Edge Orientation-based Fuzzy Hough Transform (EOFHT)

E. Montseny
ESAII Department
Technical University of Catalonia.
08028 Barcelona, Spain
eduard.montseny@upc.es

***P. Sobrevilla**
Applied Math. II Department
Technical University of Catalonia.
08028 Barcelona, Spain
pilar.sobrevilla@upc.es

P. Marés
ESAII Department
Technical University of Catalonia.
08028 Barcelona, Spain
pere.mares@upc.es

*Corresponding author

Abstract

The *Hough Transform* [6] -HT- is a standard tool in image analysis that allows recognizing global patterns in an image space. To detect shapes in noisy data, preserving the idea of the conventional HT, but allowing detection of approximate shapes, the *Fuzzy Hough Transform* -FHT- was introduced [8].
Based on the FHT way of work, we propose an *Edge Orientation-based Fuzzy Hough Transform -EOFHT-* wherein the information provided by the gradient vector is considered. The use of gradient vector's stability properties allows considering only some relevant orientations, so reducing the computational waste of time.

Keywords: Hough Transform; Fuzzy sets; Straight line detection; Fuzzy Hough Transform; Distributed voting.

1 Introduction

Usually, the most useful/interesting information in an image is that which involves lines, curves and edges. Given an image, which consists of a large number of data points, we would like to see if these represent a particular line or curve and if so, which one in the image.
Humans can easily see patterns made by lines, broken line segments, curves, arcs or even dots. On the other hand, in computer vision, these tasks turn out to be a very complicated problem to be solved.

The *Hough Tr*ansform -HT- have proved to be a good and powerful method to accomplish aforementioned tasks, since it minimizes the deviations of points from the line in a direction perpendicular to the line, and gives good results even if the points are not uniformly distributed along the line.

It is one of the most popular parameter estimation techniques that uses a voting mechanism, and is useful for extracting alignments and arrangements from images. This is why HT has found applicability in areas such as biomedical image processing, scene understanding, target tracking, object inspection and character recognition.

The *Hough transform* has become increasingly popular as a tool for image processing, computer vision, and scene understanding. It was first proposed by P.V.C. Hough [6], and later popularized by Duda and Hart [4]. A generalization of Hough transform, for detecting arbitrary shapes, was proposed by Ballard [1], and much research has be done on the improvement and analysis of it [7].

We can analytically describe a line segment in a number of forms. However, a convenient equation for describing a set of lines uses *parametric* or *normal* notation:

$$x \cos\theta + y \operatorname{sen}\theta = \rho$$

where ρ is the length of a normal from the origin to this line, and θ is the orientation of ρ with regard to the **X**-axis. For any point (x, y) on this line, ρ and θ are constant.

If we plot the possible (ρ, θ) values defined by each (x_i, y_i), points in Cartesian image space map to curves in the polar Hough parameter space. This *point-to-curve* transformation is the Hough transformation for straight lines. When viewed in Hough parameter space, points which are collinear in the Cartesian image space become readily apparent as they yield curves intersecting at a common (ρ_0, θ_0) point.

The transform is implemented by quantizing the Hough parameter space into finite intervals or *accumulator cells* $\mathbf{A}(\rho, \theta)$. As the algorithm runs, each (x_i, y_i) is transformed into a discretized (ρ, θ) curve and the accumulator cells which lie along this curve are incremented.

Lines existing in the image may be detected as high value accumulator cells, and the parameters of the detected line are specified by the *accumulator array* co-ordinates. As a result, line detection in the image is transformed into detection of local maxima in the *accumulator space*, and peaks in the *accumulator array* represent strong *evidence* that a corresponding straight line exists in the image.

However, besides its computation time waste, some problems come up when applying the exact HT, derived from data errors and distortions, image device distortion and, especially, due to random noise. Methods for reducing computational complexity have been investigated by Merlin and Farber [11], and Davis [3].

A modification of the basic HT that allows reducing its computational cost can be made by not allowing feature (edge) points to vote for all possible parametric representations but instead voting only for *those whose representation is consistent with the edge orientation*.

So, instead of casting votes for all lines through a feature point, we may choose to vote only for those with orientation θ in the range $\phi-\alpha \le \theta \le \phi+\alpha$, where ϕ is the *gradient orientation*. This idea can be taken further by weighting the strength of the vote by some function of $|\theta-\phi|$. However, due to the implicit vagueness within gradient orientation obtaining, this approach increases the vagueness of the *strength-of-evidence counter*.

As regards the problems introduced by random noise, one coming up when the exact HT is applied is that there is no difference between a point being far from the ideal line or being just little off it; this is exact voting. This is why, even if a data point is near the shape, it does not contribute to **A** for that (ρ, θ) and, therefore, a noisy or only approximately straight line will not be transformed into a point in the parameter space [8].

To detect shapes in noisy data, preserving the idea of the Hough transform, but allowing detection of approximate shapes, the *Fuzzy Hough Transform* -FHT- was introduced in [8]. The proposed technique finds shapes by approximately fitting the data points avoiding the spurious detected using the classical HT. The method can be achieved by projecting a point to the parameter space followed by a one-dimensional convolution.

To avoid the "*all-or-none*" counting of points as evidence for lines, in the method proposed in [8] by Joon et al., instead of transforming each image point (x_i, y_i) to parameter space, all points within R of (x_i, y_i) have to be transformed, and the values added to the *accumulator cells* $\mathbf{A}(\rho, \theta)$ depend upon w(r), that is a function of distance r. For a suitable choice the authors find a single (ρ, θ) for which $\mathbf{A}(\rho, \theta)$ is maximal. Because of this increase the computation time for the HT, they obtain the same effect by transforming a single point only, by a one-dimensional convolution in the parameter space.

Here we propose an *Edge Orientation-based Fuzzy Hough Transform* -EOFHT- that avoids the *all-or-none* counting of points as evidence for lines, and update the accumulator arrays by means of fuzzy membership values based on the stability, with regard to noise effects, of the gradient vector according to its magnitude [9].

For a better understanding of the algorithm, that is explained in section 3, first of all a brief description of the gradient vector stability property is given in next section.

2. Analysis of illumination gradient vector versus noise gradient vector

As said previously, unlike basic HT and FHT, in the proposed technique points in the **X**x**Y** space will be transformed into points in the parameter space by using their gradient vector representation, taking into account vagueness and problems introduced by noise.

Moreover, in the line of [8], we will avoid the *all-or-none* counting of points. To achieve this aim, for each point (x_i, y_i) in the **X**x**Y** space we will vote for the points (ϕ_i, ρ_i) in the parameter space $-\boldsymbol{\Phi}\text{x}\boldsymbol{P}-$ with orientation in the range $[\psi_i - a, \psi_i + a]$, where ψ_i is the gradient orientation of the point, and *a* is defined as a function of its magnitude (m_i).

As the gradient vector of (x_i, y_i) is obtained through the grey-level's values of the pixels in its neighbourhood, the proposed *Hough Transform* has to take into account that noise added to the image in the digitalization process, produce a random variation in gray-level from pixel to pixel.

So, to decide how and which points contribute in the values added to the *accumulator arrays*, the deviations of gradient vector's components due to errors introduced by noise are considered, based on the analysis of noise effects into the gradient vectors carried out in [9].

2.1 Noise effects within gradient vector components

Following the theory of Gonzalez [5], in [9] Larré and Montseny consider that, once digitalized, the image is represented by a discrete function F that results of adding up two discrete functions, I and N, so that for a given pixel (x_i, y_i) its image illumination value is given by:

$$F(x_i, y_i) = I(x_i, y_i) + N(x_i, y_i) \qquad (1)$$

such that

$I(x_i, y_i)$ is the pixel's ideal illumination value.

$N(x_i, y_i)$ is the error value introduced by noise.

Moreover, the noise error added to each image point ($N(x_i, y_i)$), can be modelled by a Normal random distribution N(0, σ), such that σ is dependent on the quantity of noise in the image.

So, as I and N are unrelated to each other, the gradient vector (*g.v.*) of pixel (x_i, y_i) is

$$\vec{\nabla}F(x_i, y_i) = \vec{\nabla}I(x_i, y_i) + \vec{\nabla}N(x_i, y_i) \qquad (2)$$

This vector addition is graphically represented by:

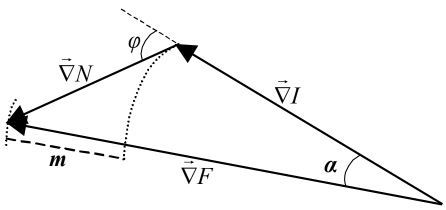

Figure. 1: Representation of vector addition (2), and errors introduced by noise

Looking at figure 1 it is observed that, as a result of the noise both, magnitude and direction of the ideal *g.v.* $\vec{\nabla}I$ are modified by ***m*** and ***α***, respectively.

2.2 Errors' distribution analysis

In the case of an ideal image, *i.e.* in absence of noise), for any point (x_i, y_i) the magnitude and direction of both, ideal and real gradient vectors - $\vec{\nabla}I(x_i, y_i)$ and $\vec{\nabla}F(x_i, y_i)$ - should coincide.

Since no vagueness is introduced by noise, in this case the only point considered in the parameter space $\boldsymbol{\Phi}$x$\boldsymbol{P}$ is the one given by (ψ_i, ρ_i), such that $\rho_i = x_i \cos\psi_i + y_i \sin\psi_i$, with a contribution of 1 to the *accumulator array*.

On the contrary, when dealing with real images, vagueness introduced by noise in the *g.v.* obtaining process has to be considered within the counting of points as evidence for lines, as well as in the updating of the *accumulator array*.

As depicted in figure 1, noise vagueness effects depend on the distribution of errors ***m*** and ***α*** throughout the image. Therefore, these distributions play an important role in the image point to parameter space transformation, as well as in the updating of the *accumulator array* when the *Hough Transform* is based on the use of *g.v.* information. This is why the method here proposed considers errors distributions by making use of the results obtained in [9].

Starting from the information provided by the distribution of direction and magnitude of $\vec{\nabla}N(x_i, y_i)$, in [9] the authors consider the vector representation of figure 1 to deduce the probability density functions of errors ***m*** and ***α***.

So, if φ denotes the deviation of noise gradient vector component $\vec{\nabla}N(x_i, y_i)$ with regard to illumination gradient vector $\vec{\nabla}F(x_i, y_i)$, and the magnitudes of the gradient vector's components are noted by $m_{\vec{\nabla}N} = \left|\vec{\nabla}N\right|$; $m_{\vec{\nabla}F} = \left|\vec{\nabla}F\right|$ and $m_{\vec{\nabla}I} = \left|\vec{\nabla}I\right|$, the joint density function of $\vec{\nabla}N$ with regard to their random polar components is given by:

$$g_{\varphi\, m_{\vec{\nabla}N}}(\varphi, m_{\vec{\nabla}N}) = \frac{m_{\vec{\nabla}N}}{2\pi\cdot\sigma_C^2}\, e^{-m_{\vec{\nabla}N}^2 / 2\sigma_C^2} \qquad (3)$$

such that σ_C is the deviation of the Normal random variables $\partial N/\partial x$ and $\partial N/\partial y$.

Then, using the change of variables:

$$r_1(m_{\vec{\nabla}F}, m_{\vec{\nabla}N}) = arcos\left(\frac{m_{\vec{\nabla}F}^2 - m_{\vec{\nabla}N}^2 - m_{\vec{\nabla}I}^2}{2\cdot m_{\vec{\nabla}I}\cdot m_{\vec{\nabla}N}}\right) =$$

$$= \varphi = s_1(\varphi, \alpha) \qquad (4)$$

$$r_2(m_{\vec{\nabla}F}, m_{\vec{\nabla}N}) = m_{\vec{\nabla}N} = (m_{\vec{\nabla}I}\cdot sin\,\alpha) / sin(\varphi - \alpha) =$$

$$= s_2(\varphi, \alpha) \qquad (5)$$

The marginal density function of magnitude and direction of *g.v.* $\vec{\nabla}F$ - $r_{m_{\vec{\nabla}F}}(m_{\vec{\nabla}F})$ and $r_\alpha(\alpha)$ - are:

$$r_{m_{\vec{\nabla}F}}(m_{\vec{\nabla}F}) = \int_a^b r_{m_{\vec{\nabla}F} m_{\vec{\nabla}N}}(m_{\vec{\nabla}F}, m_{\vec{\nabla}N})\, dm_{\vec{\nabla}N} =$$

$$= \int_a^b g_{\varphi m_{\vec{\nabla}N}}\left(r_1(m_{\vec{\nabla}F}, m_{\vec{\nabla}N}), r_2(m_{\vec{\nabla}F}, m_{\vec{\nabla}N})\right) \left|J_r\right| dm_{\vec{\nabla}N} \qquad (6)$$

$$a = |\, m_{\vec{\nabla}F} - m_{\vec{\nabla}I}\, |, \quad b = |\, m_{\vec{\nabla}F} + m_{\vec{\nabla}I}\, |$$

and

$$r_\alpha(\alpha) = \int_a^\pi s_{\varphi\alpha}(\varphi, \alpha)\, d\varphi =$$

$$= \int_a^\pi g_{\varphi m_{\vec{\nabla}N}}\left(s_1(\varphi, \alpha), s_2(\varphi, \alpha)\right) \left|J_s\right|\, d\varphi \qquad (7)$$

Where the Jacovians of the change of variables, J_r and J_s, are given by:

$$J_r = \frac{-2m_{\vec{\nabla}I}}{\sqrt{\left(2m_{\vec{\nabla}N} m_{\vec{\nabla}I}\right)^2 - \left(m_{\vec{\nabla}F}{}^2 - m_{\vec{\nabla}I}{}^2 - m_{\vec{\nabla}N}{}^2\right)^2}} \qquad (8)$$

$$J_s = \frac{m_{\vec{\nabla}I} \cdot \sin\alpha}{\sin^2(\varphi - \alpha)} \qquad (9)$$

Looking at equations (7) and (9), can be observed that marginal density function $r_\alpha(\alpha)$ depends on the noise level present within the image (σ_C), as well as the magnitude of the ideal illumination function gradient vector ($m_{\vec{\nabla}I}$).

3 Edge Orientation-based Fuzzy Hough Transform -EOFHT-

As proved in [10], g.v. magnitude gives information regarding a possible contour so that the pixels whose gradient vector magnitude is grater than, or equal to, a certain threshold can be classified as contour pixels.Otherwise, with regard to the direction, the gradient vectors of a contour pixel and its neighborhoods must have similar directions.

Moreover, according to [9], for a given noise level - i.e. a σ value- the greater the magnitude of the gradient vector, the smaller the error of the argument is. So, for larger values of the magnitude, gradient vectors in a neighborhood have a more robust distribution, which rely upon orientation.

This fact, together with the marginal density function $r_\alpha(\alpha)$, will be considered in the proposed method for deciding which points we will vote for, and their contribution to the accumulator array **A.**

3.1 Method

As said in previous section, once σ_C and $m_{\vec{\nabla}I}$ are given, the distribution of the argument error $r_\alpha(\alpha)$ is known. Due to $m_{\vec{\nabla}I}$ is unknown we can approximate it by $m_{\vec{\nabla}F}$, and using this approximation the method works as follows:

1. For a fixed noise of σ_C =20 we get, for several values of $m_{\vec{\nabla}F}$, the error distributions $r_{m_{\vec{\nabla}F}}(\alpha)$.
2. For a pair $(m_{\vec{\nabla}F}, r_{m_{\vec{\nabla}F}}(\alpha))$ we get the deviation value σ_α.
3. Given the pairs $(m_{\vec{\nabla}F}, \sigma_\alpha)$, we approximate them by means of a function $f_{\sigma_\alpha}(m_{\vec{\nabla}F})$.
4. For each image point (x_i, y_i) we get its gradient vector components (m_i, ψ_i) -$m_i = m_{\vec{\nabla}F(x_i,y_i)}$-, the corresponding point (ψ_i, ρ_i) in the parameter space, and the deviation $\sigma_{\psi_i} = f_{\sigma_\alpha}(m_i)$.
5. Using the direction ψ_i and the deviation σ_{ψ_i}, we obtain a Normal distribution N(ψ_i, σ_{ψ_i}) the area of which is normalised to 1, so that all the pixels initially detected as contour pixels have the same contribution to the accumulator array **A**.
6. If $\boldsymbol{\mu}_i$ is the distribution function of N(ψ_i, σ_{ψ_i}), we will vote for all the directions ϕ_i such that $\boldsymbol{\mu}_i(\phi_i)>\varepsilon$ (fig. 2).
   Moreover, considering $\rho_i(\phi_i)=x_i \cos\phi_i + y_i\, \mathrm{sen}\phi_i$, the pairs ($\phi_i$,$\rho_i(\phi_i)$) will contribute to the accumulator arrays with membership value $\boldsymbol{\mu}_i(\phi_i)$.

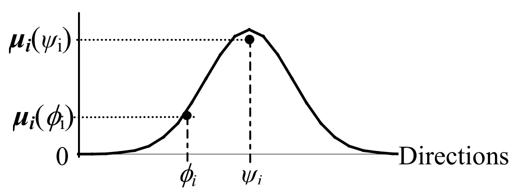

Figure 2: Example of a membership function

7. For each point in the parameter space, we add the values that it has assigned in the accumulator arrays.
8. As in [8], we convolve the parameter space along the ρ direction, with:

$$g(d) = \begin{cases} e^{-d^2/\sigma^2} & \text{if } d < R, \\ 0 & \text{otherwise.} \end{cases} \qquad (10)$$

   Where we have considered $R = \sigma$, and σ=5 so that g be numerically not too steep.
9. In a similar way to [8], we find the maximum peak from the parameter space and obtain parameter values from the position of the peak. Then, using these values, we draw a line to the image plane, and remove data within distance R.

   Go to step 6.

4 Results

In this section we will present one of our experimental results, wherein differences between the resultant accumulator arrays obtained by the HT and EOFHT algorithms are shown.

For the experiment here presented figure 3 is the original image, and view of figure 4 shows the output obtained after applying the Canny edge detector.

The edge points information of figure 4 will be the input to the HT algorithm, and this one together with the provided by the argument of their gradient vectors will be the input to the EOFHT. Then the resultant accumulator arrays, when viewed as an intensity image, are depicted by figures 5(a) and 6(a) respectively. In these images the accumulator space is plotted with θ as the ordinate and ρ as the abscissa, and darker points represent higher accumulation degrees.

To better understand and visualise the improvements brought about by the EOFHT algorithm with regard to the HT method, we can have a look to the figures 5(b) and 6(b).

Glancing over figure 5(b) a lot of discontinuities as well as lack of homogeneity can be observed, what renders the local maxima location an extremely difficult task. On the contrary, the soft transitions depicted in view 6(b) make possible an easier and reliable local maxima location when the EOFHT method is applied.

Figure 3: Original image

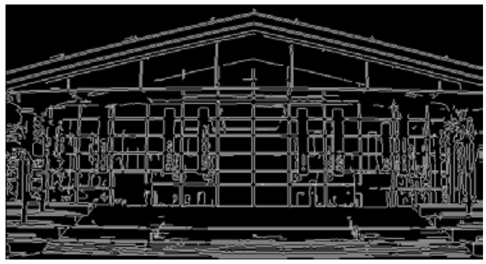

Figure 4: Contours obtained applying the Canny edge detector algorithm to image of figure 3

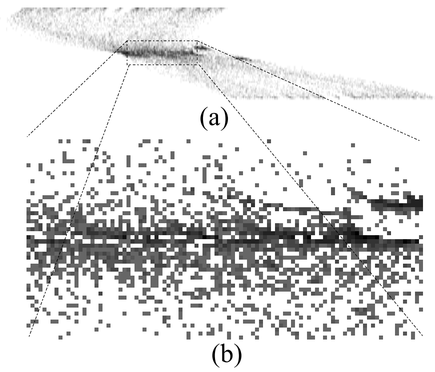

Figure 5: (a) Intensity image representation of the HT accumulator array, wherein darker points are associate to higher degrees. (b) Zoom of an interesting area of (a).

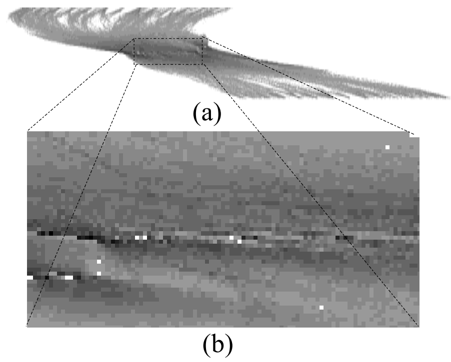

Figure 6: (a) Intensity image representation of the EOFHT accumulator array, wherein darker points are associate to higher degrees. (b) Zoom of an interesting area of (a).

5 Conclusions

In the proposed approach we introduce a technique for detecting lines from real images, taking into account the information provided by the gradient vectors of the pixels. To consider, as long as possible, vagueness introduced by noise within the gradient vector obtaining, we make use of fuzzy techniques.

Although the method here presented follows the idea of [8], the use of gradient vector information allows considering only some orientations, so reducing computational waste of time.

Moreover, we get a reduced accumulator array such that, for every point we consider, in addition to the gradient vector direction, neighbour orientations. As these orientations depend on the reliability of the gradient vector direction, and are determined considering its magnitude, they contribute to the final accumulative array with a smaller strength of evidence degree.

Acknowledgements

This work has been partially supported by the spanish CICYT Project TIC2002-02791.

References

[1] Ballard, D.H. (1981). “Generalizing the Hough Transform to Detect Arbitrary Shapes”. *Pattern Recognition*, vol. 13, no. 2, pp. 111-122.

[2] J.F.Canny, "Finding Edges and Lines in Images", *MIT, Cambridge,Technical Report 720*, June 1983.

[3] Davis, L.S. (1982). “Hierarchical Generalized Hough Transforms and Line-Segment Based Generalized Hough Transforms”. *Pattern Recognition*, vol. 15, no. 4, pp. 277-285.

[4] Duda, R.O. and Hart, P.E. (1972). “Use of the Hough Transform to Detect Lines and Curves in Pictures”. *Comm. ACM,*, vol. 15, no. 1, pp. 11-15.

[5] Gonzalez, R.C. and Wintz P. (1987). “*Digital Image Processing*”. Addison-Wesley Publishing Co.

[6] Hough, P.V.C. (1962). “Methods and Means for recognizing Complex Patterns”. U.S. Patent 3,069,654.

[7] Illington, J. and Kittler, J.(1988). “A survey of the Hough Transform”. *Computer Vision, Graphics, and Image Processing* 44, pp. 87-116.

[8] Joon, H.H., Kóczy, L.T. and Poston, T. (1994). “Fuzzy Hough Transform”. *Pattern Recognition Letters* 15, pp. 649-658.

[9] Larré, A. and Montseny, E. (1994). “A Step Detector Algorithm based on Symbolic Analysis”. *Procc. of the 12th IAPR Conference*, vol. I, Jerusalem, Israel.

[10] Larré, A. and Montseny, E. (1995). “On the Detection of Step Edge Detector Algorithms based on Gradient Vector Analysis”. *Procc. of the 8th ICIAP Conference*, San Remo, Italy.

[11] Merlin, P.M. and Farber, D.J.(1975). “A Parallel mechanism for Detecting Curves in Pictures”. *IEEE Trans. Comput.*, vol. C-24, no. 1, pp. 96-9