On Detection of Multiple Object Instances using Hough Transforms

Olga Barinova
Moscow State University*

Victor Lempitsky
University of Oxford*

Pushmeet Kohli
Microsoft Research Cambridge

Abstract

To detect multiple objects of interest, the methods based on Hough transform use non-maxima supression or mode seeking in order to locate and to distinguish peaks in Hough images. Such postprocessing requires tuning of extra parameters and is often fragile, especially when objects of interest tend to be closely located. In the paper, we develop a new probabilistic framework that is in many ways related to Hough transform, sharing its simplicity and wide applicability. At the same time, the framework bypasses the problem of multiple peaks identification in Hough images, and permits detection of multiple objects without invoking non-maximum suppression heuristics. As a result, the experiments demonstrate a significant improvement in detection accuracy both for the classical task of straight line detection and for a more modern category-level (pedestrian) detection problem.

1. Hough Transform in Object Detection

The Hough transform [12] is one of the classical computer vision techniques which dates 50 years back. It was initially suggested as a method for line detection in edge maps of images but was then extended to detect general low-parametric objects such as circles [3]. In recent years, Hough-based methods were successful adapted to the problem of part-based category-level object detection where they have obtained state-of-the-art results for some popular datasets [14, 15, 9, 10, 17, 5].

Both the classical Hough transform and its more modern variants proceed by converting the input image into a new representation called the *Hough image* which lives in a domain called the *Hough space* (Figure 1). Each point in the Hough space corresponds to a hypothesis about the object of interest being present in the original image at a particular location and configuration.

Any Hough transform based method essentially works by splitting the input image into a set of *voting elements*. Each such element votes for the hypotheses that might have generated this element. For instance, a feature that fires on faces might vote for the presence of a person's centroid (torso) in location just below it. Of course, voting elements do not provide evidence for the exact localization and thus their votes are distributed over many different hypothesis in the Hough space. Large values of the vote are given to hypotheses that might have generated the voting element with high probability. The votes from different voting elements pixels are added together into a Hough image. The objects of interest are then detected as peaks in the Hough image, with the height of the peak providing the confidence of the detection.

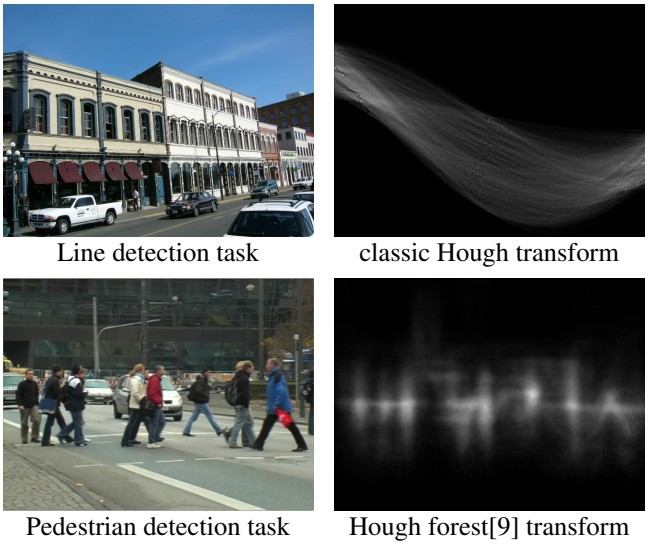


Figure 1. Variants of Hough transform render the detection tasks (left) into the tasks of peaks identification in Hough images (right). As can be seen, in the presence of multiple close objects identifying the peaks in Hough images is a highly non-trivial problem in itself. The probabilistic framework developed in this paper addresses this problem without invoking non-maxima suppression or mode seeking heuristics.

The popularity of the Hough-based approach to object detection stems from its flexibility (e.g. the primary voting elements are not restricted to be edge pixels, but can include interest points [14], image patches [9, 17], or image regions [10]). Another attractive property is the simplicity of the learning procedure. Given a set of images annotated with the location of objects of interest, learning essentially involves construction of the appearance codebook (or voting elements). The Hough vote for each codebook entry is then simply obtained from the training data by observing the distribution of object parameters (e.g. centroid displacements)

*The first two authors were with Microsoft Research through the initial stages of the work and are currently supported by Microsoft Research programs in Russia. Victor Lempitsky is also supported by EU under ERC grant VisRec no. 228180.

which generates the entry. This simple additive nature of the Hough transform makes the detection process robust to deformation, imaging noise and many kinds of occlusion.

In spite of all the advantages mentioned above, the Hough transform still has a major flaw in that it lacks a consistent probabilistic model. This leads to problems of both theoretical and practical nature. From the theoretical viewpoint, Hough-based detection does not allow hypotheses to *explain away* the voting elements. To give an example, consider the situation when the maximum in the Hough image corresponds to a correctly detected object. Consider also the voting elements that were generated by this object. These elements are likely to cast strong votes for the detected object, but they are also likely to cast votes for other hypotheses, and the strength of those spurious votes is not inhibited in any way by the fact that a good hypothesis explaining the voting element already exists. In practice, this means that various non-maxima suppression (NMS) heuristics have to be used in real detection scenario to localize peaks in the Hough image. These heuristics typically involve specification and tuning of several parameters.

The goal of the paper is to introduce a new detection method similar to the Hough transform. More precisely, we introduce a new framework, which has a probabilistic nature, and shares most of the virtues of the Hough transform. Notably, the new model can reuse the training procedures and the vote representations developed in previous works on Hough-based detection, such as Implicit Shape models[14] or Hough forests[9]. At the same time, the new approach bears some additional important advantages over the Hough Transform:

- It performs multiple objects detection via an energy optimization (MAP-inference in the probabilistic model), in contrast to heuristic peak location followed by non-maxima suppression used in vanilla Hough Transform.
- Experimental results show that it results in a better accuracy for images containing multiple objects of interest.
- Its probabilistic nature means that our model is easy to integrate with other approaches, including e.g. modeling of higher-level geometric constraints on the location of the objects.

In some sense, the proposed framework is related to the approaches

The disadvantage of the suggested approach compared to the traditional Hough transform is the increased computation time, which however, is still very competitive compared to many other modern detection techniques.

The remainder of the paper if organized as follows. We start by reviewing the Hough transform from the probabilistic viewpoint and introducing our model in section 2. We then discuss how MAP-inference in our model can be performed. We proceed to the experimental comparison of our model with the vanilla Hough transform method. This evaluation is performed on the traditional Hough task of line detection in images, as well as the more modern task of category-level (pedestrian) detection. Finally, we discuss the relation of our approach to prior art and how our framework can be extended and integrated with other approaches.

2. The framework

2.1. Analysis of the Hough transform.

We start by introducing our notation and then analyze the probabilistic interpretation of Hough-based detection. Let us assume that the image observations come in the form of N *voting elements*, which throughout the paper we will index with the letter i. These elements may correspond to e.g. pixels in the edge map (in the case of line detection), or to interest points (in the case of the implicit shape model-like framework). We also assume a Hough space $\mathcal{H}$, where each point $h \in \mathcal{H}$ corresponds to a hypothesis about the presence of an object of interest (e.g. a line, a pedestrian) in a particular location/configuration. The detection task can then be formulated as finding the finite subset of the Hough space that corresponds to objects that are actually present in the image. To formalize this, for each hypothesis $h \in \mathcal{H}$, we introduce the binary random variable y_h that takes the value 1 if the hypothesis actually corresponds to the real object and the value 0 otherwise.

The Hough transform does the detection by considering each voting element i independently and reasoning which object h might have generated it. To formalize this reasoning, we introduce a random variable x_i that takes a value in the augmented Hough space $\mathcal{H}' = \mathcal{H} \cup 0$. The assignment $x_i = h \in \mathcal{H}$ implies that the voting element i is generated by the object h, while $x_i = 0$ implies that element i comes from background clutter and is not part of any object of interest. We can now consider *votes* as (pseudo)-densities $V(x_i = h|I_i)$ in the Hough space conditioned on the descriptor I_i of the voting element i. The descriptor here might include the geometric position of the voting element in the (scale)space and/or the local image appearence. These conditional (pseudo)-densities are then added and the peaks of the resulting sum are considered as valid hypothesis.

Summing up the pseudo-densities does not have any probabilistic meaning. While one may attempt to interpret the votes as the logarithms of the likelihoods $V(x_i = h|I_i) = \log p(x_i = h|I_i)$, a careful inspection reveals that this interpretation is not viable due to the following reasons: firstly, unlike log-probability densities, the votes are typically positive and span a limited range, and secondly, in learning-based frameworks the votes are learned as non-parametric probability densities and not log-likelihoods.

Ignoring the above-mentioned, we can still view Hough votes as logarithms of $p(x_i = h|I_i)$. Then their summation corresponds to assuming that distributions over hypotheses generating voting elements are independent, i.e. $\forall i, j : p(x_i|I_i) \perp p(x_j|I_j)$. This **independence assumption** is clearly extremely crude. For instance, if voting elements i and j are adjacent in the image, then there is obviously a strong correlation between the hypothesis they come from, namely that they are very likely to be generated from the same object (or background clutter). Non-maxima suppression which is routinely performed within Hough transform can be regarded as a trick that compensates for the limitations of this independence assumption. As we will show later, the need for the non-maxima suppression goes away if we do not make the assumption.

2.2. The probabilistic framework

Hough voting builds on the fact that pseudo-likelihoods (votes) $V(x_i = h|I_i)$ can be easily defined or learned from the data. However, rather than fusing all the votes in a principled way, Hough transform takes the easiest and fastest path and simply sums them. We now describe how it is possible to overcome these problems.

Our framework departs from the Hough voting framework in the way how the votes $V(x_i = h|I_i)$ are fused. Rather than summing these votes, we model the joint distribution over all the random variables $\mathbf{x} = \{x_i\}$ and $\mathbf{y} = \{y_h\}$ in a probabilistic way, so that we can determine their values via the (MAP-) inference process. Thus, we are interested in modeling the joint posterior of $\mathbf{x}$ and $\mathbf{y}$ given image $\mathbf{I}$, where by image we mean the collection of the voting elements. Applying the Bayes theorem then gives:

$$p(\mathbf{x}, \mathbf{y}|\mathbf{I}) \propto p(\mathbf{I}|\mathbf{x}, \mathbf{y}) \cdot p(\mathbf{x}, \mathbf{y}) \tag{1}$$

We now focus on the likelihood and prior terms separately.

Likelihood Term We make a different independence assumption to handle the likelihood term $p(\mathbf{I}|\mathbf{x}, \mathbf{y})$. We assume that given the existing objects $\mathbf{y}$ and the hypotheses assignments $\mathbf{x}$, the distributions of the appearances of voting elements are independent, i.e.:

$$p(\mathbf{I}|\mathbf{x}, \mathbf{y}) = \prod_{i=1}^{N} p(I_i|\mathbf{x}, \mathbf{y}). \tag{2}$$

Furthermore, we assume that the descriptor I_i of the voting element i depends only on the object assignment x_i, and is conditionally independent of the assignments of the remaining voting elements and the existence of all other objects in the image. Thus, we get:

$$p(\mathbf{I}|\mathbf{x}, \mathbf{y}) = \prod_{i=1}^{N} p(I_i|x_i) \tag{3}$$

At a first glance, these assumptions may seem quite crude as the appearance of the element is assumed to be dependent only on the hypothesis x_i and conditionally independent from other voting elements and hypotheses. However, this dependence still may encode the relative positioning of the element i and the object corresponding to the hypothesis x_i. For instance, in the case of car detection, the expression $p(I_i|x_i)$ may model the appearance of the voting element (e.g. interest point) as a random variable dependent on the part of the car it comes from. We conclude the derivation of the likelihood part, by applying the Bayes theorem once more and then omitting the terms that are constant for the given image:

$$p(\mathbf{I}|\mathbf{x}, \mathbf{y}) = \prod_{i=1}^{N} p(I_i|x_i) \propto \prod_{i=1}^{N} \frac{p(x_i|I_i)}{p(x_i)}\,. \tag{4}$$

As a result, (1) can be rewritten as a following product:

$$p(\mathbf{x}, \mathbf{y}|\mathbf{I}) \propto \prod_{i=1}^{N} p(x_i|I_i) \cdot \frac{p(\mathbf{x}, \mathbf{y})}{\prod_{i=1}^{N} p(x_i)}. \tag{5}$$

Our expression for the likelihood is very similar to the (multiplicative) Hough transform as the data-dependent term turns out to be the product of terms $p(x_i|I_i)$, which are related to Hough votes.

Prior terms Before formulating the prior distribution $p(\mathbf{x}, \mathbf{y})$, we should note that that not all configurations $(\mathbf{x}, \mathbf{y})$ are valid. If a voting element i is assigned to a non-background hypothesis h ($x_i = h$) then the hypothesis h must correspond to an existing object, i.e. y_h must be 1. Thus, the configuration is valid if and only if $y_{x_i} = 1, \forall x_i$. To avoid treating the background assignments $x_i = 0$ as a special case, we introduce the background hypothesis variable y_0, which is always set to 1. As a result the consistency of the configuration $(\mathbf{x}, \mathbf{y})$ may be expressed by the hard constraint $\prod_{i=1}^{N} y_{x_i} = 1$.

Given that the configuration $(\mathbf{x}, \mathbf{y})$ is valid, we now assume that our prior factorizes into products of priors on $\mathbf{y}$ and individual x_i:

$$p(\mathbf{x}, \mathbf{y}) = Z_1 \prod_{i=1}^{N} y_{x_i} \cdot p(\mathbf{y}) \cdot \prod_{i=1}^{N} p(x_i) \tag{6}$$

In this work we also focus on a very general prior on $\mathbf{y}$ (*Occam razor* or *MDL prior*) that simply penalizes the number of the active hypotheses $\sum_{h \in \mathcal{H}} y_h$, preferring explanations of the scene with as little objects as possible:

$$p(\mathbf{y}) = Z_2 \exp\left(-\lambda \sum_{h \in \mathcal{H}} y_h\right) = C_2 \prod_{h \in \mathcal{H}} \exp\left(-\lambda y_h\right)\,. \tag{7}$$

In (6-7), Z_1 and Z_2 are the normalization constants.

As a result of substituting (6) and (7) into (5), we get the final expression for the posterior:

$$p(\mathbf{x},\mathbf{y}|\mathbf{I}) \propto \prod_{i=1}^{N} p(x_i|I_i) \cdot \prod_{i=1}^{N} y_{x_i} \cdot \prod_{h\in\mathcal{H}} \exp\left(-\lambda y_h\right) \quad (8)$$

Note, that there might be several other approaches to choosing the prior distribution $p(\mathbf{x},\mathbf{y})$. E.g., it may be computationally feasible to impose Potts prior on $\mathbf{x}$ ("if a voting element i is assigned to a hypothesis h, then the adjacent voting element j is also likely to be assigned to a hypothesis h"). The use of the Potts prior, however, is known to be detrimental for thin objects, e.g. lines.

It is also easy to introduce the standard non-maxima suppression via the overlap criterion into our framework. For this, one simply needs to define a prior that assigns zero probability to all configurations $\mathbf{y}$ where there exists a pair of enabled hypotheses with the bounding boxes overlapping too much. However, in our experiments, we refrain from using such a prior. This allows us to contrast our approach against traditional non-maxima suppression, and also to detect strongly overlapping object instances.

3. Inference

Log-posterior maximization and facility location. In the paper, we focus on computing the maximum-a-posteriori (MAP) configurations (MAP-inference) under the probability model (8). By taking the logarithm of (8), the MAP-inference in our model is rendered as the maximization problem for the following log-posterior function:

$$E(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{N} u_i(x_i) + \sum_{h\in\mathcal{H}} v_h(y_h) + \sum_{h\in\mathcal{H}}\sum_{i=1}^{N} w_{ih}(x_i,y_h)\,, \quad (9)$$

where $u_i(x_i) = \log p(x_i|I_i)$, $v_h(y_h) = -\lambda y_h$, and $w_{ih}(x_i,y_h) = -\infty$ if $x_i = h$ and $y_h = 0$ and $w_{ih}(x_i,y_h) = 0$ otherwise.

The maximization of (9) is known in the operation research community as the *facility location* problem. The problem is NP-hard, but approximations have been studied extensively. For instance, one can notice that (9) is "bipartite", and one can choose the optimal value of x_i independently, if the configuration of $\mathbf{y}$ is given. Thus, the log-posterior (9) can be represented only as the function of the hypotheses variables $\mathbf{y}$, after the $\mathbf{x}$ variables are "maximized out":

$$E_y(\mathbf{y}) = \max_{\mathbf{x}} E(\mathbf{x},\mathbf{y}) =$$
$$\sum_{h\in\mathcal{H}} -\lambda y_h + \sum_{i=1}^{N} \max\left(\max_{h:y_h=1} u_i(h), u_i(0)\right) \quad (10)$$

Sparse set of hypotheses. When tackling the MAP-inference in our model, we tried different algorithms. Thus, we considered loopy belief-propagation [18] in the bipartite graph defined by (9). The special form of the pairwise terms permits a very compact message representation (the same as used in the affinity propagation [8]). We have also tried simulated annealing optimization for the binary-labelled function (10). In practice, one problem with applying non-greedy algorithms is that the graphs for message-passing and the sizes of cliques to be handled by the simulated annealing are too large. To circumvent that, one can ran standard Hough voting, look for an excessive number (dozens to hundreds) of peaks in the Hough image, and restrict the Hough space $\mathcal{H}$ to those peaks. Furthermore, the majority of $p(x_i|I_i)$ are typically equal to or very close to zero, so that $u_i(x_i)$ are very close to $-\infty$. This permits to sparsify the graphs further.

While the non-greedy algorithms with sparsification heursitics discussed above work well, we have found that a simple greedy optimization provided about the same accuracy. There is a good theoretical reason for that, as the function (10) can be proven *submodular* (see [4]), and for submodular functions the provably optimal approximation bounds are achieved by greedy maximization [16]. The greedy algorithm starts with all y_h set to 0 and x_i set to 0 (background). Then, in each step t the algorithm switches on a hypothesis h^t (setting $y_{h^t} = 1$), simultaneously switching some of x_i to h^t (x_i is switched to h^t only if this increases the posterior). The hypothesis h^t is picked so that the biggest increase of the posterior is obtained.

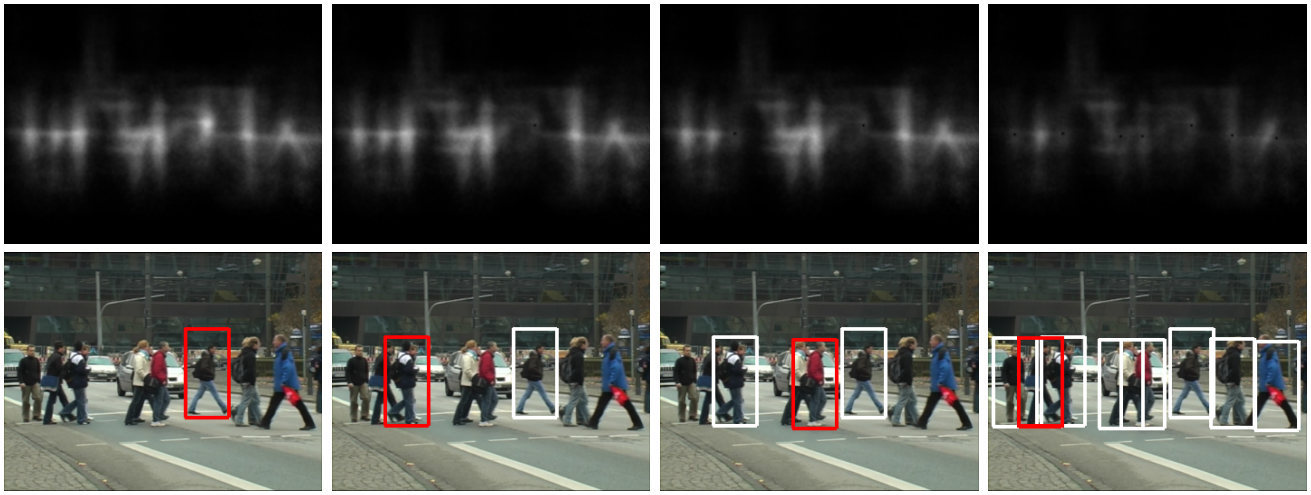
Dense set of hypotheses. Interestingly, the greedy algorithm does not require sparsification heuristics. How can such greedy-optimal hypothesis h^t be quickly identified? It turns out, that this can be done by performing Hough voting with the specific votes that are updated on each iteration. Thus, in each step t the voting element casts an (additive) vote:

$$V_i^t(h) = \max\left(\log P(x_i{=}h|I_i) - \log P(x_i{=}x_i^t|I_i), 0\right)\,, \quad (11)$$

where x_i^t denotes the hypothesis that the voting element i is assigned by the step t. Each vote thus encodes the potential increase of the log-posterior part for the element i should the hypothesis h be enabled.

The votes are accumulated into a Hough image $M^t(h) = \sum_{i=1}^{N} V_i^t(h)$. Then, the maximal value hypothesis $h^t = \mathrm{argmax}(M^t(h))$ is considered. If $M^t(h^t)$ is less or equal λ then the algorithm terminates, as the log-posterior (10) cannot be increased any further in a greedy way. Otherwise, y_{h^t} is set to 1 and each x_i is switched to h^t (i.e. x_i^{t+1} is set to h^t), provided that this increases the log-posterior.

Thus, the difference between the algorithm above and the traditional Hough transform is that Hough voting is performed here multiple times with dynamically changing



1st iteration 2nd iteration 3rd iteration 8th iteration

Figure 2. Greedy MAP-inference in our model for pedestrian-detection example from Figure 1. For each iteration, we give the Hough image M^t (top) and highlight in red the detection corresponding to its maximum (bottom). Note how the Hough images $M^t(h)$ are changed between iterations, so that implicit "non-maximum suppression" driven by the probability function is performed. As a result, multiple pedestrians are detected despite significant overlaps between them.

votes. An example of this process is visualized in Figure 2. These dynamic changes driven by the the probability distribution (8) can be regarded as a principled non-maximum suppression procedure.

Several optimizations can be performed to ensure a reasonable computational speed. For one thing, $\log P(x_i = y_{x_i^t}|I_i)$ can be stored from the previous iteration rather than recomputed afresh each time. More generally, the new Hough image M^t can be in many cases computed incrementally from M^{t+1} by subtracting the previous votes and adding the new votes for the voting elements i that have changed x_i.

4. Experiments

In this section, we present the experimental comparison of our approach with the traditional Hough transform. For Hough transform, we used a greedy procedure with non-maximum suppression to process the Hough images: we subsequently pick the hypotheses corresponding to the highest-value point in the Hough image, that is not closer than R to the previously selected hypotheses. We keep picking the hypothesis as long as the highest-value point (that is not close to previously selected) has the value greater than some threshold τ. Quantitative comparisons are performed via Recall-Precision curves generated by varying λ parameter in the case of our method and τ parameter in the case of the traditional Hough transform.

4.1. Line detection

Experimental protocol. We first start with the classical problem of line detection in images. As a benchmark, we considered the YorkUrbanDB [1] dataset [7]. The dataset contains 102 images of urban scenes, of which 20 were used for parameter validation and 82 for testing the performance. The scenes in the dataset have predominantly "Manhattan" geometry, with the majority of straight lines belonging to the three orthogonal families. The authors of the dataset also provide a set of "Manhattan" line segments semi-automatically annotated in each image as well as the locations of "Manhattan" vanishing points.

Given a set of straight lines detected with some algorithm, we define the *recall* to be the ratio of the straight segments that lie within 2 pixels from one of the detected lines (a generous 2 pixel threshold was allowed to account both for discretization effects and for the edge detector imperfections). We also considered two measures of *precision*. For the first, more traditional measure, we just matched detected lines to the ground truth segments that lied within 2 pixels from them (at most one line could be matched to each segment), and counted the ratio of matched lines to all lines. This measure however often penalizes correctly detected lines that were not annotated. We therefore computed the second measure of precision by counting the number of non-Manhattan lines treating them as errors. Such approach still penalizes correctly detected non-Manhattan lines, but there are few of them in the particular dataset, we have considered. To determine whether a line is a Manhattan one, we look at the angles between the line and the directions towards the ground truth vanishing points from the midpoint of the part of the line visible in the image. If all three angles are greater than 2 degrees, then we treat the line as non-Manhattan and erroneous detection.

[1] available at http://www.elderlab.yorku.ca/YorkUrbanDB/

Algorithmic details. For each image a Canny edge detection (OpenCV implementation, Canny thresholds = 200 and 80) was performed. Each edge pixel was considered a voting element, described by its position I_i in the image plain. We defined:

$$p(x_i = l|I_i) = Z_3 \exp\left(-\operatorname{dist}(i,l)^2\right) \quad (12)$$

$$p(x_i = 0|I_i) = Z_3 \exp\left(-C_1\right) , \quad (13)$$

where $\operatorname{dist}(i,l)$ is the distance between the edge pixel i and the line l, Z_3 is a normalizing constant, C_1 is a constant set up by validation[2]. We then used the greedy version of our framework to detect the lines in the images.

For benchmarking, we compared the results of our framework with the results of the Hough transform followed by non-maximum supression. We used the "soft" voting scheme, where each edge pixel i voted for the line l with the strength $\max(C_2 - \operatorname{dist}(i,l)^2, 0)$, so that the Hough images produced during the Hough voting were essentially the same as on the first step of our greedy algorithm (given $C_1 = C_2$). We then identified the local maxima in the Hough images, and performed non-maxima suppression. These required some reasonable distance measure between the lines, for which we used the following. Given two lines l_1 and l_2, we again clipped them against the image boundaries obtaining segments s_1 and s_2. We then defined the distance between lines l_1 and l_2 to be the maximum over 4 numbers corresponding to the distances from each endpoint of each clipped segment (s_1 and s_2) to the other line (l_2 or l_1 respectively). The minimal distance R within the non-maximum suppression as well as the optimal C_2 were set up by validation.

Finally, as the mode-seeking algorithms are a popular alternative to non-maxima suppression within Hough transform [14], we tried the medoid-shift algorithm [19] to prune the set of local maxima in the Hough map. The distance function between lines was the same as above and the value of the bandpass paramter σ was set-up by validation. We thus run the medoid-shift algorithm leaving only the maxima that were found to be the medoids. Unfortunately, the results we got using the medoid-shift algorithm were uniformly worse than those with the non-maxima suppression, therefore we do not report them here.

Results. The recall-precision curves for our algorithm as well as the baseline algorithm (Hough transform + NMS) on the test set are given in Figure 3. As can be seen, our approach outperforms the baseline considerably with respect to both precision measures. In particular, the optimal maximum suppression radius for the baseline algorithm ($R = 32$) makes the baseline algorithm unsuitable when higher values of recall are desired (as in our experiments we considered 6000 local minima in Hough image with the highest value, we had to truncate the respective curves at about 0.15 recall, that corresponded to the highest τ-value for the 6000th strongest local maximum across the images in the test dataset). Some qualitative examples at a low recall-high precision value are given in Figure 4.

[2]While it is possible to introduce an additional variance parameter σ into the exponent in (12), a careful inspection reveals that this would not change the family of energies (9) spanned by different C_1 and λ.

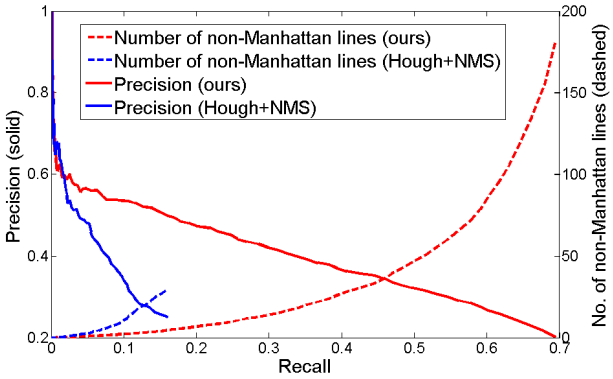


Figure 3. Recall-precision and recall-(average number of non-Manhattan lines) curves for our framework and for the baseline approach, i.e. Hough transform followed by non-maximum suppression. See the text for the description of the experimental protocol.

4.2. Pedestrian detection

Experimental protocol. We now describe the experiments on detection of pedestrians in street images. We downloaded two video sequences *TUD-campus* and *TUD-crossing* [3] containing mostly profile views of pedestrians in relatively crowded locations. The original annotations provided with [2] included only the pedestrians occluded by no more than 50%. As we were interested in the performance of the method under significant overlap, we reannotated the data by marking all pedestrians whose head and at least one leg were clearly visible. After reannotation, the *TUD-campus* and *TUD-crossing* sequences contain 71 images with 304 ground truth bounding boxes annotated, and 201 images with 1018 bounding boxes accordingly.

To obtain the probabilistic votes we used the Hough forest [9] learned on the separate training dataset (considered in [9]). Hough forests are learned on a dataset of 16x16 patches extracted from images with the objects of interest (pedestrians) at a fixed scale, and from the set of background images. After training, Hough forests are able to map the patch appearance and location (encoded by I_i) directly to $p(x_i|I_i)$, which is exactly what is needed in our framework.

Algorithmic details. For single-scale scenario, Hough forests can be seamlessly incorporporated into our framework. Full version of the greedy algorithm is directly ap-

[3]available at `http://www.mis.tu-darmstadt.de/node/382`

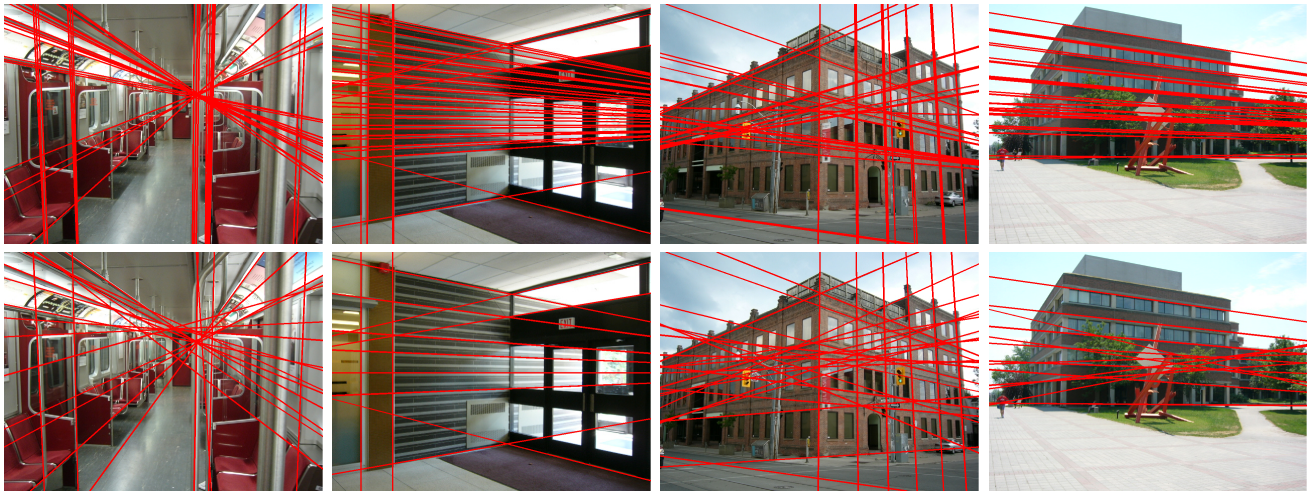


Figure 4. Sample detections for our framework (top) and Hough transform+NMS (bottom). The λ and τ parameters were set so that both methods detect on average 25 lines per image. Note, how our framework is able to discern very close yet distinct lines, and is in general much less plagued by spurious detections.

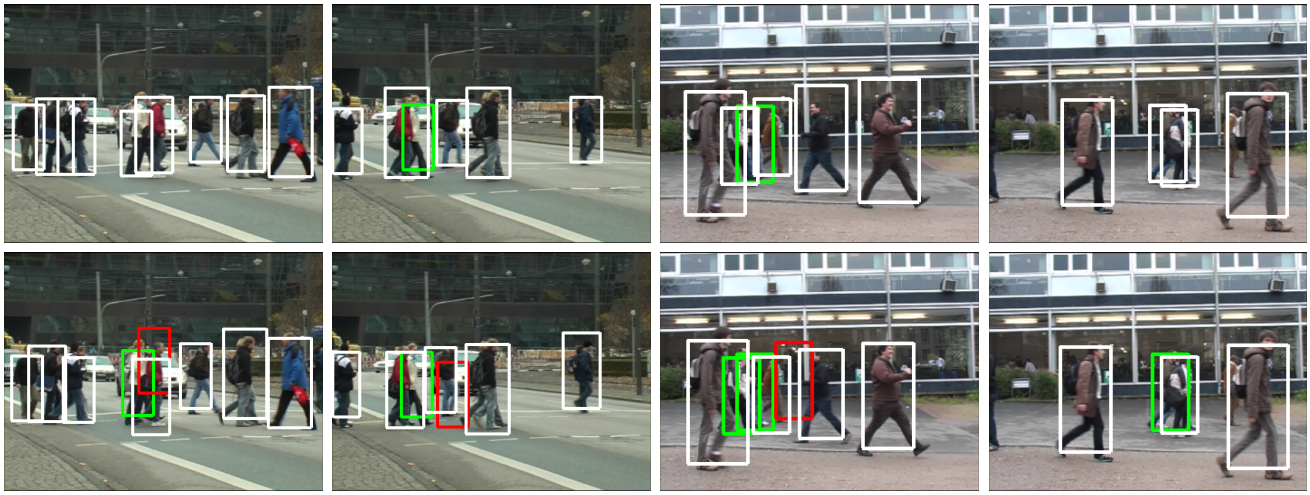


Figure 5. Sample detection results for our framework (top) and Hough transform+NMS (bottom) for the TUD-crossing and TUD-campus sequences at equal error rates (white = correct detection, red = false positive, green = missed detection). Note how our framework is capable of detecting strongly overlapping objects without producing many false positives.

plicable here, with both set of the voting elements and set of possible centroid locations beeing the set of all pixels (Figure 2 provides an example of the greedy algorithm in a single-scale case). We were however interested in the detection in a multiscale setting, in which the Hough space is parameterized by the centroid location χ_i and the scale s_i of the pedestrian. To get an estimate of $p(x_i = (\chi_i, s_i)|I_i)$ we apply Hough forest to the image resized by the scale s_i. The background probability $p(x_i = 0|I_i)$ in our framework was set to a constant chosen on the validation set.

In our experiments the set of voting elements was parameterized by pixels at the largest scale. As objects at different scales are of different sizes, detection of larger object should require more evidence than detection of a smaller object. So in the experiments we upscaled λ proportionally to the area of objects at a particular image scale.

The performance of our algorithm was compared with the performance of [9] that uses non-maxima suppression based on the overlap criterion. The overlap threshold within the NMS was set up by cross-validation. For both algorithms, we used one of the sequences for the validation and then tested on the other.

We used 3 scales for the *TUD-crossing* sequence and 5 scales for the *TUD-campus* sequence all differing by a factor of 0.85. For the matching between the produced set of detections and the ground truth, the standard 50% overlap criterion was used.

Results. Resulting recall-precision curves are shown in Figure 6. The Hough voting [9] with non-maxima suppression fails to achieve high recall values in multi-scale setting. This happens in part because close peaks corresponding to the same object arise in Hough images of different scales,

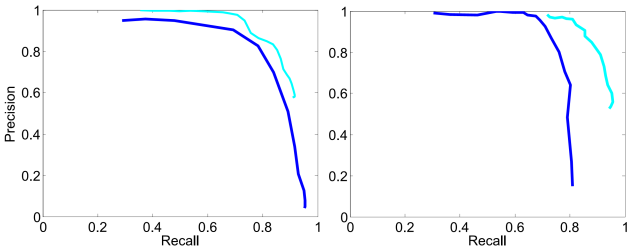


Figure 6. Precision-recall curves for our framework (light-blue) and Hough transform+NMS (dark-blue) on *TUD-crossing*(left) and *TUD-campus* (right) sequences. Our approach achieves better performance on these datasets containing a significant number of object overlaps.

and non-maximum suppression could not filter out these duplicated detections without filtering out close correct detections as well (see Figure 5). The proposed framework does not require discerning between such peaks and thus shows better performance on both datasets.

5. Related work

The model for Hough voting presented in this paper is related to a number of existing frameworks for object recognition and segmentation.

The method most closely related to ours is the work of Leibe et al [14] on Hough-based object detection using interleaved categorization and segmentation. Their method works by using object segmentation and a MDL prior to prune out false hypothesis. The segmentation obtained in their framework is in some sense similar to the assignment $\mathbf{x}$ of voting elements in our framework (assuming we consider all pixels as voting elements). However, unlike our method which performs inference jointly over $\mathbf{x}$ and $\mathbf{y}$, the method presented in [14] works in stages. In the first step, it estimates the segmentation of different objects detected in the scene. This segmentation result along with a MDL prior is used in the second step to perform hypothesis verification and prune out misleading object detections.

MDL prior for object recognition was also considered by Hoiem et al. in [11], where they showed how this prior can be incorporated in the layout CRF model [20]. Another approach that optimizes over sets of object instances, treating these sets as competing interpretations of the object parts was presented by Amit et al. in [1]. Most recently, Desai et al. [6] showed how priors on the configurations of hypothesis variables $\mathbf{y}$ can be learned in a discriminative max-margin fashion. Finally, the work of Lazic et al.[13] considered the use of similar ideas (facility location) for the subspace segmentation problems.

6. Summary

We have presented a framework for detecting multiple object instances in images, which is similar to the traditional Hough transform. It was demonstrated that by redeveloping Hough transform within a probabilistic framework, one can avoid solving the tricky problem of distinguishing multiple local peaks in the Hough image. Instead, the greedy inference in our framework only requires picking the overall (global) maxima of a sequence of Hough images. In this way, non-maxima suppression step can be bypassed altogether, and, according to our experiments, a significant increase in accuracy can be obtained.

The code for line and pedestrian detection based on greedy inference within our framework as well as the additional annotations for the TUD datasets publicly will be made available at the project webpage by the time of the publication.

References

[1] Y. Amit, D. Geman, and X. Fan. A coarse-to-fine strategy for multi-class shape detection. *TPAMI*, 26(12):1606–1621, 2004.

[2] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. *CVPR*, 2008.

[3] D. H. Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122, 1981.

[4] O. Barinova, V. Lempitsky, and P. Kohli. On detection of multiple object instances using hough transforms. Technical report, 2010.

[5] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. *ICCV*, 2009.

[6] C. F. C. Desai, D. Ramanan. Discriminative models for multi-class object layout. *ICCV*, 2009.

[7] P. Denis, J. H. Elder, and F. J. Estrada. Efficient edge-based methods for estimating manhattan frames in urban imagery. *ECCV*, 2008.

[8] B. J. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315:972–976, 2007.

[9] J. Gall and V. Lempitsky. Class-specific hough forests for object detection. *CVPR*, 2009.

[10] C. Gu, J. J. Lim, P. Arbelaez, and J. Malik. Recognition using regions. *CVPR*, 2009.

[11] D. Hoiem, C. Rother, and J. M. Winn. 3d layoutcrf for multi-view object class recognition and segmentation. *CVPR*, 2007.

[12] P. Hough. Machine analysis of bubble chamber pictures. *Int. Conf. High Energy Accelerators and Instrumentation*, 1959.

[13] N. Lazic, I. Givoni, B. Frey, and P. Aarabi. Floss: Facility location for subspace segmentation. *ICCV*, 2009.

[14] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *IJCV*, 77(1-3):259–289, 2008.

[15] S. Maji and J. Malik. Object detection using a max-margin hough tranform. *CVPR*, 2009.

[16] G. L. Nemhauser and L. A. Wolsey. Best algorithms for approximating the maximum of a submodular set function. *Mathematics of operations research*, 3(3):177–188, 1978.

[17] R. Okada. Discriminative generalized hough transform for object detection. *ICCV*, 2009.

[18] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, Palo Alto, 1988.

[19] Y. Sheikh, E. A. Khan, and T. Kanade. Mode-seeking by medoidshifts. *ICCV*, 2007.

[20] J. M. Winn and J. Shotton. The layout consistent random field for recognizing and segmenting partially occluded objects. *CVPR (1)*, pp. 37–44, 2006.