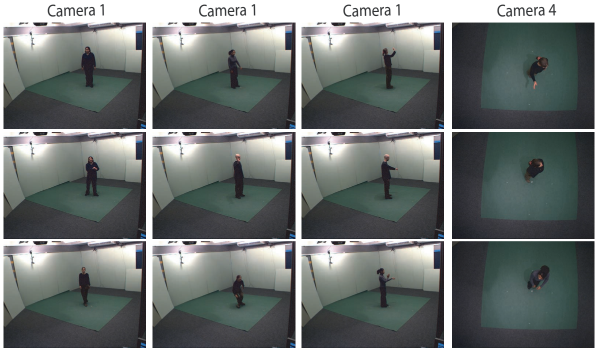


Figure 5. **Clusters of aspect variable:** The v variable inferred in training using our methods agrees strongly with the camera angle. This figure shows images for four clusters of v's inferred for training examples. Clusters are obtained by K-means. Each column is a different cluster. Notice that the first three clusters are from the camera 1. This means that even for frames coming from the same camera we can find cluster of aspects which are different across the clusters and similar inside clusters. Column 1 shows frames from camera 1 which has frontal aspect, column 2 has $3/4$ view, and column 3 has lateral view.

Aspect inference in training: We are obliged to infer the correct value for aspects in learning. One test of merit is to determine whether the inferred values make sense. We do so by clustering aspect variables, and comparing clusters to intuition. Figure 5 shows three examples of 4 different cluster from different cameras. The first three columns come from camera 1, and the last one from camera 4. The v variable inferred in training is strongly coupled to camera angle. Notice that in column 1, frames from camera 0 give a frontal view, in column 2, a $3/4$ view, and in column 3 a lateral view. However our procedure infers v's that cluster together these aspects. This means that $3/4$, frontal, and lateral views are each assigned aspect variables that are similar for different images from the same view direction, but differ for images from different view directions.

	Camera 0			Camera 1			Camera 2			Camera 3			Camera 4		
	QV	SS	CV	QV	SS	CV	QV	SS	CV	QV	SS	CV	QV	SS	CV
Camera 0	76	76	**84**	72	78	**79**	61	69	**79**	62	**70**	68	30	45	**76**
Camera 1	69	**77**	72	76	78	**85**	64	**74**	**74**	68	67	**70**	41	44	**66**
Camera 2	62	66	**71**	67	71	**82**	68	74	**87**	67	64	**76**	43	54	**72**
Camera 3	63	69	**75**	72	70	**75**	68	63	**79**	73	68	**87**	44	44	**76**
Camera 4	51	39	**80**	55	39	**73**	51	52	**73**	53	34	**79**	51	66	**80**

Table 1. **Results**: Columns give the result of testing on the camera heading the column (in bold, the target), when trained on the camera given in the row (normal, the source). There are three cases for each transfer scenario: first, using the method in [6] with quantized aspect (QV). Second, using the self similarity metrics in [15] (SS). Third using our continuous model of aspect (CV). For example, training on camera 4 and testing on camera 0 gives an accuracy of 51% if one uses the quantized aspect and 39% if one uses self similarity measures, and 80% if one uses our continuous model of aspect. Notice that our continuous model of aspect significantly improves the accuracy of activity recognition in novel aspects. Transferring classifiers across aspects itself is a hard and challenging problem. Average accuracy of 23% clearly shows how challenging this problem is. On average, the quantized method of [6] gets an accuracy of 60.2%, the self similarity method of [15] gets an accuracy of 62%, and our continuous model of aspect gets an accuracy of **76.7%**. Our discriminative continuous model of aspect outperforms other methods significantly.

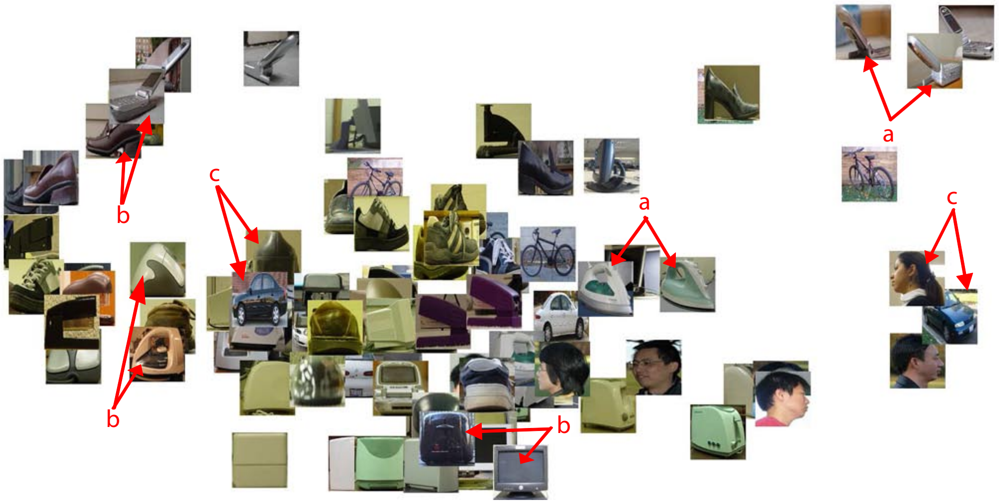


Figure 7. **The discriminative aspect map:** We project the discriminative aspect learned during the training process into a two dimensional space using multidimensional scaling. This figure shows the tendency of particular objects to share discriminative aspectual behavior. Sometimes objects from the same class share discriminative aspect (a), for example, the two cell phones in the top right corner are assigned discriminative aspects that agree with their geometric aspect. This agreement between discriminative and geometric aspect may also occur across classes (b) when the objects share rough appearance features, such as the shoe and cell phone or toaster and monitor. However, it is not necessary for discriminative and geometric aspects to agree, such as when objects at different viewpoints share strong rough visual appearances (c). Examples include the contours of the back of a shoe and the side of a car, or the strong diagonal of the head and car.

5. Conclusion

We have introduced a novel model of aspect. Discriminative aspect represents phenomena that interfere with discrimination, causing images of the same object to look different or images of different objects to look similar. A significant component of discriminative aspect is produced by view effects, and we use this fact to infer discriminative aspect parameters for images during training. Another important component is similarity in appearance, and this means that nearest-neighbor estimates are sufficient to provide useful discriminative aspect information at testing time. We have shown that using discriminative aspect information produces substantial improvements over the state of the art on standard datasets for two important problems: view transfer in object recognition and in activity recognition. Our maps of discriminative aspect indicate that the parameters have useful semantics; images from similar views tend to share discriminative aspects as do images of different objects that have strong appearance similarities. This means that discriminative aspect can reweight features to take account of likely local confusions, and we conjecture that this reweighting is the source of the improvements. It is intriguing to speculate that our methods might be applicable to any discriminative task, if one could provide a proxy to drive the initial inference of discriminative aspect.

6. Acknowledgments

This work was supported in part by the National Science Foundation under IIS -0803603 and in part by the Of-

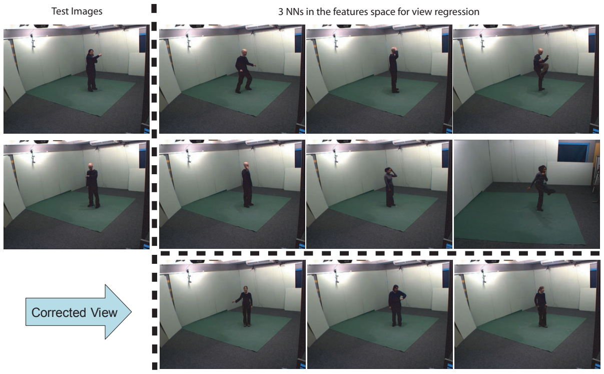


Figure 6. **Aspect inference for test images,** On the top left, we show a test image and its three nearest neighbors in the appearance features shown on the top right. Typically, the average discriminative aspect for these nearest neighbors is a good estimate of v. In the second row, we show a test frame whose nearest neighbors are of varying aspect and whose aspect estimate is poor as a result. If the estimate is not very bad, we can still produce an action label. Conditioned on this correct action label, we can produce the improved aspect estimate whose nearest neighbors in aspect space are given in the bottom row.

fice of Naval Research under N00014-01-1-0890 as part of the MURI program. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect those of the National Science Foundation or the Office of Naval Research.

References

[1] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *IEEE International Conference on Computer Vision (ICCV)*, 2005. 2

[2] DEnnis Decoste Bottou Leon, Chapelle Oliver and Weston Jason. *Large-scale kernel machines*. 3

[3] K.W. Bowyer and C.R. Dyer. Aspect graphs: an introduction and survey of recent results. 2:315–328, 1990. 2

[4] H-P. Chiu, L. P. Kaelbling, and T. Lozano-Perez. Virtual training for multi-view object class recognition. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, 2007. 2

[5] A. Farhadi, D. Forsyth, and R. White. Transfer learning in sign language. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2007. 2

[6] A. Farhadi and M. Kamali. Learning to recognize activities from the wrong view point. In *European Conf. Computer Vision*, 2008. 2, 5, 6, 7

[7] O.D. Faugeras, J.L. Mundy, N. Ahuja, C.R. Dyer, A.P. Pentland, R. Jain, K. Ikeuchi, and K.W. Bowyer. Why aspect graphs are not (yet) practical for computer vision. *Computer Vision, Graphics and Image Processing*, 55(2):212–218, March 1992. 2

[8] Xiaolin Feng and P. Perona. Human action recognition by sequence of movelet codewords. In *3D Data Processing Visualization and Transmission, 2002. Proceedings. First International Symposium on*, pages 717–721, 2002. 2

[9] D.A. Forsyth, Okan Arikan, Leslie Ikemoto, James O'Brien, and Deva Ramanan. Computational aspects of human motion i: tracking and animation. *Foundations and Trends in Computer Graphics and Vision*, 1(2/3):1–255, 2006. 2

[10] D.A. Forsyth, J.L. Mundy, A.P. Zisserman, C. Coelho, A. Heller, and C.A. Rothwell. Invariant descriptors for 3d object recognition and pose. *PAMI*, 13(10):971–991, 1991. 2

[11] D.A. Forsyth and J. Ponce. *Computer Vision: a modern approach*. Prentice-Hall, 2002. 2

[12] William T. Freeman and Joshua B. Tenenbaum. Learning bilinear models for two-factor problems in vision. In *IEEE Conf. on Computer Vision and Pattern Recognition*. 2

[13] Weiming Hu, Tieniu Tan, Liang Wang, and Steve Maybank. A survey on visual surveillance of object motion and behaviors. *IEEE transactions on systems, man, and cyberneticspart c: applications and reviews*, 34(3), 2004. 2

[14] C-Y. Huang, O.T. Camps, and T. Kanungo. Object recognition using appearance-based parts and relations. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 877–83, 1997. 2

[15] I. Laptev I. Junejo, E. Dexter and P. Perez. Cross-view action recognition from temporal self-similarities. In *Publication interne N 1895, ISSN 1166-8687, Irisa, Rennes*, 2008. 5, 6, 7

[16] Nazli Ikizler and D.A. Forsyth. Searching video for complex activities with finite state models. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007. 2

[17] I. Junejo, E. Dexter, I. Laptev, and P. Perez. Cross-view action recognition from temporal self-similarities. Technical report, Irisa, Rennes, 2008. Publication interne N 1895, ISSN 1166-8687. 2

[18] A. Kushal, C. Schmid, and J. Ponce. Flexible object models for category level 3d object recognition. In *IEEE Conf. on Computer Vision and Pattern Recognition*, 2007. 2

[19] I. Laptev and P. Pérez. Retrieving actions in movies. In *Int. Conf. on Computer Vision*, 2007. 2

[20] S. Lazebnik, C Schmid, and J Ponce. Semi-local affine parts for object recognition. In *British Machine Vision Conference*, 2004. 2

[21] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stero vision. *IJCAI*, 1981. 5

[22] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, August 1999. 3

[23] S. Savarese and L. Fei-Fei. View synthesis for recognizing unseen poses of object classes. In *European Conf. Computer Vision*, 2008. 2, 4, 5, 6

[24] S. Savarese. and Li Fei-Fei. 3d generic object categorization, localization and pose estimation. In *Int. Conf. on Computer Vision*, pages 1–8, 2007. 2, 4, 5, 6

[25] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, B. Schiele, and L. Van Gool. Towards multi-view object class detection. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1589–1596, 2006. 2

[26] Daniel Weinland, Remi Ronfard, and Edmond Boyer. Free viewpoint action recognition using motion history volumes. *Computer Vision and Image Understanding*, 2006. 5