Int J Comput Vis (2014) 106:332–341
DOI 10.1007/s11263-013-0623-2

Demisting the Hough Transform for 3D Shape Recognition and Registration

Oliver J. Woodford · Minh-Tri Pham · Atsuto Maki · Frank Perbet · Björn Stenger

Received: 19 September 2012 / Accepted: 1 April 2013 / Published online: 14 April 2013

霍夫变化

Abstract In applying the Hough transform to the problem of 3D shape recognition and registration, we develop two new and powerful improvements to this popular inference method. The first, *intrinsic Hough*, solves the problem of exponential memory requirements of the standard Hough transform by exploiting the sparsity of the Hough space. The second, *minimum-entropy Hough*, explains away incorrect votes, substantially reducing the number of modes in the posterior distribution of class and pose, and improving precision. Our experiments demonstrate that these contributions make the Hough transform not only tractable but also highly accurate for our example application. Both contributions can be applied to other tasks that already use the standard Hough transform.

内在霍夫变换

最小熵

Keywords Hough transform · Object recognition · 3d shape · Registration

1 Introduction

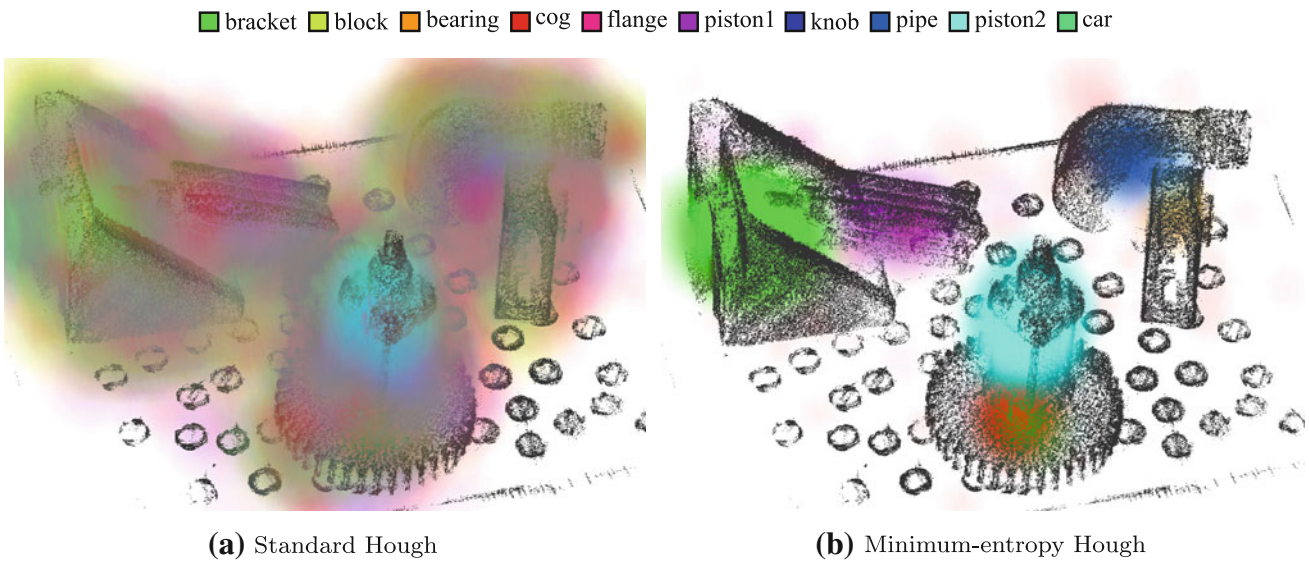
The Hough transform Duda and Hart (1972), named after Hough (1962) patent describing a method for detecting lines in images, has since been generalized to detecting, as well as recognizing, many other objects or instances: parameterized curves (Duda and Hart 1972), arbitrary 2D shapes (Ballard 1981), object motions (Bober and Kittler 1993), cars (Gall and Lempitsky 2009; Leibe et al. 2008) pedestrians (Barinova et al. 2010; Gall and Lempitsky 2009), hands Okada (2009) and 3D shapes (Knopp et al. 2010; Pham et al. 2011; Tombari and Di Stefano 2010), to name but a few. This popularity stems from the simplicity and generality of the first step of the Hough transform—the conversion of *features*, found in the data space, into sets of *votes* in a Hough space, parameterized by the pose of the object(s) to be found. Various different approaches to learning this feature-to-vote conversion function have been proposed, including the *implicit shape model* Leibe et al. (2008) and *Hough forests* (Gall and Lempitsky 2009; Okada 2009).

The second stage of the Hough transform simply sums the likelihoods of the votes at each location in Hough space, then selects the modes. One problem with this step is that the summation can create modes where there are only a few outlier votes. A second problem is that, given a required accuracy, the size of the Hough space is exponential in its dimensionality. The application we are concerned with, object recognition and registration (R&R) from 3D geometry (here, point clouds), suffers significantly from both these problems. The Hough space, at 8D (one dimension for class, three for rotation, three for translation and one for scale), is to our knowledge the largest to which the Hough transform has been applied, and the feature-to-vote conversion generates a high proportion of incorrect votes, creating a "mist" of object likelihood throughout that space, as shown in Fig. 1a.

In the face of this adversity, we have developed two important contributions which enable inference on this task, and potentially many others, using the Hough transform to be both feasible and accurate:

- We introduce the *intrinsic Hough transform*, which substantially reduces memory and computational requirements in applications with a high dimensional Hough space.
- We introduce the *minimum-entropy Hough transform*, which greatly improves the precision and robustness of the Hough transform.

O. J. Woodford (✉) · M-T. Pham · A. Maki · F. Perbet · B. Stenger
Toshiba Research Europe Ltd., 208 Cambridge Science Park, Milton Road, Cambridge CB4 0GZ, UK
e-mail: oliver.woodford@crl.toshiba.co.uk



(a) Standard Hough **(b)** Minimum-entropy Hough

Fig. 1 Demisting the Hough transform. Posterior distributions over translation and ten object classes (six of which are present in the scene), with scale and rotation marginalized out, for (**a**) the standard Hough transform, and (**b**) the minimum-entropy Hough transform introduced here

These extensions of the Hough transform are not task specific; they can be applied, either together or independently, to any application that does or is able to use the standard Hough transform.

The rest of this paper is organized as follows: The next section describes inference using the Hough transform, and briefly reviews the literature relevant to our contributions. In Sect. 3 we describe our new inference methods. The section following that describes and discusses our experiments. Finally, we conclude in Sect. 5.

2 Background

2.1 3D Shape Recognition and Registration

The *implicit shape model* of Leibe et al. (2004, 2008) pioneered the use of the Hough transform for object recognition in 2D images. This approach has since been applied to object recognition in 3D geometric data (Knopp et al. 2010), and extended to object registration (Pham et al. 2011; Tombari and Di Stefano 2010). For the dual problem of R&R in 3D, the Hough space is either 7D (if scale is known) (Tombari and Di Stefano 2010) or 8D (Pham et al. 2011).

The feature extraction stages of these methods follow the same pipeline: features are detected at a given scale and position; a canonical orientation of the feature is estimated; a descriptor for the feature is computed. The votes are then computed by matching features in the test data with features from training data with ground truth class and pose, either directly (i.e. a nearest neighbour search) (Pham et al. 2011), or via a codebook created by clustering the feature descriptors (Knopp et al. 2010; Leibe et al. 2008; Tombari and Di Stefano 2010).

In this work we will be using the feature-to-vote conversion process of Pham et al. (2011) as an off-the-shelf method, since our contributions lie in the second stage of the Hough transform. It is this process that generates a high proportion of incorrect votes, amongst which the correct votes need to be found.

2.2 The Hough Transform

The earliest descriptions of the Hough transform (Ballard 1981; Duda and Hart 1972; Hough 1962) present it as an algorithm, but more recently there has been a desire to cast the framework in a probabilistic light. *Generative* model interpretations (Allan and Williams 2009; Barinova et al. 2010; Stephens 1991) in which the votes represent likelihoods of features, given an object pose, require that the likelihoods of these independent variables be multiplied, in contrast to the summation of the Hough transform. The summation has been explained in two ways: firstly that it is in fact over the log likelihood of features (Barinova et al. 2010; Stephens 1991), though this requires a differently shaped distribution for each vote than is typically given (Barinova et al. 2010), or secondly that it is a first order approximation to a robustified product of likelihoods (Allan and Williams 2009; Minka 2003). We prefer to interpret the second stage of the Hough transform as a *discriminative* model of the posterior distribution of an object's location, phrased simply as a kernel density estimate over all the votes (Bober and Kittler 1993; Zhang and Chen 2010).

Let $\mathbf{y}$ be an object's location in a Hough space, $\mathcal{H}$, which is the space of all object poses (usually real) and, in the case of object recognition tasks, object classes (discrete). Furthermore, let the list of votes, cast in $\mathcal{H}$ by N features, which are computed in some first stage feature-to-vote conversion process (not addressed here) be denoted by $\mathbf{X} = \{\{\mathbf{x}_{ij}\}_{j=1}^{J_i}\}_{i=1}^{N}$. The posterior probability of an object's location is then given by

$$p(\mathbf{y}|\mathbf{X}, \boldsymbol{\omega}, \boldsymbol{\theta}) = \sum_{i=1}^{N} \omega_i \sum_{j=1}^{J_i} \theta_{ij} K(\mathbf{x}_{ij}, \mathbf{y}), \tag{1}$$

where J_i is the number of votes generated by the i^{th} feature, $K(\cdot,\cdot)$ is a density kernel in Hough space, and $\boldsymbol{\omega} = \{\omega_i\}_{i=1}^{N}$ and $\boldsymbol{\theta} = \{\theta_{ij}\}_{\forall i,j}$ are feature and vote weights respectively, s.t. $\omega_i \geq 0,\ \forall i,\ \sum_{i=1}^{N} \omega_i = 1$, and

$$\theta_{ij} \geq 0, \quad \forall i, j, \qquad \sum_{j=1}^{J_i} \theta_{ij} = 1, \quad \forall i \in \{1, .., N\}. \tag{2}$$

For example, in the original Hough transform used for line detection (Duda and Hart 1972), the features are edgels, votes are generated for a discrete set of lines (parameterized by angle) passing through each edgel, the kernel, $K(\cdot,\cdot)$, returns 1 for the nearest point in the discretized Hough space to the input vote, 0 otherwise, and the weights, $\boldsymbol{\omega}$ and $\boldsymbol{\theta}$, are set to uniform distributions. Recently methods have been proposed for learning *a priori* more discriminative weights (Maji and Malik 2009; Zhang and Chen 2010) for object detection, as well as evaluating over different kernel shapes (Zhang and Chen 2010).

The final stage of the Hough transform involves finding, using non-maxima suppression, the modes of this distribution whose probabilities are above a certain threshold value, τ.

2.2.1 Computational Feasibility

Finding the modes in $\mathcal{H}$ involves sampling that space, the volume of which increases exponentially with its dimensionality, d. Several approaches have been proposed to reduce this burden, which we categorize as one of approximate, hierarchical, irregular or mode-seeking.

Approximate methods use reduced-dimensionality approximations of the full Hough space to find modes. For example, given a 6D pose (translation and rotation), Fisher et al. (1998) quantize translations and rotations in two separate 3D arrays (peak entries in both arrays indicate an object, but multiple objects create ambiguities), while Tombari and Di Stefano (2010) find modes over translation only, then compute an average rotation for each mode. Geometric hashing techniques, e.g. (Drost et al. 2010; Lamdan and Wolfson 1988; Mian et al. 2006), also fall into this category.

Hierarchical approaches, such as the *fast* (Li et al. 1986) and *adaptive* (Illingworth and Kittler 1987) Hough transforms, sample the space in a coarse-to-fine manner, exploiting the sparsity of some areas, though their complexity is still exponential in d.

Irregular methods do not sample the Hough space regularly, but rather sample only where objects are likely to be detected, again exploiting potential sparsity in $\mathcal{H}$. For example, the *combinatorial* (Ben-Tzvi and Sandler 1990) and *randomized* (Xu et al. 1990) Hough transforms generate lists of sampling locations, the former for all lines (in line detection) joining pairs of edgels in confined regions, the latter for curves (in curve detection) defined uniquely by random sets of edgels. Both these approaches are task specific, whereas the intrinsic Hough transform introduced here, which also falls into this category, is not.

Mode-seeking methods find modes in $\mathcal{H}$ through iterative optimization (Bober and Kittler 1993; Cheng 1995). Mean shift (Cheng 1995) is the most commonly used approach, the complexity of which is $O(nd^2)$, where $n = \sum_{i=1}^{N} J_i$ (the total number of votes). It has successfully been applied to an 8D Hough space (Pham et al. 2011). However, it needs to be initialized in many, perhaps $O(n)$, locations, making the total complexity $O(n^2d^2)$, and is not guaranteed to find every mode. Two extensions of this approach, though generally applied to clustering rather than mode seeking, are medoid shift (Sheikh et al. 2007) and quick shift (Vedaldi and Soatto 2008).

These approaches can also be combined. For example, modes found in a coarse sampling of $\mathcal{H}$ can be refined using mean shift (Leibe et al. 2008), an approach we employ here.

2.2.2 Explaining Away Votes

The summing of votes in the Hough transform enables incorrect votes to generate modes in $\mathcal{H}$, and since most applications tend to produce a large number of incorrect votes, this can lead to false detections, especially in multi-object detection scenarios. The problem arises from the fact that each test feature generates a number (often quite large) of votes, which represent the locations of all objects that *could* have generated that feature, but usually only one of those votes will actually be correct, because most features are generated by only one object. Figure 1a visualizes the ambiguity caused by these incorrect votes in our R&R application.

If we assume, usually correctly, that a feature is generated by only one object, we can then enforce the resulting implicit constraint that *only one vote cast by each feature is correct*. By choosing which vote this is for each feature, the other votes can then be dismissed as being incorrect, removing them from the transform—the correct vote essen-

tially explains away all the other votes. This assumption was first applied to the Hough transform in the 1980s by Gerig (1987), using a two stage approach, first computing the standard Hough transform, then, simultaneously for each feature, collating the values of the Hough transform at the locations of all votes of a given feature, and keeping only the vote at the highest value.

The idea was resurrected more recently by Barinova et al. (2010), using an approach akin to the Hough transform, in that it exhaustively samples the Hough space while searching for objects. However, they directly enforce the constraint that *a feature is generated by only one object*, using feature-to-object assignments, with a cost per object detection. Phrasing the problem as an energy minimization, they greedily detect objects in Hough space, assigning to them features which decrease the overall energy. Furthermore, rather than using kernels that tail off to zero, their kernels continue decreasing away from the vote, with an explicit background assignment for outlier features.

Several other multi-object detection frameworks also make explicit feature-to-object assignments: energy-minimization-based methods (Birchfield and Tomasi 1999; Delong et al. 2012a,b; Isack and Boykov 2012; Woodford et al. 2012), which iteratively update the assignments; RANSAC, similar to energy-based methods but focusing more on the algorithm than the objective function, with features assigned either greedily (Vincent and Laganiere 2001) or with iterative refinement (Zhang and Kŏsecká 2007; Zuliani et al. 2005); non-parametric methods, which cluster features into groups representing objects (Toldo and Fusiello 2008).

A benefit of methods using feature-to-object assignments, as opposed to the feature-to-vote assignments of Hough-based methods, is that they avoid the last step of the Hough transform: non-maxima suppression of accumulated votes in Hough space.

3 Our Framework

This section describes our improvements to the Hough transform. In Sect. 3.1 we introduce the intrinsic Hough transform, which overcomes the high memory requirements of the standard Hough transform with high-dim-ensional Hough spaces. In Sect. 3.2 we introduce a method which exploits the assumption that only one vote per feature is correct.

3.1 The Intrinsic Hough Transform

As discussed in Sect. 2.2, high-dimensional Hough spaces require infeasible amounts of memory to sample regularly. However, we note that while the volume of the Hough space increases exponentially with its dimensionality, the number of votes generated in applications using the Hough transform generally does not, implying that higher dimensional Hough spaces are often sparser. We exploit this sparsity by sampling the Hough space only at locations where the probability (given by Eq. 1) is likely to be non-zero. Assuming that the density kernel, $K(\cdot,\cdot)$, in Eq. (1) is zero-mean and unimodal (which is generally true for kernel density estimation), the modes of the distribution will be at or near the locations of the votes. We therefore simply sample the Hough space at the locations of the votes themselves. Since the votes define the distribution, therefore are intrinsic to it, we call this approach the *intrinsic Hough transform*.

While similar in some respects to intrinsic mode-seeking algorithms (Sheikh et al. 2007; Vedaldi and Soatto 2008), the intrinsic Hough transform does not seek modes through iterative updates. Rather, the modes of the distribution are detected using non-maxima suppression, as per the standard Hough transform; here, a sample location, $\mathbf{y}$, is classified as a mode if no other sample location, $\mathbf{z}$, within a certain distance, s.t. $K(\mathbf{y},\mathbf{z}) > \gamma$, has a higher probability. Implicit in this approach is the assumption that the local modes of the distribution given by Eq. (1) lie very close to a vote—this is the case for most shapes of kernel used in practice. As a final step to improve accuracy, the location of each mode found is updated with one step of mean shift. The memory and computational requirements of this approach are $\mathrm{O}(n)$ and $\mathrm{O}(n^2d^2)$ respectively.

3.2 The Minimum-Entropy Hough Transform

Making the assumption that only one vote per feature is correct, a vote that is believed to be correct should explain away the other votes from that feature. This suggests that, rather than being given θ a priori, it would be beneficial to optimize over its possible values, giving those votes which agree with votes from other features more weight than those which do not.

One way of achieving this is by minimizing the information *entropy*[1] of $p(\mathbf{y}|\mathbf{X},\omega,\theta)$ w.r.t. θ. A similar approach, but minimizing entropy w.r.t. some parameters of the vote generation process, has already been used for lens distortion calibration (Rosten and Loveland 2009). A lower entropy distribution contains less information, making it more peaky and hence having more votes in agreement. Since information in Hough space is the location of objects, minimizing entropy constrains features to be generated by as few objects as possible. This can be viewed as enforcing Occam's razor. The objective function to be minimized is therefore

[1] Specifically we use the Shannon (1948) entropy, $H = E[-\ln p(x)] = -\int p(x)\ln p(x)\,\mathrm{d}x$.

$$f(\theta) = -\int_{\mathcal{H}} p(\mathbf{y}|\mathbf{X}, \boldsymbol{\omega}, \theta) \ln p(\mathbf{y}|\mathbf{X}, \boldsymbol{\omega}, \theta)\, \mathrm{d}\mathbf{y} \tag{3}$$

However, computing this entropy involves an integration over Hough space; for our application this is very large. To make this integration tractable we sample the space at discrete locations using importance sampling (MacKay 2009, Section 29.2); as with the intrinsic Hough transform, we sample the Hough space at the locations of all the votes. The value of θ is therefore approximated by

$$\theta = \underset{\theta'}{\operatorname{argmin}} \left[-\sum_{i=1}^{N} \sum_{j=1}^{J_i} \frac{p(\mathbf{x}_{ij}|\mathbf{X}, \boldsymbol{\omega}, \theta')}{q(\mathbf{x}_{ij})} \ln p(\mathbf{x}_{ij}|\mathbf{X}, \boldsymbol{\omega}, \theta') \right] \tag{4}$$

where $q(\cdot)$ is the (unknown) sampling distribution from which the votes are drawn. Once this optimization (described below) is done, the estimated θ is applied to Eq. (1), and inference continues as per the standard (or intrinsic) Hough transform. We call this approach the *minimum-entropy Hough transform*.[2]

3.2.1 Optimization Framework

It turns out, as we show in the Appendix, that a global minimum of Eq. (3) must lie at an extremum of the parameter space, which is constrained by Eq. (2), such that at least one optimal value of $\theta_i = \{\theta_{ij}\}_{j=1}^{J_i}$ (i.e. the vector of feature i's vote weights) will be an all 0 vector, except for one 1, i.e. minimizing entropy naturally enforces the one-correct-vote-per-feature constraint. As a result, a global minimum can always be found if we limit the search space for each θ_i to integer values, making a discrete set of J_i possible vectors, s.t. the total number of possible solutions is $\prod_{i=1}^{N} J_i$. It should be noted that this search space is not uni-modal—for example, if there are only two features and they each identically generate two votes, one for location $\mathbf{y}$ and one for location $\mathbf{z}$, then both $\mathbf{y}$ and $\mathbf{z}$ will be modes. Furthermore, as the search space is exponential in the number of features, an exhaustive search is infeasible for all but the smallest problems.

We therefore use a local approach, iterated conditional modes (ICM) (Besag 1986), to quickly find a local minimum of this optimization problem. This involves updating the vote weights of each feature in turn, by minimizing Eq. (4) conditioned on the current weights of all other votes, and repeating this process until convergence. The correct update equation for the vote weights of a feature f is as follows:

$$p_{fk}(\mathbf{y}|\mathbf{X}, \boldsymbol{\omega}, \theta) = \omega_f K(\mathbf{x}_{fk}, \mathbf{y}) + \sum_{\forall i \neq f} \omega_i \sum_{j=1}^{J_i} \theta_{ij} K(\mathbf{x}_{ij}, \mathbf{y}), \tag{5}$$

$$k = \underset{k'=1}{\overset{J_f}{\operatorname{argmax}}} \left[\sum_{i=1}^{N} \sum_{j=1}^{J_i} \frac{p_{fk'}(\mathbf{x}_{ij}|\mathbf{X}, \boldsymbol{\omega}, \theta)}{q(\mathbf{x}_{ij})} \ln p_{fk'}(\mathbf{x}_{ij}|\mathbf{X}, \boldsymbol{\omega}, \theta) \right], \tag{6}$$

$$\theta_{fk} = 1, \qquad \theta_{fj} = 0, \qquad \forall j \neq k. \tag{7}$$

However, since this update not only involves $q(\cdot)$, which is unknown, but is also relatively costly to compute, we replace it with a simpler proxy which in practice performs a similar job of encouraging the resulting posterior distribution to be as peaky as possible:

$$k = \underset{k'=1}{\overset{J_f}{\operatorname{argmax}}}\, p_{fk'}(\mathbf{x}_{fk'}|\mathbf{X}, \boldsymbol{\omega}, \theta). \tag{8}$$

This is effectively the strategy of Gerig, but applied sequentially rather than simultaneously. Since the optimization is local, a good initialization of θ is key to reaching a good minimum. In our experiments we start at the value of θ used in the standard Hough transform, then applied the following update to each vote weight simultaneously:

$$\theta_{ik} = \frac{p_{ik}(\mathbf{x}_{ik}|\mathbf{X}, \boldsymbol{\omega}, \theta)}{\sum_{j=1}^{J_i} p_{ij}(\mathbf{x}_{ij}|\mathbf{X}, \boldsymbol{\omega}, \theta)}, \tag{9}$$

iterating this five times before starting ICM. Initially updating weights softly, i.e. not fixing them to 0 or 1, and synchronously, avoiding ordering bias, in this way helped to avoid falling into a poor local minimum early on, thus improving the quality of solution found.

4 Experiments

4.1 Setup

For our test application, 3D shape R&R, we use the framework introduced by Pham et al. (2011), outlined in Fig. 2, the evaluation data from which can be found online (Toshiba CAD model point clouds dataset 2011). It consists of 100 test instances, each containing one object, for each of 10 object classes, shown in Fig. 3, i.e. 1,000 test instances in total. Each test instance provides ground truth 7D object pose (scale and 3D rotation and translation) and class, and a set of input votes, with weights, for object pose and from all 10 classes.

For the density kernel, $K(\cdot, \cdot)$, of Eq. (1) we use a Gaussian kernel on a symmetric version of the SRT distance between direct similarities[3] (Pham et al. 2011). For two object poses, $\mathbf{y}$ and $\mathbf{z}$, of the same class, it is defined as

[2] Strictly speaking, the minimum-entropy Hough transform is not a transform, because the probability of each location in Hough space cannot be computed independently.

[3] A direct similarity is a transformation consisting of a rotation, a translation and a uniform scaling.

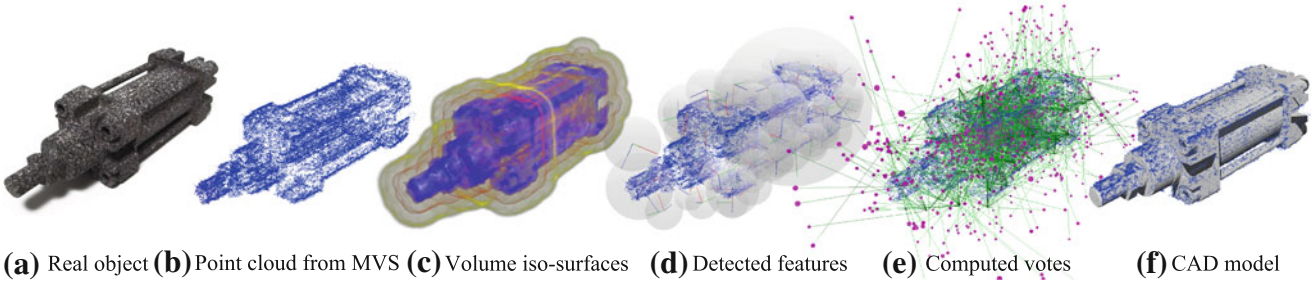


Fig. 2 The application: 3D-shape-based object R&R (framework and figure from Pham et al. 2011). (**a**) Real object, fabricated from a CAD model. (**b**) Point cloud extracted using a multi-view stereo (MVS) system (Vogiatzis and Hernández 2011). (**c**) Iso-surfaces of the scalar volume computed from the points. (**d**) Features (with position, scale and orientation) detected in the volume. (**e**) Votes for the object centre, based on detected features matched with a library of learnt features. (**f**) The registered CAD model

Fig. 3 Test objects. CAD models for the 10 test object classes

Table 1 Parameter values for the inference methods tested: **a** mean shift, **b** intrinsic Hough, **c** minimum-entropy Hough, **d** Gerig, **e** Greedy, **f** Barinova et al. (2010)

σ_s a–f	σ_r a–f	σ_t a–f	γ b–e	λ f
0.0694	0.12	0.12	exp(−8)	10

$$K(\mathbf{y},\mathbf{z}) = \frac{1}{\zeta}\exp\left(-\frac{d_s^2(\mathbf{y},\mathbf{z})}{\sigma_s^2} - \frac{d_r^2(\mathbf{y},\mathbf{z})}{\sigma_r^2} - \frac{d_t^2(\mathbf{y},\mathbf{z})}{\sigma_t^2}\right), \quad (10)$$

$$d_s(\mathbf{y},\mathbf{z}) = \left|\log\frac{s(\mathbf{y})}{s(\mathbf{z})}\right|, \quad (11)$$

$$d_r(\mathbf{y},\mathbf{z}) = \sqrt{1 - |\mathbf{q}(\mathbf{y})^{\mathsf{T}}\mathbf{q}(\mathbf{z})|}, \quad (12)$$

$$d_t(\mathbf{y},\mathbf{z}) = \frac{||\mathbf{t}(\mathbf{y}) - \mathbf{t}(\mathbf{z})||}{\sqrt{s(\mathbf{y})s(\mathbf{z})}}, \quad (13)$$

where $s(\mathbf{y})$, $\mathbf{q}(\mathbf{y})$ and $\mathbf{t}(\mathbf{y})$ are the scale, rotation (as a quaternion) and translation components of $\mathbf{y}$ respectively. If $\mathbf{y}$ and $\mathbf{z}$ specify different classes, then $K(\mathbf{y},\mathbf{z}) = 0$. The values of the bandwidth parameters, σ_s, σ_r and σ_t, given in Table 1, are those learned in Pham et al. (2011). The normalization factor, ζ, cannot easily be computed, but is independent of $\mathbf{z}$ (Pham et al. 2011), therefore, since our Eqs. (8) and (9) are scale independent,[4] it can be ignored.

[4] The requirement for a scale independent optimization strategy is a further reason to use the proxy of Eq. (8).

4.2 Methods

As well as evaluating the relative performance of the two Hough transforms introduced in Sects. 3.1 and 3.2, we compare them with the SRT mean shift method of Pham et al. (2011) (henceforth referred to as "mean shift"), and the inference methods of Gerig (1987) and Barinova et al. (2010) (here referred to simply as Gerig and BLK, after the authors, for short), and finally a *Greedy* approach which computes the standard Hough transform, finds the maximum and adds the corresponding object to the list of found objects, then removes all of the votes of all features that voted for that object, and repeats the process until no votes are left. Apart from mean shift, the methods all use the intrinsic Hough transform to make sampling $\mathcal{H}$ feasible. For the mean shift refinement step of the intrinsic Hough transform, we use the closed-form mean given in Pham et al. (2011), despite our slightly different density kernel. However, we do not refine the detections of BLK because their probability distribution is not amenable to this, since the likelihoods are multiplied. The likelihood function used in our implementation of BLK is the same kernel density function used in the other methods, defined in Eq. (10). We note that the parameters of this kernel were learned in Pham et al. (2011) specifically for Hough-based inference, therefore might not be optimal for BLK. Parameter values used for the various methods are summarized in Table 1.

4.3 Results

4.3.1 Quantitative Results

Quantitative results, computed using the ground truth classes and poses provided in the evaluation set, and using the registration criterion in Pham et al. (2011), are given in Tables 2 and 3 and Fig. 4. There is a small improvement in performance in both registration and recognition moving from mean shift to intrinsic Hough, which is most likely due to

Table 2 Quantitative results for the inference methods tested

	Recognition (%)	Registration (%)	Time (s)
Mean shift	64.9	72.8	0.427
intrinsic Hough	67.6	73.0	**0.192**
Min.-entropy	**98.5**	**79.6**	0.214
Gerig	71.8	73.3	0.218
Greedy	85.7	70.3	0.226
BLK	98.1	75.1	0.224

Bold values indicate the best achieved result across the methods tested

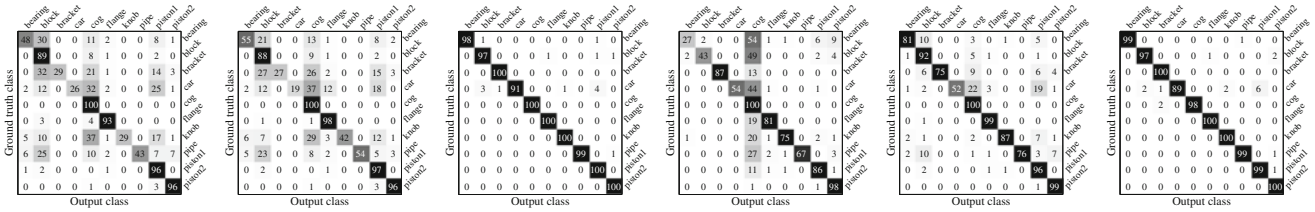
modes being missed by mean shift. Recognition rates then increase rapidly moving to Gerig, then Greedy, then finally minimum-entropy Hough, whose recognition rate, the largest seen, with only 1.5 % of objects left unrecognized, the majority of those in the car class, is a huge improvement on mean shift, providing a 96 % reduction in misclassifications. This improvement is due to the improved assignment of the correct vote per feature, from a one-shot simultaneous assignment, to a greedy assignment, to an iteratively refined assignment. Minimum-entropy Hough also shows a significantly improved registration rate, with top scores on 7/10 classes. BLK, though greedy, performs almost as well as minimum-entropy Hough in terms of recognition, though less well in terms of registration, in part due to a lack of mean shift pose the end.

However, because these results only reflect the best detection per test, they do not tell the whole story; we do not know how many other (incorrect) detections had competitive weights. To see this, we generated the precision-recall curves shown in Fig. 5, by varying the detection threshold, τ (or λ for BLK). A correct detection in this test required the class and pose to be correct simultaneously, and allowed only one correct detection per test. The curves show that precision remains high as recall increases for the minimum-entropy Hough transform, and marginally less so for BLK and Greedy, all of which are able to explain away

Table 3 Registration rate per class (%) for the six inference methods tested

	Bearing	Block	Bracket	Car	Cog	Flange	Knob	Pipe	Piston1	Piston2
Mean shift	77	13	95	75	**100**	86	88	86	44	64
intrinsic Hough	77	15	96	76	**100**	83	86	86	44	67
Minimum-entropy Hough	**83**	**20**	**98**	91	**100**	86	**91**	**89**	**54**	84
Gerig (1987)	76	13	96	84	**100**	84	85	83	46	66
Greedy	**83**	15	83	54	**100**	**89**	81	82	49	67
(BLK) Barinova et al. (2010)	79	**20**	97	**93**	**100**	74	73	81	48	**86**

Bold values indicate the best achieved result across the methods tested



(a) Mean shift **(b)** Intrinsic Hough **(c)** Min.-entropy **(d)** Gerig **(e)** Greedy **(f)** BLK

Fig. 4 Confusion matrices for the six inference methods tested

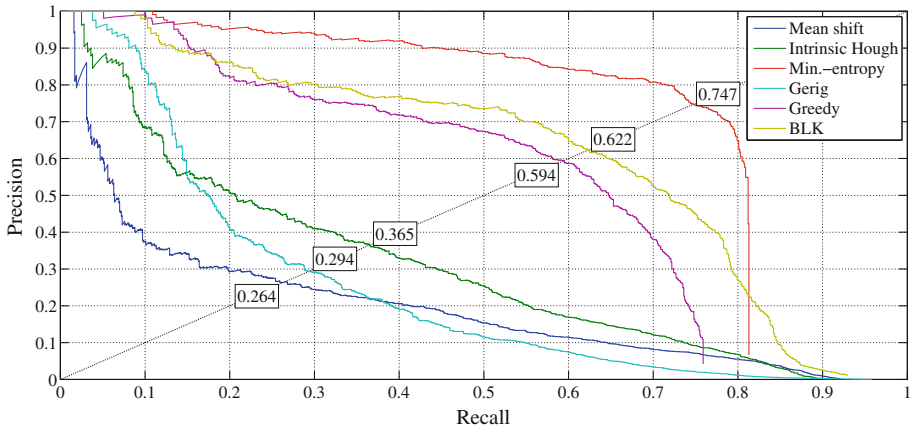
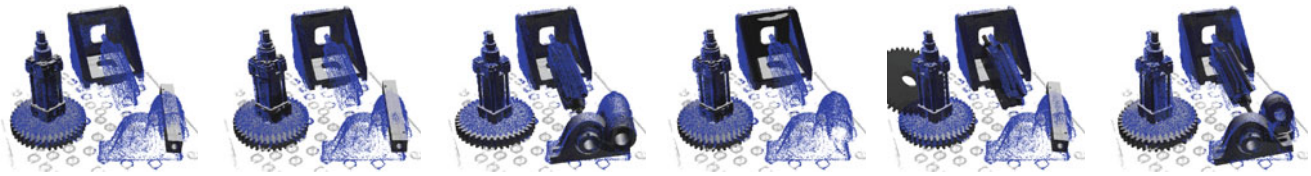


Fig. 5 Precision-recall curves for the six inference methods tested, with equal error rates



(a) Mean shift **(b)** Intrinsic Hough **(c)** Min.-entropy **(d)** Gerig **(e)** Greedy **(f)** BLK

Fig. 6 Qualitative results for the six inference methods tested, showing the first 6 objects (in order of decreasing weight) detected by each method in a point cloud containing 6 objects. Only minimum-entropy Hough (**c**) and the method of Barinova et al. (2010) (**f**) find all the objects

incorrect votes, while it drops off rapidly with recall for the other methods, indicating that the latter methods suffer from greater ambiguity as to which modes correspond to real objects, or perhaps in the case of Gerig, the wrong correct-vote assignments being made. Interestingly, Greedy and minimum-entropy Hough have lower maximum recall rates (of 0.759 and 0.813 respectively), which we propose is due to some correct modes being "explained away". Since, in the case of minimum-entropy Hough, our optimization strategy finds only a local minimum, we cannot be sure whether this effect is due to the objective function or the optimization strategy.

In terms of computation time (Table 2), all methods tested had the same order of magnitude speed, with the mean shift approach being about twice as slow as the others, though there is a trade-off of time versus accuracy with this approach, by changing the number of starting points of the optimization. However, we noticed that the speed of BLK was dependent on the value of λ, its detection threshold, and therefore equally the number of objects in the scene, unlike the other methods.

4.3.2 Qualitative Results

The benefit of explaining away incorrect votes is demonstrated in Fig. 1. While the standard Hough transform shows a great deal of ambiguity as to where and how many objects there are, the minimum-entropy Hough transform is able to clear away the "mist" of incorrect votes, leaving six distinct modes corresponding to the objects present; there are some other modes, but these are much less significant, corroborating the results seen in Fig. 5.

The benefit of having correct and clearly defined modes is demonstrated in Fig. 6, using the same point cloud as in Fig. 1, a challenging dataset containing three pairs of touching objects. Both minimum-entropy Hough and BLK find all six objects in the top six detections (though both mis-register the piston lying at a shallow angle), whereas the other methods find not only incorrect objects, but also multiple instances of correct objects (particularly the piston on the cog).

5 Conclusion

We have introduced two key extensions of the Hough transform, which can be applied to any approach using the Hough transform. The first, the intrinsic Hough transform, changes the memory requirements of the Hough transform from $O(k^d),\ (k > 1)$ to $O(n)$, making it feasible for high-dimensional Hough spaces such as that of our 3D shape R&R application. The second, the minimum-entropy Hough transform, was shown to significantly increase detection precision over mean shift on our task. We also showed that it marginally outperformed the probabilistic method of Barinova et al. (2010), as well as benefiting from a computation time that is independent of the number of objects in the scene, and allowing the straightforward refinement of modes using mean shift.

However, given that the kernel density parameters used were optimized for Hough-based approaches and not for BLK, the real "take home" message of this paper is that the assumption that only one vote generated by each feature is correct is a powerful constraint in Hough-based frameworks, which can dramatically improve inference by "clearing the mist" of incorrect votes, as long as the correct vote is chosen well. We also note that several inference approaches outside the Hough domain enforce a similar constraint, that only one object generates each feature, e.g. (Delong et al. 2012a,b; Isack and Boykov 2012; Woodford et al. 2012); these methods may well perform similarly, and potentially even better, on the same problem.

Acknowledgments The authors are extremely grateful to Bob Fisher, Andrew Fitzgibbon, Chris Williams, John Illingworth and the anonymous reviewers for providing valuable feedback on this work.

Appendix Proof of the Integer Nature of Vote Weights

Theorem 1 *Given Eq. (3), an integer set of optimal values of* θ *exists, i.e. for which* $\theta_{ij} \in \{0, 1\}\ \forall i, j$.

Proof Let θ' denote a globally optimal value of θ, *i.e.* one that minimizes Eq. (3). Let us consider only the vote weights of the i^{th} feature, and assume the other vote weights are fixed

at their optimal value, i.e. $\Theta_j = \Theta'_j \; \forall j \neq i$. The objective function can then be written as

$$f(\Theta_i) = -\int_{\mathcal{H}} p(\mathbf{y}|\Theta_i) \ln p(\mathbf{y}|\Theta'_i)\, \mathrm{d}\mathbf{y}, \tag{14}$$

$$p(\mathbf{y}|\Theta_i) = C(\mathbf{y}) + \omega_i \sum_{j=1}^{J_i} \theta_{ij} K(\mathbf{x}_{ij}, \mathbf{y}), \tag{15}$$

where $C(\mathbf{y})$ is a function which is independent of Θ_i. Note that we have further assumed that the instance of Θ_i in the $\ln p(\mathbf{y}|\Theta_i)$ term in Eq. (14) is also at its optimal value, Θ'_i. This allows us to rewrite the objective function as follows:

$$f(\Theta_i) = D - \sum_{j=1}^{J_i} \theta_{ij} a_{ij} \tag{16}$$

$$a_{ij} = \int_{\mathcal{H}} \omega_i K(\mathbf{x}_{ij}, \mathbf{y}) \ln p(\mathbf{y}|\Theta'_i)\, \mathrm{d}\mathbf{y} \tag{17}$$

where D is a constant, as are the values a_{ij}. Given the constraints of Eq. (2), minimizing Eq. (16) with respect to Θ_i, can always be achieved by setting $\theta_{ij} = 1$ for one j for which a_{ij} is largest, and setting all other weights to 0. In addition, Gibbs' inequality Falk (1970) implies that Eq. (14) is minimized when $\Theta_i = \Theta'_i$ (as we require them to be). Therefore the i^{th} feature must have an integer set of optimal weights. This argument can be applied to each feature independently. □

References

Toshiba CAD model point clouds dataset (2011). http://www.toshiba-europe.com/research/crl/cvg/projects/stereo_points.html.

Allan, M., & Williams, C. K. I. (2009). Object localisation using the generative template of features. *Computer Vision and Image Understanding*, *113*, 824–838.

Ballard, D. H. (1981). Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, *13*(2), 111–122.

Barinova, O., Lempitsky, V., & Kohli, P. (2010). *On detection of multiple object instances using Hough transforms*. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Ben-Tzvi, D., & Sandler, M. B. (1990). A combinatorial Hough transform. *Pattern Recognition Letters*, *11*(3), 167–174.

Besag, J. (1986). On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society, Series B*, *48*(3), 259–302.

Birchfield, S., & Tomasi, C. (1999). *Multiway cut for stereo and motion with slanted surfaces*. In: Proceedings of the IEEE International Conference on Computer Vision.

Bober, M., & Kittler, J. (1993). *Estimation of complex multimodal motion: An approach based on robust statistics and Hough transform*. In: Proceedings of the British Machine Vision Conference.

Cheng, Y. (1995). Mean shift, mode seeking, and clustering. *Transactions on Pattern Analysis and Machine Intelligence*, *17*(8), 790–799.

Delong, A., Osokin, A., Isack, H., & Boykov, Y. (2012). Fast approximate energy minimization with label costs. *International Journal of Computer Vision*, *96*(1), 1–27.

Delong, A., Veksler, O., & Boykov, Y. (2012). *Fast fusion moves for multi-model estimation*. In: Proceedings of the European Conference on Computer Vision.

Drost, B., Ulrich, M., Navab, N., & Ilic, S. (2010). *Model globally, match locally: Efficient and robust 3D object recognition*. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 998–1005).

Duda, R. O., & Hart, P. E. (1972). Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, *15*, 11–15.

Falk, H. (1970). Inequalities of J. W. Gibbs. *American Journal of Physics*, *38*(7), 858–869.

Fisher, A., Fisher, R. B., Robertson, C., & Werghi, N. (1998). *Finding surface correspondence for object recognition and registration using pairwise geometric histograms* (pp. 674–686). In: Proceedings of the European Conference on Computer Vision.

Gall, J., & Lempitsky, V. (2009). *Class-specific Hough forests for object detection* (pp. 1022–1029). In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Gerig, G. (1987). *Linking image-space and accumulator-space: A new approach for object-recognition* (pp. 112–117). In: Proceedings of the IEEE International Conference on Computer Vision.

Hough, P.V.C. (1962) Method and means for recognizing complex patterns. U.S. Patent 3,069,654.

Illingworth, J., & Kittler, J. (1987). The adaptive Hough transform. *Transactions on Pattern Analysis and Machine Intelligence*, *9*(5), 690–698.

Isack, H., & Boykov, Y. (2012). Energy-based geometric multi-model fitting. *International Journal of Computer Vision*, *97*(2), 123–147.

Knopp, J., Prasad, M., Willems, G., Timofte, R., & Van Gool, L. (2010). *Hough transform and 3D SURF for robust three dimensional classification* (pp. 589–602). In: Proceedings of the European Conference on Computer Vision.

Lamdan, Y., & Wolfson, H. (1988). *Geometric hashing: A general and efficient model-based recognition scheme* (pp. 238–249). In: Proceedings of the IEEE International Conference on Computer Vision.

Leibe, B., Leonardis, A., & Schiele, B. (2004). *Combined object categorization and segmentation with an implicit shape model*. In: ECCV Workshop on Statistical Learning in Computer Vision.

Leibe, B., Leonardis, A., & Schiele, B. (2008). Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision*, *77*(1–3), 259–289.

Li, H., Lavin, M. A., & Le Master, R. J. (1986). Fast Hough transform: A hierarchical approach. *Computer Vision, Graphics, and Image Processing*, *36*(2–3), 139–161.

MacKay, D. J. C. (2009). *Information theory. Inference and learning algorithms*. Cambridge: Cambridge University Press.

Maji, S., & Malik, J. (2009). *Object detection using a max-margin Hough transform*. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Mian, A., Bennamoun, M., & Owens, R. (2006). Three-dimensional model-based object recognition and segmentation in cluttered scenes. *Transactions on Pattern Analysis and Machine Intelligence*, *28*(10), 1584–1601.

Minka, T. P. (2003). *The 'summation hack' as an outlier model*. Technical note.

Okada, R. (2009). *Discriminative generalized Hough transform for object detection* (pp. 2000–2005). In: Proceedings of the IEEE International Conference on Computer Vision.

Pham, M. T., Woodford, O. J., Perbet, F., Maki, A., Stenger, B., & Cipolla, R. (2011). *A new distance for scale-invariant 3D shape recognition and registration*. In: Proceedings of the IEEE International Conference on Computer Vision.

Rosten, E., & Loveland, R. (2009). *Camera distortion self-calibration using the plumb-line constraint and minimal Hough entropy*. Machine Vision and Applications.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, *27*(379–423), 623–656.

Sheikh, Y. A., Khan, E. A., & Kanade, T. (2007). *Mode-seeking by medoidshifts*. In: Proceedings of the IEEE International Conference on Computer Vision.

Stephens, R. S. (1991). A probabilistic approach to the Hough transform. *Image and Vision Computing*, *9*(1), 66–71.

Toldo, R., & Fusiello, A. (2008). *Robust multiple structures estimation with j-linkage*. In: Proceedings of the European Conference on Computer Vision.

Tombari, F., & Di Stefano, L. (2010). *Object recognition in 3D scenes with occlusions and clutter by Hough voting* (pp. 349–355). In: Proceedings of PSIVT.

Vedaldi, A., & Soatto, S. (2008). *Quick shift and kernel methods for mode seeking* (pp. 705–718). In: Proceedings of the European Conference on Computer Vision.

Vincent, E., & Laganiere, R. (2001). *Detecting planar homographies in an image pair* (pp. 182–187). In: Proceedings of the International Symposium on Image and Signal Processing and Analysis.

Vogiatzis, G., & Hernández, C. (2011). Video-based, real-time multi view stereo. *Image and Vision Computing*, *29*(7), 434–441.

Woodford, O. J., Pham, M. T., Maki, A., Gherardi, R., Perbet, F., & Stenger, B. (2012). *Contraction moves for geometric model fitting*. In: Proceedings of the European Conference on Computer Vision.

Xu, L., Oja, E., & Kultanen, P. (1990). A new curve detection method: Randomized Hough transform (RHT). *Pattern Recognition Letters*, *11*(5), 331–338.

Zhang, W., & Kosecká, J. (2007). Nonparametric estimation of multiple structures with outliers. In R. Vidal, A. Heyden, & Y. Ma (Eds.), *Dynamical Vision, Lecture Notes in Computer Science, vol. 4358* (pp. 60–74). Heidelberg: Springer.

Zhang, Y., & Chen, T. (2010). *Implicit shape kernel for discrimintative learning of the Hough transform detector*. In: Proceedings of the British Machine Vision Conference.

Zuliani, M., Kenney, C. S., & Manjunath, B. S. (2005). *The multi-RANSAC algorithm and its application to detect planar homographies*. In: Proceedings of the IEEE International Conference on Image Processing.