Latent Hough Transform for Object Detection

Nima Razavi[1], Juergen Gall[2], Pushmeet Kohli[3], and Luc van Gool[1,4]

[1]Computer Vision Laboratory, ETH Zurich
[2]Preceiving Systems Department, MPI for Intelligent Systems
[3]Microsoft Research Cambridge
[4] IBBT/ESAT-PSI, K.U. Leuven

Abstract. Hough transform based methods for object detection work by allowing image features to vote for the location of the object. While this representation allows for parts observed in different training instances to support a single object hypothesis, it also produces false positives by accumulating votes that are consistent in location but inconsistent in other properties like pose, color, shape or type. In this work, we propose to augment the Hough transform with latent variables in order to enforce consistency among votes. To this end, only votes that agree on the assignment of the latent variable are allowed to support a single hypothesis. For training a Latent Hough Transform (LHT) model, we propose a learning scheme that exploits the linearity of the Hough transform based methods. Our experiments on two datasets including the challenging PASCAL VOC 2007 benchmark show that our method outperforms traditional Hough transform based methods leading to state-of-the-art performance on some categories.

1 Introduction

Object category detection from 2D images is an extremely challenging and complex problem. The fact that individual instances of a category can look very different in the image due to pose, viewpoint, scale or imaging nuisances is one of the reasons for the difficulty of the problem. A number of techniques have been proposed to deal with this problem by introducing invariance to such factors. While some approaches [1–3] aim at achieving partial invariance through specific feature representations, others [4–9] divide the object into parts, assuming less variation within each part and thus a better invariant representation.

Codeword or voting based methods for object detection belong to the latter category. These methods work by representing the image by a set of *voting elements* such as interest points, pixels, patches or segments that vote for the values of parameters that describe the object instance. The Implicit Shape Model [4], an important instance of the Generalized Hough Transform (GHT), represents an object by the relative locations of specific image features with respect to a reference point. For object detection, the image features vote for the possible locations of this reference point. Although this representation allows for parts from different training examples to support a single object hypothesis, it also produces false positives by accumulating votes that are consistent in location but inconsistent in other properties like pose, color, shape or type. For example, features taken from a frontal view image of a training example and a side view image of another training example might agree in location, but an object can not be seen from frontal and side views at the same time. It is our understanding that this accumulation of inconsistent votes is the main reason behind the poor performance of the voting based approaches.

To improve the detection performance, researchers have proposed enforcement of consistency of the votes by estimating additional parameters like aspect ratio [5] or pose [10–14]. While the use of more parameters obviously improves consistency, it also increases the dimensionality of the Hough space. However, Hough transform-based methods are known to perform poorly for high-dimensional spaces [15]. Consistency can also be enforced by grouping the training data and voting for each group separately. Such a grouping can be defined based on manual annotations of the objects, if available, or obtained by clustering the training data. While this does not increase the dimensionality of the voting space, the votes for each group can become sparse due to a limited number of training examples, which impairs the detection performance. Even if annotations are available for the training examples, it is not clear which properties to annotate for an optimal detection performance since the importance of properties differs from case to case. For instance, viewpoint is important for detecting airplanes but far less so for detecting balls.

In this work, we propose to augment the Hough transform by latent variables to enforce consistency of the votes. This is done by only allowing votes that agree on the values of the latent variables to support a single hypothesis. We discriminatively learn the optimal assignments of the training data to an arbitrary latent space to improve object detection performance. To this end, starting from a random assignment, the training examples are reassigned by optimizing an objective function that approximates the average precision on the training set. In order to make the optimization feasible, the objective function exploits the linearity of voting approaches. Further, we extend the concept that training instances can only be assigned to a single latent value. In particular, we let training examples assume multiple values and further allow these associations to be weighted, i.e. modeling the uncertainty involved in assigning a training example to them. This generalization makes the latent Hough transform robust with respect to the number of quantized latent values and we believe that the same is applicable when learning latent variable models in other domains.

Experiments on the Leuven cars [10] and the PASCAL VOC 2007 benchmark [16] show that our latent Hough transform approach significantly outperforms the standard Hough transform. We also compare our method to other baselines with unsupervised clustering of the training data or by manual annotations. In our experiments, we empirically demonstrate the superior performance of our latent approach over these baselines. The proposed method performs better than the best Hough transform based methods. And it even outperforms state-of-the-art detectors on some categories of the PASCAL VOC 2007.

2 Related Work

A number of previous works have investigated the idea of using object properties to group training data. For instance, the training data is clustered according to the aspect ratios of the bounding boxes in [8]. Other grouping criteria like user-annotated silhouettes [14], viewpoints [13, 17–19], height [21], and pose [11, 20] have been considered as well. To enforce consistency, the features vote for objects' depth in addition to locations in [22], the close-by features are grouped in [23] to vote together. In [24], two subtypes are trained for each part by mirroring the training data. The location of each part with respect to its parent joint is also used in [25] to train sub-types. Instead of grouping the training examples, [26] train a model for every single positive instance in the training data. While all of these works divide the training data into disjoint groups, [27] proposes a generative clustering approach that allows overlapping groups.

Latent variable models have been successfully used in numerous areas of computer vision and machine learning to deal with unobserved variables in training data, e.g. [28–31]. Most related to our work, [8, 32] learn a latent mixture model for object detection by discriminatively grouping the training examples and training a part model consisting of a root filter and a set of (hierarchical) parts for each group. In contrast to these works, our approach is a non-parametric voting based method where we assume the parts are given in the form of a shared vocabulary. As has been shown in previous works [17, 33], the shared vocabulary allows better generalization when learning a model with few examples as it makes use of the training data much more effectively. Given this vocabulary, we aim at learning latent groupings of the votes, i.e. training patches, which lead to consistent models and improve detection accuracy. The advantage of this approach is that we need to train the parts only once and not re-train them from scratch as in [8, 32].

Several approaches have been proposed for training the parts vocabulary. Generative clustering of the interest points is used in [4, 12, 34] as the codebook whereas [5, 13] discriminatively train a codebook by optimizing the classification and localization performance of image patches. Discriminative learning of weights for the codebook has been addressed before as well. In [34], a weight is learned for each entry in a Max-Margin framework. [35] introduced a kernel to measure similarity between two images and used it as a kernel to train weights with an SVM classifier. Although increasing the detection performance, those works differ from our approach in that they train weights within a group.

Detecting consistent peaks in the Hough-space for localization has been the subject of many investigations as well. While Leibe et al. [4] used a mean-shift mode estimation for accurate peak localizations, [36] utilize an iterative procedure to "demist" the voting space by removing the improbable votes from the voting space. Barinova et al. [37] pose the detection as a labeling problem where each feature can belong to only a single hypothesis and propose an iterative greedy optimization by detecting the maximum scoring hypothesis and removing its votes from the voting space.

3 Detection with the Hough Transform

In this section, we briefly describe Hough transform based object detection approaches that learn a codebook of voting elements, which can be some image features [4] or simply dense image patches [5]. Since our method does not require any retraining of the codebook, we refer for the details of the codebook learning to the corresponding works. Having such a codebook, the voting elements of an image $I_i \in \mathcal{I}$ are extracted and matched against the codebook to cast weighted votes $V(h|I_i)$ for an object hypothesis h, which encodes the position and scale of the object in the image.

For a given object hypothesis $h \in \mathcal{H}$, the score of h is determined by the sum of votes that support the hypothesis: $S(h) = \sum_i V(h|I_i)$. In fact , the accumulated weights of the votes that agree on the location and scale of the object. For detecting multiple objects, following the probabilistic approach of [37], the maximum scoring hypothesis is localized and its supporting votes are removed from the voting space. This process is iterated until the desired number of objects are detected or a confidence threshold is reached.

Since in an Implicit Shape Model [4], the votes are estimated from training patches in a non-parametric way and the score of each hypothesis is linear in the votes, S can be written as sum of votes from training instances $t \in \mathcal{T}$:

$$S(h) = \sum_{t \in \mathcal{T}} \sum_i V(h|t, I_i). \tag{1}$$

where $V(h|t, I_i)$ denotes the votes of element I_i from training image t. Although the votes originating from a single training example are always consistent, this formulation accumulates the votes over all training examples even if they are inconsistent, e.g., in pose or shape. In the next section, we show how one can use latent variables to enforce consistency among votes.

4 Latent Hough Transform (LHT)

In detecting objects with the latent Hough transform, we augment the hypothesis space by a latent space $\mathcal{Z}$ to enforce consistency of the votes in some latent properties $z \in \mathcal{Z}$. The score of a hypothesis in the augmented space can be determined as

$$S(h) = \max_{z \in \mathcal{Z}} \sum_{t \in \mathcal{T}} \sum_i V(h, z|t, I_i) \tag{2}$$

where $V(h, z|t, I_i)$ are the votes of an image element I_i from training image t to the augmented latent space $\mathcal{H} \times \mathcal{Z}$. For instance, $\mathcal{Z}$ can be quantized viewpoints of an object. Voting in this augmented space allows only votes that are consistent in viewpoint to support a single hypothesis.

Similar to other latent variable models [8, 29, 30, 32], one can associate each training image t to a single latent assignment z. This association groups the training data into $|\mathcal{Z}|$ disjoint groups. Note that the number of these groups is limited by the size of training data $|\mathcal{T}|$.

The grouping of the training data by latent assignments can be represented by a binary $|\mathcal{Z}| \times |\mathcal{T}|$ matrix, which we denote by W and refer to as *latent matrix*. The elements of W are denoted by $w_{z,t}$, where $w_{z,t} \in \{0,1\}$ and $\sum_z w_{z,t} = 1, \forall t \in \mathcal{T}$. Observe that every W that satisfies these constraints defines a grouping of the training data. Given a latent matrix W, we can rewrite the hypothesis score as

$$S(h,W) = \max_{z \in \mathcal{Z}} \sum_{t \in \mathcal{T}} w_{z,t} V(h|t), \quad \text{where} \quad V(h|t) = \sum_i V(h|t, I_i). \tag{3}$$

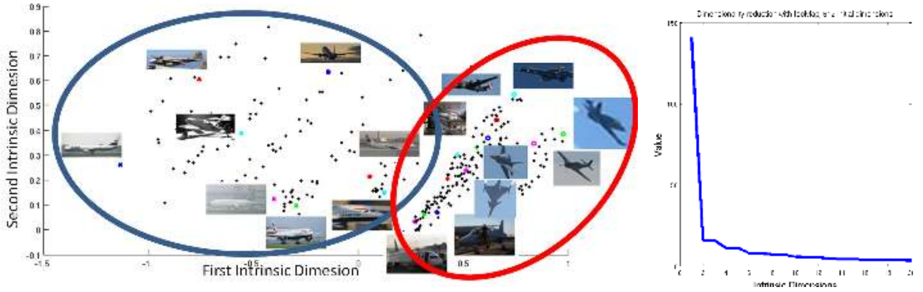
The term $V(h|t)$ is the sum of the votes originated from the training example t. This term will be very important while learning the optimal W as we will discuss in Sec. 5.

4.1 Generalized Latent Assignments

The association of the training data to a single latent value z does not make use of the training data effectively. We therefore generalize the latent assignments of the training data by letting a training image assume multiple latent values. To this end, we relax the constraints on W and allow $w_{z,t}$ to be real-valued in $[0,1]$ and non-zero for more than a single assignment z. This can be motivated by the uncertainties of the assignments for the training examples. In particular, the latent space can be continuous and, with an increasing quantization of $\mathcal{Z}$, two elements of $\mathcal{Z}$ can become very similar and thus need to “share” training examples. As we show in our experiments, this generalization makes the latent Hough transform less sensitive to the number of quantizations, i.e., $|\mathcal{Z}|$.

4.2 Special Cases of the Latent Matrix

The most basic special case of the latent matrix is when $|\mathcal{Z}| = 1$ and $w_{z,t} = 1, \forall t \in \mathcal{T}$ which is equivalent to the original Hough transform formulation in Eq.(1). The splitting of the training data by manual annotations or unsupervised clustering are also other special cases of the latent matrix where each row of the matrix represents one cluster. For splitting the training data, we have considered manual annotations of the viewpoints and two popular methods for clustering. Namely, agglomerative clustering of the training imnstances based on their similarity and k-means clustering of the aspect ratios of ground truth bounding boxes. Similar to [35, 13], we define the similarity of two training hypotheses as the χ^2 distance of their occurrence histograms. Groupings of the training data based on the similarity measure and manual view annotations are visualized in Figs. 1(a) and 3(b) respectively. As shown in these figures, the groupings based on annotations or clustering might be visually very meaningful. However, as we illustrate in our experiments the visual similarity alone may not be optimal for the detection. In addition, it is not clear what similarity measure to choose for grouping and how to quantize it. These problems underline the importance of learning the optimal latent matrix for detection.



(a) (b)

Fig. 1. (a) Visualization of the χ^2 similarity metric using Isomap. The two ellipses show the clustering of the training instances of the 'Aeroplane‘ category of PASCAL VOC 2007. As can be seen the clustering splits the data into very meaningful groups. (b) The visualization is accurate since the first two dimensions cover most of the variation.

5 Discriminative Learning of the Latent Matrix

We formulate the problem of learning the optimal latent matrix as the optimization problem

$$\hat{W} = \arg\max_{W} O(W, R). \tag{4}$$

where $R = \{(h, y)\}$ denotes the set of hypotheses h and their labels $y \in \{0, 1\}$, and $O(W, R)$ is the objective function. A hypothesis is assigned the label $y = 1$ if it is a true positive and $y = 0$ otherwise. For each hypothesis h, we pre-compute the contribution of every training instance $t \in \mathcal{T}$ to that hypothesis, i.e., $V(h|t)$ (3). It is actually the linearity of the Hough transform based approaches in Eq.(1) that allows for this pre-computation, which is essential for learning the latent matrix W in reasonable time.

As our objective function, we use the average precision measure on the validation set which is calculated as

$$O(W, R) = \frac{1}{\sum_j y_j} \sum_{k, y_k = 1} \frac{\sum_{j, y_j = 1} \mathbb{I}(h_k, h_j)}{\sum_j \mathbb{I}(h_k, h_j)} \tag{5}$$

where, for a given latent matrix W, $\mathbb{I}(h_k, h_j)$ indicates whether the score of hypothesis h_k is smaller than that of hypothesis h_j or not

$$\mathbb{I}(h_k, h_j) = \begin{cases} 1 & \text{if } S(h_j, W) \geq S(h_k, W) \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

Learning the latent matrix to optimize the detection performance is very challenging. First, the number of parameters to be learned is proportional to the number of training instances in the codebook which is usually very large. Another problem is that to be faithful to the greedy optimization in [37], with every update in the weights, one needs to run the detector on the whole validation dataset in order to measure the performance.

Algorithm 1 Interacting Simulated Annealing (ISA) [38] with cross-validation.

```
$\{R^s\} \leftarrow sample(R, maxNeg, maxPos)$
$\epsilon \leftarrow 0.6$
**for** $p = 1 \rightarrow n$ **do**
    $W_p \leftarrow$ *initialize W at random*
**end for**
**for** $epoch = 1 \rightarrow maxEpochs$ **do**
    $\{R^s\} \leftarrow sample(R, maxNeg, maxPos)$
    $c \leftarrow getMaxPerturbations(iter, epoch)$ //adaptively reduce perturbation
    **for** $iter = 1 \rightarrow maxIter$ **do**
        $W \leftarrow perturb(W, c)$ // perturb c elements of W at random
        **for** $p = 1 \rightarrow n$ **do**
            $o_p \leftarrow O(R^s, W_p)$
        **end for**
        $\beta \leftarrow 20 * (epoch * maxIter + iter)$
        $\{W\} \leftarrow selection(\{W\}, \{o\}, \beta, \epsilon)$
    **end for**
**end for**
```

In practice, we make an approximation and deal with the detection problem by sampling a sparse set of hypotheses from the validation set assuming that the positions of detections remain the same. To this end, we run the detector on the validation set once and collect a large number of hypotheses R. To increase the number of positive hypotheses, we also generate new object hypotheses from the positive training examples by mirroring and rescaling the training examples.

The objective function in Eq. 5 is non-convex and is not even continuous and thus it is not possible to optimize it with a gradient-based approach. For optimization, we used the Interacting Simulated Annealing (ISA) [38]. ISA is a particle-based global optimization method similar to the simulated annealing. Starting from an initial set of weights for n particles, it iteratively, i) perturbs the weights of selected particles ii) evaluates the objective value for each particle, exponentiates these values with the algorithm parameter β, and normalizes them to create a probability distribution. iii) randomly selects a number of particles using this distribution. This process is continued until a strong local optimum is reached. The perturbation of W at each iteration is done by selecting a random number of elements c (maximum 10) and changing their weights randomly. c is adaptively decreased at each epoch by the factor $\frac{1}{sqrt(epoch)}$.

Algorithm 1 gives an overview of the optimization with ISA. To avoid overfitting effects due to the large number of parameters to be estimated, we run a cross-validation loop inside the global optimization. For cross-validation, we use a random subset R^s of R at each epoch. In practice we have kept all the positive examples and 5% of the negatives for training at each epoch. We have also found well performing solutions, in detection performance, by running the optimization for a reasonable amount of time (a couple of hours on a single machine).

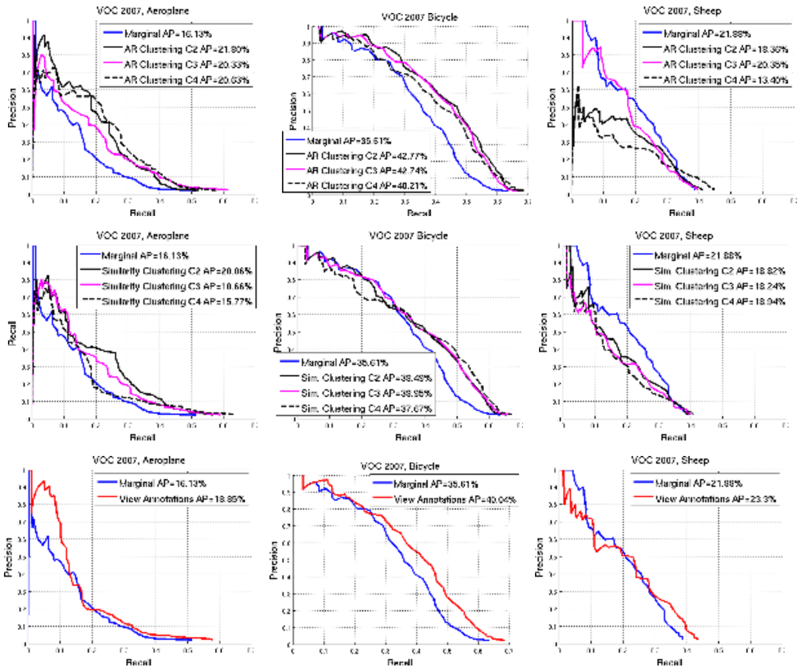
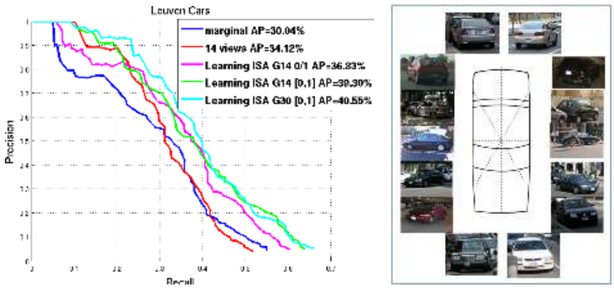


Fig. 2. This figure illustrates the result of using view annotations and unsupervised clustering for grouping training data of "aeroplane", "bicycle" and "sheep" categories of PASCAL VOC 2007. Groupings based on aspect ratio are shown in the first row, similarity clustering in the second, and the manual view annotations in the third row. Although clustering increases the performance for aeroplane, it is reducing it for the sheep. Also the AR clustering is performing better than similarity clustering for aeroplane and bicycle, yet clustering similarities leads to better results for the sheep. By using four clusters, the results are deteriorating in all three categories which is due to the insufficiency of the number of training data per cluster.

6 Experiments

We have evaluated our latent Hough transform on two popular datasets, namely the Leuven cars dataset [10] and the PASCAL VOC 2007 [16]. As a baseline for our experiments, we compare our approach with the marginalization over latent variables by voting only for locations ("Marginal"), unsupervised clusterings of aspect ratios ("AR clustering") and image similarities ("Similarity clustering"), and the manually annotated viewpoints ("View Annotations") provided for both Leuven cars and PASCAL VOC 2007.

In all our experiments, the codebook of the ISM is trained using the Hough forests [5] with 15 trees and the bounding boxes for a detection are estimated using backprojection. The trees are trained up to the maximum depth of 20 such that at least 10 occurrences are remained in every leaf. For performing detection on a test image, we have used the greedy optimization in [37]. The multi-scale detection was performed by doing detection on a dense scale pyramid with a $\frac{1}{\sqrt[4]{2}}$ resizing factor. In addition, instead of penalizing the larger hypotheses by



(a) (b) annotated views

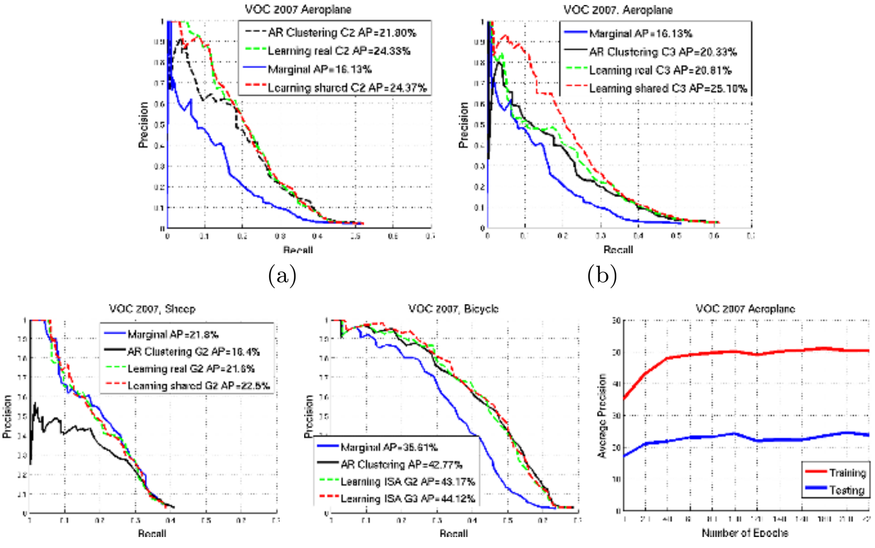
Fig. 3. (a) Performance comparison of our latent Hough transform with the baselines on the Leuven cars. As can be seen, in the red curve, AP is clearly increased by manually splitting the data to 14 views. By learning 14 latent groups, the performance is significantly improved over both baselines, the magenta curve. Learning the generalized latent matrix (green) and increasing the number of groups (cyan) improves the results further. (b) Examples of the 14 views manually annotated in the training data.

adding a negative bias as in [37], similar to [4], we allow larger deformations by increasing the standard deviation of the smoothing kernel proportional to the scale. The smoothing kernel is chosen as a gaussian with $\sigma = 1.25$ at scale one. In every test, 40 bounding boxes are detected and the hypothesis score in Eq.(2) is assigned as the confidence of every detection. According to [16], precision/recall curves and average precision measure (AP) were used for the evaluations.

Prior to learning the latent groups, we have collected a set of positive and negative detections by running the detector on the whole validation set of a category and detecting 100 bounding boxes from each image. The bounding boxes with more than 60% overlap with ground truth were considered as positive and the ones with less than 30% as negatives.

Leuven cars: For the Leuven cars dataset, 1471 cropped training images of cars are provided. The viewing angle is divided into 14 views and training images are annotated for 7 viewpoints. The training data of the other 7 viewpoints is obtained by mirroring the traning images, creating the total of 2942 training images annotated for 14 viewpoints. Prior to the training, all positive objects in the training instances rescaled to have the height of 70 pixels. In addition, for the background category, we are using the clutter set of Caltech 256 [39]. A third of positive images and 200 negative images and from each of which 250 patches are randomly sampled for training each tree. As the validation set for learning the latent groupings, the Amsterdam cars dataset [10] and the Graz02 cars [40] are used. The Leuven sequence [10] is used as the test set and the detection was performed on 12 scales starting from 2.7.

PASCAL VOC 2007: A seperate forest is trained for each category. The training is carried out by using all the positive examples and their mirrors in the "trainval" set of a category as the positive set and the images not containing the category as the negative set. The partial view annotations are ignored for the training. Similar to the cars, the positive training instances are cropped and normalized to have the maximum height or width of 100 pixels. For training



(a) (b)

(c) (d) (e)

Fig. 4. This figure shows the result of learning the latent matrix on three categories. By learning the latent matrix we can consistently outperform the clustering ("AR clustering") and the Hough transform baseline ("Marginal"). (a) When learning 2 latent groups, there is not much benefit in assigning training examples to multiple groups. (b) However, doing so already gives a benefit for learning three latent groups as it models the uncertainty in the assignments. (c-d) Results for two other categories. (e) Shows the comparison of the training and testing performance as a function of the number of epochs. Since the same training data is used for creating the ISM codebook and learning the latent matrix, the overall performance of the training is much better. Yet, the two curves correlate well and the training shows little overfitting.

each tree, 200 training images from the positive set and 200 from the negative set and from each of which 250 patches are sampled at random. The "trainval" set of a category was used as the validation set for learning the latent groupings. The performance of the method is evaluated on the "test" set. The multi-scale detection is done with 18 scales starting from 1.8.

To evaluate the benefits of the discriminative learning against unsupervised clustering and manual view annotations, we compared the results of learning on the Leuven cars and PASCAL VOC 2007 datasets. For a fair comparison, we train the Hough forests [13] for a category only once and without considering the view annotations or learned groupings. For the optimization with ISA, we used 500 particles and the number of epoch and iterations were both set to 40.

Figure 3 compares the performance of the learning with our baselines on the Leuven cars [10]. Disjoint groupings of the training data based on view annotations, improves the result by 4 AP percentage points. By learning the latent groups the performance improves by 6.8 and 2.7 points w.r.t the marginalization and manual view annotations respectively. Learning the generalized latent matrix improves the result further by about 2.5 points. By allowing more la-

VOC 2007	Aeroplane	Bicycle	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Din. Table	Dog	Horse	Motorbike	Person	Potted Plant	Sheep	Sofa	Train	TV/Mon.	mean AP
Our Approach																					
HT Marginal	16.1	35.6	2.9	3.3	20.4	15.8	25.5	7.5	10.9	37.2	10.3	3.2	28.6	34.2	4.5	19.2	21.9	10.3	9.8	43.4	18.03
HT + AR	21.8	42.7	11.4	10.2	19.6	19.1	25.4	6.0	6.3	38.2	7.6	6.1	30.5	39.0	4.9	20.5	18.4	10.8	16.9	41.3	19.83
HT + View	18.85	40.0	5.3	3.0	-	18.3	28.7	6.6	4.7	34.9	-	2.9	29.8	38.6	6.1	-	23.3	10.5	21.2	38.7	19.50
LHT Ours	**24.3**	**43.2**	**11.1**	**10.5**	**20.7**	**20.3**	**24.0**	**8.0**	**11.9**	**38.8**	**10.5**	**4.9**	**33.2**	**39.4**	**8.2**	**21.3**	**22.5**	**10.5**	**17.3**	**44.1**	**21.23**
Competing Approaches																					
HT ISK [35]	24.6	32.1	5.0	9.7	9.2	23.3	29.1	11.3	9.1	10.9	8.1	13.0	31.8	29.5	16.6	6.1	7.3	11.8	22.6	21.9	16.65
MKL [41]	37.6	47.8	15.3	15.2	21.9	50.7	50.6	30.0	17.3	33.0	22.5	21.5	51.2	45.5	23.3	12.4	23.9	28.5	45.3	48.5	31.10
Context [42]	53.1	52.7	18.1	13.5	30.7	53.9	43.5	40.3	17.7	31.9	28.0	29.5	52.9	56.6	44.2	12.6	36.2	28.7	50.5	40.7	36.77
LSVM [8]	29.0	54.6	0.6	13.4	26.2	39.4	46.4	16.1	16.3	16.5	24.5	5.0	43.6	37.8	35.0	8.8	17.3	21.6	34.0	39.0	26.26
VOC best	26.2	40.9	9.8	9.4	21.4	39.3	43.2	24.0	12.8	14.0	9.8	16.2	33.5	37.5	22.1	12.0	17.5	14.7	33.4	28.9	23.33

Table 1. Detection results on the PASCAL VOC 2007 dataset [16]. The first block compares the performance of the Hough transform without grouping (HT Marginal), aspect ratio clustering (HT + AR), view annotations (HT+View), and our proposed latent Hough transform (LHT Ours). As can be seen the clustering improves the results for 14 categories over the marginalization but reduces it for the other 6. Yet, by learning latent groups we outperform all three baselines on most categories and perform similar or slightly worse (red) on others. The comparison to the state-of-the-art approaches is shown in the second block. We outperform the best previously published voting-based approach (ISK [35]) in mAP. Our performance is competitive on many categories with the latent part model of [8] and is state-of-the-art on two categories (green).

tent assignments, one can learn finer groupings of the data and increase the performance by 10 AP points compared to the marginalization baseline.

The detection performance with the two clusterings and view annotations on three distinct categories aeroplane, bicycle and sheep of VOC'07 dataset are summarized in Fig. 2. As can be seen, although grouping training examples may improve performance, this improvement is very much dependent on the grouping criteria, the category and the number of training data per group. For example, in detecting airplanes, although using two clusters improves performance, using more clusters impairs the results. As another example, in detecting sheep, the marginalization is clearly outperforming clustering with two clusters. The clustering or the view annotations do not lead to optimal groupings and even finding the well performing ones requires plenty of trial and error.

In contrast to clustering, one can discriminatively learn optimal groupings by treating them as latent variables. Figure 4 compares the performance of the learning with the clustering and marginalization. By learning the latent groups, we outperform clustering baselines. In addition, the learning is not sensitive to selecting the right number of latent groups as it, unlike clustering, shares training examples between groups. Table 1, gives the full comparison of our latent Hough transform method with our baselines and other competing approaches. Some qualitative results on the VOC'07 dataset are shown in Fig. 5.

7 Conclusions

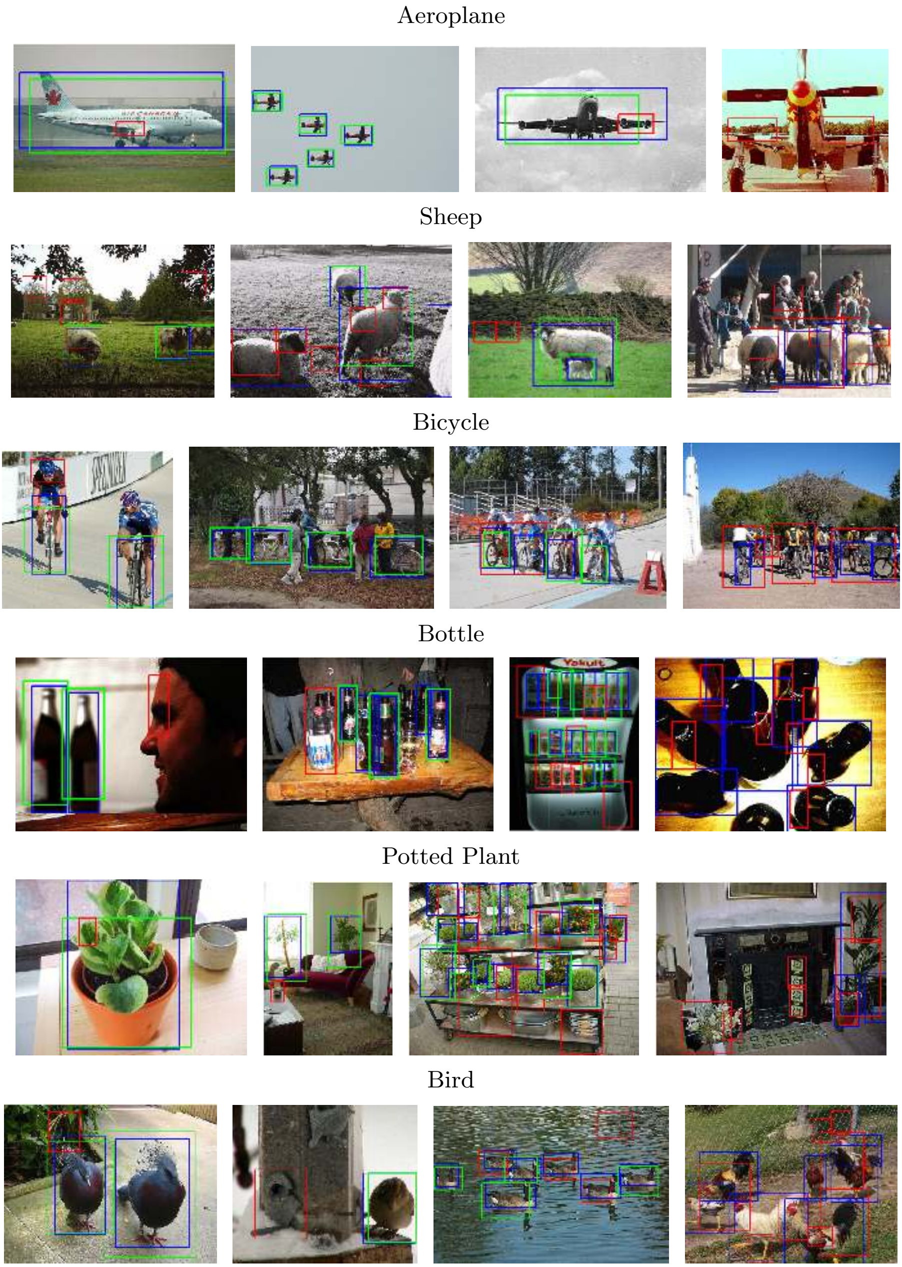
In this paper, we have introduced the Latent Hough Transform (LHT) to enforce consistency among votes that support an object hypothesis. To this end, we have augmented the Hough space with latent variables and discriminatively learned the optimal latent assignments of the training data for object detection. Further, to add robustness to the number of quantizations, we have generalized the latent variable model by allowing the training instances to have multiple weighted assignments and have shown how the previous grouping approaches can be cast as special cases of our model. In the future, it would be interesting to use the latent formulation in a more general context e.g., learning a multi-class LHT model or learning latent transformations of the votes for better detection accuracy.

Acknowledgements: This project has been funded in parts by Toyota Motor Corporation/Toyota Motor Europe, and the European projects RADHAR (FP7 ICT 248873) and IURO (FP7 ICT 248314).

References

1. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: CVPR. (2005)
2. Ferrari, V., Fevrier, L., Jurie, F., Schmid, C.: Groups of adjacent contour segments for object detection. TPAMI **30** (2008) 36–51
3. Ojala, T., Pietikinen, M., Menp, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. TPAMI **24** (2002) 971–987
4. Leibe, B., Leonardis, A., Schiele, B.: Robust object detection with interleaved categorization and segmentation. IJCV **77** (2008) 259–289
5. Gall, J., Yao, A., Razavi, N., Van Gool, L., Lempitsky, V.: Hough forests for object detection, tracking, and action recognition. TPAMI **33** (2011) 2188–2202
6. Fergus, R., Perona, P., Zisserman, A.: Object class recognition by unsupervised scale-invariant learning. In: CVPR. (2003)
7. Hoiem, D., Rother, C., Winn, J.: 3d layoutcrf for multi-view object class recognition and segmentation. In: CVPR. (2007)
8. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part based models. TPAMI **32** (2009) 1627 – 1645
9. Bergtholdt, M., Kappes, J., Schmidt, S., Schnörr, C.: A study of parts-based object class detection using complete graphs. IJCV **87** (2010) 93–117
10. Leibe, B., Cornelis, N., Cornelis, K., Van Gool, L.: Dynamic 3d scene analysis from a moving vehicle. In: CVPR. (2007)
11. Seemann, E., Leibe, B., , Schiele, B.: Multi-aspect detection of articulated objects. In: CVPR. (2006)
12. Seemann, E., Fritz, M., Schiele, B.: Towards robust pedestrian detection in crowded image sequences. In: CVPR. (2007)
13. Razavi, N., Gall, J., Van Gool, L.: Backprojection revisited: Scalable multi-view object detection and similarity metrics for detections. In: ECCV. (2010)
14. Marszałek, M., Schmid, C.: Accurate object localization with shape masks. In: CVPR. (2007)
15. Stephens, R.: Probabilistic approach to the hough transform. Image and vision computing **9** (1991) 66–71
16. Everingham, M., Van Gool, L., Williams, C., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV **88** (2010) 303–338
17. Torralba, A., Murphy, K.P., Freeman, W.T.: Sharing features: efficient boosting procedures for multiclass object detection. In: CVPR. (2004)

18. Thomas, A., Ferrari, V., Leibe, B., Tuytelaars, T., Schiele, B., , Van Gool, L.: Towards multi-view object class detection. In: CVPR. (2006)
19. Ozuysal, M., Lepetit, V., Fua, P.: Pose estimation for category specific multiview object localization. In: CVPR. (2009)
20. Dantone, M., Gall, J., Fanelli, G., Van Gool, L.: Real-time facial feature detection using conditional regression forests. In: CVPR. (2012)
21. Sun, M., Kohli, P., Shotton, J.: Conditional regression forests for human pose estimation, CVPR (2012)
22. Sun, M., Bradski, G., Xu, B.X., Savarese, S.: Depth-encoded hough voting for coherent object detection, pose estimation, and shape recovery. In: ECCV. (2010)
23. Yarlagadda, P., Monroy, A., Ommer, B.: Voting by grouping dependent parts. ECCV (2010)
24. Girshick, R.B., Felzenszwalb, P.F., McAllester, D.: Object detection with grammar models. In: NIPS. (2011)
25. Yang, Y., Ramanan, D.: Articulated pose estimation with flexible mixtures-of-parts. In: CVPR. (2011)
26. Malisiewicz, T., Gupta, A., Efros, A.A.: Ensemble of exemplar-svms for object detection and beyond. In: ICCV. (2011)
27. Torsello, A., Bulò, S., Pelillo, M.: Beyond partitions: Allowing overlapping groups in pairwise clustering. In: ICPR. (2008)
28. Hofmann, T.: Unsupervised learning by probabilistic latent semantic analysis. Machine Learning **42** (2001) 177–196
29. Farhadi, A., Tabrizi, M., Endres, I., Forsyth, D.: A latent model of discriminative aspect. In: ICCV. (2009)
30. Wang, Y., Mori, G.: A discriminative latent model of object classes and attributes. In: ECCV. (2010)
31. Bilen, H., Namboodiri, V., Van Gool, L.: Object and action classification with latent variables. In: BMVC. (2011)
32. Zhu, L., Chen, Y., Yuille, A., Freeman, W.: Latent hierarchical structural learning for object detection. In: CVPR. (2010)
33. Razavi, N., Gall, J., Van Gool, L.: Scalable multiclass object detection. In: CVPR. (2011)
34. Maji, S., Malik, J.: Object detection using a max-margin hough transform. In: CVPR. (2009)
35. Zhang, Y., Chen, T.: Implicit shape kernel for discriminative learning of the hough transform detector. In: BMVC. (2010)
36. Woodford, O., Pham, M., Maki, A., Perbet, F., Stenger, B.: Demisting the hough transform. In: BMVC. (2011)
37. Barinova, O., Lempitsky, V., Kohli, P.: On detection of multiple object instances using hough transforms. In: CVPR. (2010)
38. Gall, J., Potthoff, J., Schnörr, C., Rosenhahn, B., Seidel, H.: Interacting and annealing particle filters: Mathematics and a recipe for applications. Journal of Mathematical Imaging and Vision **28** (2007) 1–18
39. Griffin, G., Holub, A., Perona, P.: Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology (2007)
40. Opelt, A., Pinz, A., Fussenegger, M., Auer, P.: Generic object recognition with boosting. TPAMI **28** (2006) 416–431
41. Vedaldi, A., Gulshan, V., Varma, M., Zisserman, A.: Multiple kernels for object detection. In: ICCV. (2009)
42. Song, Z., Chen, Q., Huang, Z., Hua, Y., Yan, S.: Contextualizing object detection and classification. In: CVPR. (2011)



Aeroplane

Sheep

Bicycle

Bottle

Potted Plant

Bird

Fig. 5. Some qualitative results on the test set of PASCAL VOC 2007 database. Ground-truth bounding boxes are in blue, correctly detected boxes in green and false positives in red.