MEAN-SHIFT FOR STATISTICAL HOUGH TRANSFORM

BY ROZENN DAHYOT,
Trinity College Dublin

This paper presents a kernel density estimate of the Hough Transform [1, 2] for inference of hyperplanes in a multidimensional space. This new analytical representation allows us to define a specific nonlinear Mean shift algorithm to detect the maxima of the distribution.

1. Introduction. The Hough Transform is a well known robust technique to infer shapes from a set of spatial points [3–6]. Having a parametric form of the pattern of interest w.r.t. a latent variable Θ, the Standard Hough Transform computes an estimate of the density function of Θ using a histogram. Maxima of the histogram are then located to infer the instance(s) of the shape of interest. This generic approach has been successfully used in many domains such as image and video processing [7], astronomy [8] or geoscience [9].

The problems of using multi-dimensional histograms are well-known. The trade off in between the number of bins in the histogram and the number of available observations is crucial. Too many bins for too few observations would lead to a memory consuming and sparse representation of the probability density $p_\Theta(\Theta)$. In addition, too few bins would reduce the resolution in the $\Theta$–space and therefore limit the precision of the estimates.

To overcome these limitations, we have introduced a kernel modelling of the Hough transform, called Statistical Hough Transform (SHT) [1, 2]. This new representation is smooth and continuous. This density function can be computed on a fine grid on the $\Theta$–space, and the maxima can then be searched for to infer the shape of interest. This is illustrated in [1] for finding lines. This approach is very accurate depending on the resolution of the grid chosen for computing $p_\Theta(\Theta)$ but can be memory demanding in particular when dealing with high dimensional spaces.

In this paper, we extend the Statistical Hough Transform for inference of hyperplanes in multi-dimensional spatial domain $\mathbb{R}^d$. We introduce a new gradient ascent (i.e. mean shift) algorithm to estimate the maxima of $p_\Theta(\Theta)$ without computing this density function everywhere in the $\Theta$–space. Since the density function of $\Theta$ is nonlinear, the standard Mean Shift approach

Keywords and phrases: Hough transform, Radon transform, kernel probability density function, Mean shift, Riemannian manifold, Markov chain
is not applicable (e.g. [10, 11]). Our approach is then taking advantage of recent advances on non-linear mean shift over Riemannian manifolds [12].

Starting from an initial guess in the $\Theta$–space, we define a Markov chain that converges to one maxima of $p_{\Theta}(\Theta)$. We illustrate this algorithm to infer lines in $\mathbb{R}^2$ and planes in $\mathbb{R}^3$, and compare it to the standard Principal Component Analysis (PCA). However, contrary to PCA, SHT is robust and is able to deal with multiple solutions. This is illustrated by recovering multiple lines in an image.

2. Statistical Hough Transform.

2.1. Notations and hypotheses. 

Having a set of observations $\mathcal{S}_x = \{x_i\}_{i=1,\ldots,N}$ of a random vector $x \in \mathbb{R}^d$, we assume that $x$ is related to the latent variables $\Theta = (\rho, \theta)$ by the normal equation:

$$\rho - x^T n = 0 \quad (\text{hyperplanes})$$

where $n$ is a vector orthogonal to the hyperplane with unit norm that can be defined with $d - 1$ angles $\theta = (\theta_1, \ldots, \theta_{d-1})$:

$$n = \begin{pmatrix}
\prod_{j=1}^{d-1} \cos \theta_j \\
\sin \theta_1 \prod_{j=2}^{d-1} \cos \theta_j \\
\sin \theta_2 \prod_{j=3}^{d-1} \cos \theta_j \\
\vdots \\
\sin \theta_{d-2} \cos \theta_{d-1} \\
\sin \theta_{d-1}
\end{pmatrix}$$

For symmetry reason ($(\rho, n)$ and $(-\rho, -n)$ define the same hyperplane), it is enough to limit the angles to $\theta_j \in [-\pi/2, \pi/2]$ to define hyperplanes in a unique fashion. The vector $\theta = (\theta_1, \ldots, \theta_{d-1})$ and scalar $\rho$ form the latent vector $\Theta = (\rho, \theta)$ that we wish to infer from the set of observations $\mathcal{S}_x$.

The spatial density function of $x$ is modelled by a mixture of kernels [13]:

$$\hat{p}(x|\mathcal{S}_x) = \sum_{i=1}^{N} \left( \frac{1}{h_i} \right)^d k \left( \frac{\|x - x_i\|}{h_i} \right) p_i$$

For simplicity, we have chosen an isotropic variable bandwidth $h_i$ that represents the standard deviation of the noise on the observation $x_i$. $p_i$ models a prior on the observation $x_i$ that quantifies a level of trust or certainty that may be available a priori. The standard choice is an equiprobable prior...
\[ p_i = \frac{1}{N}, \quad \forall i = 1, \ldots, N. \] However others could be used, subject to the following constraints:

\[ \left( \sum_{i=1}^{N} p_i = 1 \right) \land \left( \forall i, \ 0 \leq p_i \leq 1 \right) \]

Next we show how \( p_\Theta(\Theta) \) is estimated using the relation (2.1) between the dual spaces \( x \)-space and \( \Theta \)-space.

2.2. Inference of the probability density function \( p_\Theta(\Theta) \). Assuming the independence of the angles \( \theta \) and the spatial vector \( x \), the distribution of \( \Theta \) can be estimated by [1, 2]:

\[
\hat{p}_\rho(\rho, \theta|S) = \int_{\mathbb{R}^d} p_\rho(\rho, x|S) \, dx
\]

From equation (2.1), we can write:

\[
p_\rho(\rho|\theta, x) = \delta(\rho - x^T n)
\]

where \( \delta(\cdot) \) is the Dirac density function. The integral (2.5) using (2.6) is the Radon transform [14] of the spatial density function \( \hat{p}_x(x|S) \) defined in equation (2.3). Consequently an estimate of the probability density function of the latent variable \( \Theta \) can be computed [1, 2]:

\[
\hat{p}_\rho(\rho, \theta|S) = p_\theta(\theta) \sum_{i=1}^{N} R_i(\rho, \theta) \, p_i
\]

where the kernel \( R_i(\rho, \theta) \) corresponds to the Radon transform of the spatial kernel \( k \) centred on the observation \( x_i \). For a gaussian kernel:

\[
\left( \frac{1}{h_i} \right)^d k \left( \frac{\|x - x_i\|}{h_i} \right) = \left( \frac{1}{\sqrt{2\pi} h_i} \right)^d \exp \left( -\frac{\|x - x_i\|^2}{2 h_i^2} \right)
\]

it can be shown that the kernel created in the \( \Theta \)-space is [1, 14]:

\[
R_i(\rho, \theta) = \frac{1}{\sqrt{2\pi} h_i} \exp \left( -\frac{(\rho - x_i^T n)^2}{2 h_i^2} \right)
\]

This gaussian kernel is used in the following. In addition, the prior \( p_\theta(\theta) \) is chosen uniform: \( p_\theta(\theta) = \left( \frac{1}{\pi} \right)^{d-1} \). The probability density function of the latent variable \( \Theta \) is then estimated by:

\[
\hat{p}_\rho(\rho, \theta|S) = \left( \frac{1}{\pi} \right)^{d-1} \sum_{i=1}^{N} R_i(\rho, \theta) \, p_i
\]
3. Mean shift. A standard method to find the maxima of $p_{\Theta}(\Theta)$ is to use the gradient ascent Mean Shift algorithm. Starting from an initial guess in the $\Theta$–space, it is an iterative procedure that converges towards the nearest local maximum. It has been used with great success for clustering in applications in computer vision [10]. It is a standard approach for stochastic exploration using gradient methods [15].

The derivative of the p.d.f. (2.7) needs first to be computed to define the Mean shift algorithm in our case. For simplicity, the derivatives of the Radon transform are computed w.r.t. $\rho$ and $n$ instead of $\theta$ [14] (see section 3.1). The kernel in the $\Theta$–space, now rewritten $R_i(\rho, n)$, is defined on $\mathbb{R}^1 \times S^{d-1}$ where $S^{d-1}$ is a unit hypersphere:

\begin{equation}
S^{d-1} = \{ n \in \mathbb{R}^d : ||n|| = 1 \}
\end{equation}

To update $n$ (or $\theta$), we need to define a nonlinear Mean Shift step over the Riemannian manifold $S^{d-1}$ [12] (see 3.3). A contrario, the variable $\rho$ is updated on the linear space $\mathbb{R}^1$ and this is explained in section 3.2.

3.1. Derivatives of $R_i(\rho, n)$. The derivative of the gaussian kernel $R_i(\rho, \theta)$ w.r.t. $\rho$ is:

\begin{equation}
\frac{\partial R_i(\rho, n)}{\partial \rho} = \left( \frac{-(\rho - x_i^T n)}{h_i^2} \right) R_i(\rho, n)
\end{equation}

The derivatives w.r.t. $n$ are:

\begin{equation}
\frac{\partial R_i(\rho, n)}{\partial n_k} = -x_{ik}(\rho - x_i^T n) h_i^{-2} R_i(\rho, n) \quad \forall k = 1, \ldots, d
\end{equation}

3.2. Mean shift step in $\mathbb{R}^1$. Assuming an initial guess $(\rho^{(m)}, \theta^{(m)})$ (or $(\rho^{(m)}, n^{(m)})$) available and using the derivative w.r.t. $\rho$ (equation (3.2)), the mean shift update for the variable $\rho$ is defined as [11]:

\begin{equation}
\rho^{(m+1)} = \frac{\sum_{i=1}^N \left( \frac{x_i^T n^{(m)}}{h_i^2} \right) R_i(\rho^{(m)}, n^{(m)}) p_i}{\sum_{i=1}^N \frac{1}{h_i^2} R_i(\rho^{(m)}, n^{(m)}) p_i}
\end{equation}

If the denominator is equal to zero (at computer precision level), then the update is simply:

\begin{equation}
\rho^{(m+1)} = \rho^{(m)}
\end{equation}
3.3. **Mean shift step in** $\mathbb{S}^{d-1}$. Similarly we can find the following updates for $\mathbf{n}$ using the derivatives in equation (3.3). In that case we get the following linear system to solve:

\[
A^{(m)} \mathbf{n}^{(m+1)} = \mathbf{b}^{(m)}
\]

The column vector $\mathbf{b}^{(m)} = [b_k]$ is defined as:

\[
b_k = \rho^{(m)} \sum_{i=1}^{N} \frac{x_{ik}}{h_i^2} R_i(\rho^{(m)}, \mathbf{n}^{(m)}) \ p_i \quad \forall k = 1, \ldots, d
\]

$A^{(m)} = [a_{kj}]$ is a $d \times d$ square symmetric matrix defined by:

\[
a_{kj} = \sum_{i=1}^{N} \frac{x_{ik} x_{ij}}{h_i^2} R_i(\rho^{(m)}, \mathbf{n}^{(m)}) \ p_i
\]

Note that the matrix $A^{(m)}$ is a weighted covariance matrix of the random vector $\mathbf{x}$ in $\mathbb{R}^d$ that can be rewritten as:

\[
A^{(m)} = \mathbf{X}^T \mathbf{D}^{(m)} \mathbf{X}
\]

with the $(N \times d)$ matrix $\mathbf{X} = [x_{ik}]$ of the observations, and assuming that $p_i > 0, \forall i$ (no observation is discarded), then $D^{(m)}$ is the $(N \times N)$ positive diagonal matrix with elements $D_{ii} = R_i(\rho^{(m)}, \mathbf{n}^{(m)}) \frac{p_i}{h_i}$.

By definition in equation (3.8), the matrix $A^{(m)}$ is symmetric positive-semidefinite. Providing that $A^{(m)}$ is not singular in the linear system (3.6), we can then compute the following update:

\[
\mathbf{n}^{(m+1)} = (A^{(m)})^{-1} \mathbf{b}^{(m)}
\]

$A^{(m)}$ can be singular however and in this case the update is computed by:

\[
\mathbf{n}^{(m+1)} = \mathbf{n}^{(m)}
\]

Because the update (3.10) of $\mathbf{n}^{(m+1)}$ is computed as a linear combination on the tangent space, it does not belong to the manifold $\mathbb{S}^{d-1}$. $\mathbf{n}^{(m+1)}$ is therefore projected onto $\mathbb{S}^{d-1}$ by dividing it by its norm $\|\mathbf{n}^{(m+1)}\|$ [12]. Using the relation (2.2) between $\mathbf{n} = (n_1, \ldots, n_d)^T$ and $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_{d-1})^T$, each angle of the vector $\boldsymbol{\theta}$ can be inferred in $[-\pi/2; \pi/2]$ by:

\[
\begin{cases}
\theta_1 = \arctan \left( \frac{n_2}{n_1} \right) \\
\theta_j = \arctan \left( \frac{n_{j+1}}{\sqrt{\sum_{k=1}^{j} n_k^2}} \right) \quad \forall j = 2, \ldots, d-1
\end{cases}
\]
Remember that \((-\rho, -\mathbf{n})\) are also possible parameters for the hyperplane \(\rho - \mathbf{x}^T \mathbf{n} = 0\). So before converting \(\mathbf{n}\) into \(\theta\) using 3.12, we ensure first that \(n_1 \geq 0\). If \(n_1 < 0\) then \(\mathbf{n}\) is updated to \(-\mathbf{n}\) and \(\rho\) is changed in \(-\rho\).

The proposed algorithm can be summarized as follows. Starting from an initial position \((\rho^{(1)}, \mathbf{n}^{(1)})\) in the \(\Theta\)-space, the following steps are repeated until convergence:

1. Update \(\rho^{(m)}\) by \(\rho^{(m+1)}\) using equation (3.4) or (3.5).
2. Compute \(A^{(m)}\) and \(b^{(m)}\) by equations (3.8) and (3.7):
   (a) if \(A^{(m)}\) is not singular compute \(\mathbf{n}^{(m+1)} = (A^{(m)})^{-1} b^{(m)}\)
   (b) if \(A^{(m)}\) is singular then \(\mathbf{n}^{(m+1)} = \mathbf{n}^{(m)}\)
3. If the first component of the vector \(\mathbf{n}^{(m+1)}\), \(n_1^{(m+1)}\), is negative then change \((\rho^{(m+1)}, \mathbf{n}^{(m+1)})\) to \((-\rho^{(m+1)}, -\mathbf{n}^{(m+1)})\).
4. Project \(\mathbf{n}^{(m+1)}\) onto \(S^{d-1}\) by dividing \(\mathbf{n}^{(m+1)}\) by its norm \(\|\mathbf{n}^{(m+1)}\|\). The update \(\theta^{(m+1)}\) can then be inferred using (3.12).

Convergence is assumed when the probability \(\hat{p}_{\rho\theta}(\rho^{(m)}, \theta^{(m)})\) is equal to \(\hat{p}_{\rho\theta}(\rho^{(m+1)}, \theta^{(m+1)})\) at the computer precision level. The sequence \(\{(\rho^{(m)}, \theta^{(m)})\}_{m=1, \ldots, \infty}\) defines a Markov chain [15] that converges to a local maximum. In the following, the iteration of the Mean shift algorithm is noted \(f(\rho^{(m)}, \theta^{(m)}) = (\rho^{(m+1)}, \theta^{(m+1)})\), and starting from an initial position, the solution after convergence is noted:

\[
(3.13) \quad f^{(\infty)}(\rho^{(1)}, \theta^{(1)}) = (\hat{\rho}, \hat{\theta})
\]

with \(f^{(\infty)} = f \circ f \circ \cdots \circ f\).

4. Stochastic exploration of \(p_{\rho\Theta}(\Theta)\). One method for finding the maxima is to compute the probability density function \(p_{\rho\Theta}(\Theta)\) on a fine grid and pick the maxima [1]. However, this approach is memory demanding especially in high dimensional spaces. The previous section has defined an iterative algorithm that moves an initial guess in the \(\Theta\)-space to a local maximum. So having a set of initial positions in the \(\Theta\)-space \(\{(\rho_j^{(1)}, \theta_j^{(1)})\}_{j=1, \ldots, M}\), the Mean Shift procedure computes the set of their associated maxima: \(\{(\hat{\rho}_j, \hat{\theta}_j) = f^{(\infty)}(\rho_j^{(1)}, \theta_j^{(1)})\}_{j=1, \ldots, M}\). Note that after convergence, some positions coincide and define the estimates (or classes) of interest [10].

In the standard application of Mean shift [10], the observations are used as initial positions. However here, the observations available in the set \(\mathcal{S}_x\)
are in the $x$–space and not in the $\Theta$–space. The $\Theta$–space is bounded [1] and random sampling can be used to create the starting positions:

- $\rho^{(1)}$ is sampled uniformly on the interval $[-\rho_{\text{max}}; \rho_{\text{max}}]$.
- We use Muller's method [16] to generate a uniformly distributed vector $n^{(1)}$ on the hypersphere $S^{d-1}$. We ensure that $n^{(1)}$ is on the right half $S^{d-1}$ by changing $n^{(1)}$ to $-n^{(1)}$ if its first coordinate $n_{1}^{(1)}$ is negative. Then relation (3.12) can be used to compute the corresponding $\theta^{(1)}$.

In the following we initialise the starting positions either on a regular grid spanning the $\Theta$–space or by random sampling.

5. Illustrations. We illustrate our approach for the inference of lines in $\mathbb{R}^2$ and the inference of planes in $\mathbb{R}^3$. The bandwidth is chosen $h_i = h, \forall i = 1, \cdots, N$. At the exception of paragraph 5.4, the priors $p_i$ are chosen equiprobable $p_i = \frac{1}{N}, \forall i$.

Paragraph 5.1 illustrates the convergence of the algorithm and the role of the bandwidth $h$ is highlighted. Section 5.2 proposes to compare Principal Component Analysis and Statistical Hough Transform for the inference of one plane in $\mathbb{R}^3$. We propose a simulated annealing scheme in section 5.3 to find the global maximum of $p_\Theta(\Theta)$, and show the robustness of SHT compared to PCA. At last, SHT is applied to multiple line detection in the edges of images in paragraph 5.4.

5.1. Convergence of Mean shift algorithm. In figure 1, only two spatial points are observed in $\mathbb{R}^2$ (or $N = 2$):

$$S_x = \{x_1 = (20, 10), x_2 = (50, 20)\}$$

Only one maximum is to expect on the density function $\hat{p}_\Theta(\Theta)$ (any 2 points passes only one straight line). The mean shift iteration is illustrated in fig. 1 (a) where in just two iterations the maximum is reached (shown on a contour plot of $\hat{p}_\Theta(\Theta)$). Figure 1 (b) shows the density $\hat{p}_\Theta(\Theta)$ as a grey level map (dark areas indicates high probability), and the blue points on the grid are all the starting positions in the $\Theta$–space and moved using our Mean shift algorithm. The yellow circle indicates the final position $(\hat{\rho}, \hat{\theta})$ to which they all have converged.

A second simulation is run using three observations

$$S_x = \{x_1 = (20, 10), x_2 = (50, 20), x_3 = (35, 30)\}$$

Figure 2 presents the density function $\hat{p}_\Theta(\Theta)$ as contour plots, and the estimated lines in the $x$–space after convergence of the Mean shift algorithm.
Several bandwidths have been used and one can notice that the number of maxima detected decreases as the bandwidth increases. The bandwidth can be interpreted as having the role of the temperature in simulated annealing [15, 17] and a mean shift algorithm with decreasing bandwidth has been proposed by Shen et al. [17] to find the global maximum of a kernel density function. A similar strategy is used in paragraph 5.3.

5.2. Inferring planes in $\mathbb{R}^3$ with comparison with PCA. In this experiment, $N$ observations are generated on a plane of parameters $(\rho, \theta_1, \theta_2) = (10, \pi/3, -\pi/3)$ with additional centred gaussian noise. These observations are created as follow:

1. $x_1$ and $x_2$ are both uniformly distributed on $[-5; 5]$.
2. $x_3$ is computed by:

$$x_3 = \left(10 - x_1 \cos \left(\frac{\pi}{3}\right) \cos \left(-\frac{\pi}{3}\right) - x_2 \sin \left(\frac{\pi}{3}\right) \cos \left(-\frac{\pi}{3}\right)\right) / \sin \left(-\frac{\pi}{3}\right)$$

3. Normally distributed noise with mean zero and standard deviation $\sigma = .5$ is added on all components $(x_1, x_2, x_3)$.

Principal component Analysis (PCA) can also be used to estimate the parameters $(\rho, n)$. The eigenvector associated with the lowest eigenvalue of the covariance matrix of the data in $S_X$, is an estimate of $n$. It is noted $n_{pca}$ and $\rho$ is estimated by $\rho_{pca} = n_{pca}^T \overline{x}$ with $\overline{x}$ the mean of the observations in $S_X$. Using relation (3.12), $\rho_{pca}$, and $\theta_{pca}$ is computed from $\rho_{pca}$, $n_{pca}$.

The Mean-shift algorithm is computed, starting from initial positions on a grid in the $\Theta$–space. The intervals of the grid are $\delta_{\rho} = 5$ and $\delta_{\theta_1} = \ldots$

Fig 1. Example of using Mean Shift iterations on a set of two observations in $\mathbb{R}^2$ ($h = 10$).
$\delta_{\theta_2} = 0.5$ (in radians). The 294 starting position are reported in table 1. We can note that no starting point is exactly on the expected solution $(\rho, \theta_1, \theta_2) = (10, 1.0472, -1.0472)$. After convergence, the parameters $(\rho, \theta)$ with the highest probability $p_{\theta} (\rho, \theta | S_x)$, are kept as final estimates $(\rho_{sht}, \theta_{sht})$.

<table>
<thead>
<tr>
<th>$\theta_1, \theta_2$</th>
<th>-1.5708</th>
<th>-1.0708</th>
<th>-0.5708</th>
<th>-0.0708</th>
<th>0.4292</th>
<th>0.9292</th>
<th>1.4292</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho$</td>
<td>-14</td>
<td>-9</td>
<td>-4</td>
<td>1</td>
<td>6</td>
<td>11</td>
<td></td>
</tr>
</tbody>
</table>

**Table 1**

Values of the grid of starting positions for $(\rho, \theta_1, \theta_2)$ to estimate the plane in $\mathbb{R}^3$. This corresponds to 294 initial guesses that are updated in the $\Theta$-space with the Mean-shift procedure.

The main difference between SHT and PCA lies in what density function they are maximising. We note the errors:

\[
\epsilon_i = \rho - x_i^T \mathbf{n}, \quad \forall i = 1, \cdots, N
\]

Then

- $(\rho_{pca}, \mathbf{n}_{pca})$ is estimated by minimizing the mean square error. Assuming a Normal distribution of these (independent) errors then we can also say that:

\[
(\rho_{pca}, \mathbf{n}_{pca}) = \arg \max_{\rho, \mathbf{n}} \left\{ p(\epsilon_1, \cdots, \epsilon_N) = \prod_{i=1}^N \frac{1}{\sqrt{2\pi h}} \exp \left( -\frac{\epsilon_i^2}{2h^2} \right) \right\}
\]

\[
= \arg \min_{\rho, \mathbf{n}} \left\{ -\log (p(\epsilon_1, \cdots, \epsilon_N)) \propto \sum_{i=1}^N \epsilon_i^2 \right\}
\]

Note that the parameter $h$ does not need to be known to perform the estimation.

- Looking back at equations (2.9) and (2.8), the mean shift algorithm aims at computing the estimate $(\rho_{sht}, \mathbf{n}_{sht})$ such that:

\[
(\rho_{sht}, \mathbf{n}_{sht}) = \arg \max_{\rho, \mathbf{n}} \left\{ \hat{p}_\theta (\rho \theta) = \left( \frac{1}{\pi} \right)^{d-1} \sum_{i=1}^N \frac{1}{\sqrt{2\pi h}} \exp \left( -\frac{\epsilon_i^2}{2h^2} \right) \right\}
\]

In that case, the bandwidth $h$ needs to be known and its choice will impact on the estimate. In the case when $h \to \infty$, using Taylor expansion
near 0 for the exponential function, the maximisation becomes:

\[(\rho_{sht}, n_{sht}) \simeq \arg \max_{\rho, n} \left\{ \hat{p}_{\rho\theta}(\rho, \theta) = \left( \frac{1}{\pi} \right)^{d-1} \sum_{i=1}^{N} \frac{1}{\sqrt{2\pi h}} \left( 1 - \frac{\epsilon_i^2}{2h} \right) \right\} \]

\[
\simeq \arg \min_{\rho, n} \left\{ -\hat{p}_{\rho\theta}(\rho, \theta) \propto \sum_{i=1}^{N} \epsilon_i^2 \right\}
\]

In other words, when the bandwidth (or temperature) is large, the estimate computed with the Statistical Hough Transform is equivalent to the one computed by PCA.

We present in table 2 the results of simulations run for \(N = 10\) and \(N = 100\) observations. For different values of the bandwidth \(h\), we report the average estimates of PCA and SHT computed over 100 runs. Are also reported the mean square error for both methods. Both estimation methods give equivalent results apart when the bandwidth \(h\) is chosen too small and as expected, SHT fails to reach the global maximum (see for \(h = 0.025\)). Note however the stability of the SHT estimation when \(h\) increases.

<table>
<thead>
<tr>
<th>(h)</th>
<th>(N = 10)</th>
<th>(N = 100)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(sht)</td>
<td>(pca)</td>
</tr>
<tr>
<td>(h = 0.025)</td>
<td>(1.10, 1.09, -0.1)</td>
<td>(9.92, 1.06, -1.03)</td>
</tr>
<tr>
<td>(h = 0.5)</td>
<td>(9.7, 1.02, -1.02)</td>
<td>(9.93, 1.06, -1.042)</td>
</tr>
<tr>
<td>(h = 2)</td>
<td>(10.03, 1.08, -1.058)</td>
<td>(9.96, 1.08, -1.046)</td>
</tr>
<tr>
<td>(h = 5)</td>
<td>(9.98, 1.04, -1.051)</td>
<td>(9.91, 1.04, -1.040)</td>
</tr>
<tr>
<td>(h = 40)</td>
<td>(10.02, 1.06, -1.055)</td>
<td>(9.94, 1.06, -1.042)</td>
</tr>
</tbody>
</table>

**Table 2**

*Estimation accuracy of the SHT estimator with comparison to PCA. The true plane parameters are \((10, 1.0472, -1.0472)\).*

We repeated the experience with initial guesses generated by random sampling in the \(\Theta\)-space. The bandwidth is chosen identical to the standard deviation of the noise on the \(N = 10\) observations \(S_x\) (i.e. \(h = 0.5\)). In the figure 3, the estimates \((\rho, \theta_1, \theta_2)\) with standard errors, averaged over 100 runs, are reported for both PCA and SHT. On the abscissa, the number of starting positions varies from 1 to 200 in the SHT algorithm. With only 10 random starting positions, the estimate computed by the SHT gives already very accurate results.
The main advantage of SHT over PCA is its robustness and its ability to propose multiple solutions when several models can explain the observations. This is illustrated in the paragraph 5.4. But first we show next an example of ‘simulated annealing’- like strategy to find the global maximum of $p_\Theta(\Theta)$.

5.3. Finding the Global maximum. In the following experiment, $N_1$ points are distributed on the line $L_1$ of coefficients $(\rho_1 = 10, \theta_1 = \frac{\pi}{4})$ and $N_2$ points are distributed on the line $L_2$ of coefficients $(\rho_2 = -10, \theta_2 = \frac{\pi}{4})$. Note that the two lines $L_1$ and $L_2$ are parallel. More explicitly, for line $L_1$ for instance, the $N_1$ observations in $\mathbb{R}^2$ are created as follow:

1. $x_1$ is uniformly generated on $[-10; 10]$.
2. $x_2$ is computed by:
   $$x_2 = (\rho_1 - x_1 \cos(\theta_1)) / \sin(\theta_1)$$
3. Normally distributed noise with mean zero and standard deviation $\sigma = .5$ is added on all components $(x_1, x_2)$.

The same procedure is used to create the $N_2$ observations on $L_2$. The total number of observations in $S_\chi$ is $N = N_1 + N_2 = 100$. The number of points $N_1$ is changing from $N_1 = 0$ to $N_1 = 100$ and for each value of $N_1$, 100 simulations are run.

The bandwidth is iteratively decreased from $h_0 = 10$ to $h_{\text{min}} = 0.5$ (chosen equal to the noise standard deviation). Instead of the logarithm rate, a geometric rate is chosen for fast convergence [15]:

$$h_k = \alpha^k h_0 \quad \text{until} \quad h_k = h_{\text{min}} \quad \text{with} \quad \alpha = 0.99$$

Following the remark in the previous paragraph, the initial estimate (at $h \to \infty$) for the SHT Mean-shift algorithm is set equal to the PCA solution. The algorithm to find the global maximum can be summarised as follow:

$$\begin{align*}
(\hat{\rho}_0, \hat{\theta}_0) &= (\hat{\rho}_\text{pca}, \hat{\theta}_\text{pca}) \\
(\hat{\rho}_k, \hat{\theta}_k) &= f_k^\infty(\hat{\rho}_{k-1}, \hat{\theta}_{k-1}) \text{ computed with } h_k \\
\text{Until} \quad h_k &= h_{\text{min}}
\end{align*}$$

$f_k^\infty$ corresponds the function $f^\infty$ defined in equation (3.13) and computed with the bandwidth $h_k$.

Figure 4 shows the results of the simulation: the means of the estimated parameters $(\rho, \theta)$ by both SHT (blue curve) and PCA (red curve), and the standard errors of these estimates. The green lines indicate the true parameters $(\rho, \theta)$ of the lines $L_1$ and $L_2$. 
• When $N_1 \in [0;34]$, more points are on the line $L_2$ than $L_1$. SHT with the annealing scheme robustly estimates the coefficients of the main mode corresponding to $L_2$.

• When $N_1 \in [35;61]$, we can note that the SHT estimates are not anymore precise on average (over 100 runs). However, looking at each simulated run, with $N_1 = 45$, 60% of the runs converged to the parameters of $L_2$, while the other 40% failed to converge anywhere interesting (i.e. neither on $L_1$ or $L_2$ parameters), stopped on local maxima.

• When $N_1 \in [62;100]$, more points are on the line $L_1$ than $L_2$. SHT with the annealing scheme robustly estimates the coefficients of the main mode corresponding to $L_1$.

In any case, the PCA estimates are only accurate when only one line occurs (i.e. $N_1 = 0$ and $N_1 = 100$). This experiment illustrates how robust point estimate can be using SHT combined with a simulated annealing scheme using the bandwidth as temperature. This strategy has defined one Markov chain that converges toward the global maximum. On the contrary, in the next section, all local maxima are searched for (the bandwidth remains fixed), and several Markov chains are randomly generated to explore $p_\Theta(\Theta)$. Such a strategy could also be used to find the global maximum by selecting the estimate that maximizes $\hat{p}_\Theta(\Theta)$.

5.4. Application in image processing. As a final example, we illustrate the application of the Mean shift algorithm for line detection in images. Having an image $I(x,y)$, the priors can be chosen to emphasize the edge content of the image. Traditionally, binary priors corresponding to the segmented edges are used in the Standard Hough Transform. Equiprobable priors were used in the Statistical Hough Transform in [1]. As an alternative here, we propose to use the normalised magnitude of the gradient of each pixel:

$$p_i = \frac{\| \nabla I(x_i, y_i) \|}{\sum_{i=1}^{N} \| \nabla I(x_i, y_i) \|} \quad i = 1, \cdots, N$$

The pixel positions $S_x = \{(x_i, y_i)\}_{i=1,\cdots,N}$ are located on a regular grid, and the bandwidth is chosen $h = 1$ to reflect the uncertainty on the position of the pixels [1]. Figure 5(a) shows the image diamond with its priors 5(b) (Similarly with the image heathrow in figure 6(a) and priors 6(b)). Black pixels indicate high priors while white pixels indicate low priors. Figures 5(c) and 6(c) show the density function $\hat{p}_\Theta(\Theta)$ (from above) where dark areas indicate high probability. In both case, several maxima occur corresponding to the straight edges in the images. 100 initial positions are randomly generated in the $\Theta$–space and updated until convergence using our mean
shift algorithm. The values of \( \{ \hat{p}_\Theta(\hat{\Theta}_j) \}_{j=1,\ldots,100} \) are reported in figures 5(d) and 6(d). A few first estimates \( \{ \Theta_j \} \) are selected and shown as red circles in figures 5(d-e) and 6(d-e). The corresponding lines are drawn in figures 5(f) and 6(f). The number of selected maxima is usually chosen by the number of estimates above a given threshold (manually selected here). One can notice that not all lines have been detected and possibly more markov chains should have been used to explore \( \hat{p}_\Theta(\Theta) \).

6. Conclusion. We have presented a new non linear Mean-shift algorithm suited for optimising the Statistical Hough Transform [1]. Our formulation requires an estimate of the bandwidths \( \{ h_i \}_{i=1,\ldots,N} \) and discussion on how to estimate them is available in [1, 13]. In this paper, the bandwidth was chosen the same for all observations and, as proposed by Shen et al. [17], this bandwidth can be iteratively decreased as in a simulated annealing approach to reach the global maximum. No prior on the latent variable \( \Theta \) (or \( n \)) was used in this paper, and future effort will aim at developing more dedicated Mean-shift algorithms with appropriate priors as in [1, 2].

The Hough Transform has been applied for the inference of parametric shapes more diverse than straight lines and hyperplanes [18, 19]. Future work will also aim at extending further this statistical framework to infer other shapes and developed their dedicated Mean-shift algorithm.

References.


---

School of Computer Science and Statistics  
Trinity College Dublin  
Ireland  
E-mail: Rozenn.Dahyot@tcd.ie
Fig 2. Example of the effect of the bandwidth $h$ on the SHT estimates computed on a set of three observations in $\mathbb{R}^2$. 
Fig 3. Estimates with standard errors of $\rho$, $\theta_1$ and $\theta_2$ (from top to bottom) computed with the PCA (red) and SHT (blue) w.r.t. the number of random starting positions generated on $\mathbb{R} \times S^2$ (with $N = 10$ and $h = .5$). These curves are averaged over 100 runs. The green lines are the ground truth.
Fig 4. Robustness of SHT (blue) Vs PCA (red) (see text).
Fig 5. Line detection in image Diamond (a) with priors (b); density function in the \( \Theta \)-space (c) with selected maxima using our Mean-shift algorithm (e) and the corresponding lines (f) in the \( x \)-space. The curve (d) shows the probability density function values of the 100 ‘particules’ randomly created in the \( \Theta \)-space after their convergence with Mean-shift.
Fig 6. Line detection in image Heathrow (a) with priors (b): density function in the $\Theta-$space (c) with selected maxima using our Mean-shift algorithm (e) and the corresponding lines (f) in the $x-$space. The curve (d) shows the probability density function values of the 100 'particules' randomly created in the $\Theta-$space after their convergence with Mean-shift.