AUTOMATIC RECOGNITION OF ROAD SIGNS BY HOUGH TRANSFORM

Vincenzo Barrile, Matteo Cacciola, Giuseppe M. Meduri and Francesco C. Morabito

Universitá Mediterranea degli Studi di Reggio Calabria
Via Graziella Feo di Vito, 89100 Reggio Calabria, Italy
{vincenzo.barrile, matteo.cacciola, morabito}@unirc.it, giumed@libero.it

Commission V, WG V/I

KEY WORDS: Road sign detection, Hough Transform, Road cadastral, Image processing, GIS, Mobile mapping

ABSTRACT:

The problem of road sign detection and recognition is very important in many practical problems, above all for road cadastral authorities. In this sense, an automatic application able to identify the kind of a road sign starting from common imageries such as photos could be very helpful. The main difficulty is due to a possible poor graphical definition of the imagery. In this case, a valid support can be provided by the use of Hough Transform. In this paper, our aim is to propose the implementation of a valid, automatic, robust and reliable decisional support to technicians. It is based on the use of the Standard Hough Transform in order to detect the shape, i.e. the macro-class, of road sign (e.g. circular, squared, triangular, etc.). Subsequently, the road sign characterization has been refined by using the generalization of the Hough Transform in order to detect the specific sign within its previously established macro-class.

1 INTRODUCTION

THE quality and the benefits of the services furnished by a generic Geographical Information System (GIS) do not depend only on the ability of the software component to extrapolate and make explicit the information contained in its databases in implicit form, but above all in the updating the same data bank, especially within the cases characterized by a specific dynamism. It is particularly true for the GIS oriented to the management of the road cadaster. In fact, the elements of the territory which must be represented (the horizontal and vertical system of traffic signs as well as the advertising posters) are characterized by a remarkable variety, involving both their spatial location and the alphanumeric attributes useful in order to their specialization. On the other hand, the integration of the system with a GIS concerning other territorial aspects is made possible only starting from a continuous and constant updating of the data bank related to the road cadaster, so avoiding the incongruence of the results from such an integration (Wolf and DeWitt, 2000). Nowadays, different commercial solutions are available in order to speed up the surveying of interesting elements along the roads. In this way, it is possible to collect a cartographic dataset by exploiting suitable hardware and photogrammetric software tools. The dataset is able to show the records of interesting elements along the roads in a very quick way, if compared to the classical topographic methods (Khotanzad and Zink, 2003). Nevertheless, even if the acquisition step is completely automated, the post-processing procedure is still now affected by a massive intervention of technicians, with a resulting slowdown for the surveying and positioning. Naturally, it becomes a problem in such fields which need to automatically and real-timely work, e.g. if it is necessary fast interventions on roadways with, at the same time, an accurate consideration of the available economic funds (Cotroneo and Barrile, 2006). Within this framework, the proposed approach aims to open a new experimental way by the implementation of a methodology able to restrict the human intervention as much as possible, in order to obtain a self-sufficient, fast and cheap surveying system. It is based on the Hough Transform (HT), a well known methodology used within image processing. It has been created with the aim of line identification into images, and subsequently it has

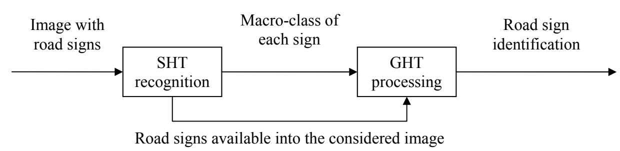


Figure 1: Block schema of proposed approach to recognize road sign in raw images.

been used in scientific literature to find simple geometrical shapes such as triangles, circumferences, rectangles and so on (Gonzalez and Woods, 2002). This simple version, known as Standard HT (SHT), can therefore be used to isolate a road sign into a photographic image by tracking down its shape. Subsequently, an extended version of the SHT, known as Generalized HT (GHT) can be used to discriminate the particular road sign within the set of signs having a specific shape, e.g. the sign ”right-hand bend” within the set of signs which shape is a triangle. For this purpose, a particular self-implemented Matlab© package has been used. We decided to work within Matlab© because of its great potentiality. A furthermore advantage is given by the fact that, if our experimentations retrieve good results, it is easy to export the whole procedure in a suitably compiled Dynamic Link Library (DLL). Figure 1 show a schematic representation of implemented method. In this way, a software tool has been realized, offering some innovative functionalities for the field of use, such as:

- the possibility to recognize entirely in an automatic way the actual elements of interest in the couples of images obtained by classical systems of ”Mobile Mapping”, with a consequent automatic storage of heir positions and the related alphanumeric data

- the exportation of the acquired data towards a GIS, where the last corrections will be carried out in order to respect the topological constraints, according to the already existing cartography.

The paper is structured as follows: Sections 2 and 3 give a brief theoretic description of SHT and GHT respectively; subsequently, the procedure of dataset acquisition and the implementation of proposed algorithm will be described in Section 4. Finally, in Section 5 we discuss obtained results and draw our conclusions.

2 A BRIEF OVERVIEW OF THE SHT

The HT is a feature extraction technique used in digital image processing. The classical transform identifies lines in the image, but it has been extended to identifying positions of arbitrary shapes. To extract features from digital images, it is useful to be able to find simple shapes - straight lines, circles, ellipses and the like - in images. In order to achieve this goal, one must be able to detect a group of pixels that are on a straight line or a smooth curve. That is what a HT is supposed to do (Wikipedia, 2007). Loosely speaking, HT is a 2D non-coherent operator which maps an image to a parameter domain (Morabito et al., 1999). The standard case of HT is a Hough linear transform. To illustrate the idea, let's start with a straight line. When the aim of the analysis is to detect straight lines in an image, the parameter of interest completely defines the straight lines. In the image space, the straight line can be described as $y = mx + b$ and is plotted for each pair of values (x, y). However, the characteristics of that straight line is not x or y, but its slope m and intercept b. Based on that fact, the straight line $y = mx + b$ can be represented as a point (b, m) in the parameter space. Using slope-intercept parameters could make application complicated since both parameters are unbounded: as lines get more and more vertical, the magnitudes of m and b grow towards infinity (Wikipedia, 2007). For computational purposes, however, it is better to parameterize the lines in the HT with two other parameters, commonly called ρ, i.e. the smallest distance between the line and the origin, and θ, i.e. the angle of the locus vector from the origin to this closest point. In this way, the equation (Simone et al., 2001)

$$\rho = x\cos\theta + y\sin\theta \tag{1}$$

maps the point (x, y) into the parameters (ρ, θ), which represent a straight line passing through (x, y). The couple (ρ, θ) is unique if $\theta \in [0, \pi]$ and $\rho \in \mathbf{R}$ or if $\theta \in [0, 2\pi]$ and $\rho \geq 0$. Each pixel in the original image is transformed in a sinusoid in the (ρ, θ) domain. The presence of a line is detected by the location in the (ρ, θ) plane where more sinusoids intersect.

2.1 Implementation of the SHT

SHT algorithm uses an array called accumulator to detect the existence of a line $y = mx + b$. The dimension of the accumulator is equal to the number of unknown parameters of SHT problem. For example, the Hough linear transform problem has two unknown parameters: m and b. The two dimension of the accumulator array would correspond to quantized values for m and b. For each pixel and its neighborhood, SHT algorithm determines if there is enough evidence of an edge at that pixel. If so, it will calculate the parameters of that line, and then look for the accumulator's bin that the parameters fall into, and increase the value of that bin. By finding the bins with the highest value, the most likely lines can be extracted, and their (approximate) geometric definitions read off. The simplest way of finding these peaks is by applying some form of threshold, but different techniques may yield better results in different circumstances - determining which lines are found as well as how many. Since the lines returned do not contain any length information, it is often next necessary to find which parts of the image match up with which lines. For more details, please refer to (Wikipedia, 2007) and references within.

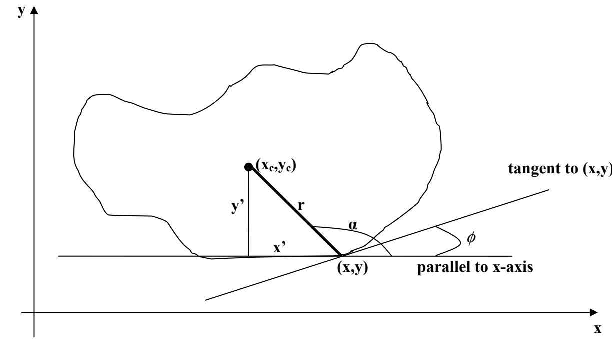


Figure 2: Geometric schema of GHT determination for an object having fixed orientation and size.

3 THE ALGORITHM EXPLOITED TO GENERALIZE THE SHT

The general definition of the GHT is that it's an extension of the SHT, used to detect shapes which cannot be described by simple analytical formulae. Instead, a more complex representation is used, e.g. matrices or other mathematical constructs representing patterns, shapes or vectors. At a computational level the GHT roughly consists in making each examined pixel in an image to "project" a copy of the searched pattern at various angles and scales, (usually, projection and comparison takes place starting from the center of a certain object, but it's possible to start elsewhere under special conditions) and then keeping track of how many pixel matches for a given scale and angle occurred between the "projection" and the tested image. Therefore, the most general algorithm definition says to do exactly that: creating a special reference data structure (usually a binary image, in the form of a table, called R-Table) and essentially comparing its "boundary" or "contour" with groups of pixels (having a central or boundary pixel for reference) (Sonka and et al., 1998). For further details, let us firstly consider to have a fixed orientation and size of in-study object (see Figure 2). Here, we pick a so called reference point (x_c, y_c), where:

$$\begin{aligned} x &= x_c + x' \\ y &= y_c + y' \end{aligned} \tag{2}$$

On the other hand, we know that:

$$\begin{aligned} \cos(\pi-\alpha) &= \frac{y'}{r} \Rightarrow y' = r\cos(\pi-\alpha) = -r\sin(\alpha) \\ \sin(\pi-\alpha) &= \frac{x'}{r} \Rightarrow x' = r\sin(\pi-\alpha) = -r\cos(\alpha) \end{aligned} \tag{3}$$

Combining Equations 2 and 3 we have:

$$\begin{aligned} x_c &= x + r\cos(\alpha) \\ y_c &= y + r\sin(\alpha) \end{aligned} \tag{4}$$

In this way we can compute ϕ, i.e. the perpendicular to gradients direction, and subsequently store the reference point (x_c, y_c) as a function off ϕ: in other words, we can build the R-table. The R-table allows us to use the contour edge points and gradient angle to recompute the location of the reference point. Let us remark that we need to build a separate R-table for each different object. Summarizing, after a quantization of the image space $P[x_{c_{min}}, ..., x_{c_{max}}][y_{c_{min}}, ..., y_{c_{max}}]$, for each edge point (x, y) and using the gradient angle ϕ, we retrieve from the R-table all the (α, r) values indexed under ϕ. Then, for each (α, r), we compute the candidate reference point according to Equation 4.

Figure 3: Frame 4155 at a 800x600 pixel resolution in 8 bit format. In the proposed procedure the RGB format has been used

Now, we increase a suitable counter measuring the votes for the considered reference point. Thus, possible locations of the object contour are given by local maxima in $P[x_c][y_c]$ (Ballard, 1982). For further details about generalization of GHT in case of unfixed size or orientation, please refer to the proposed scientific bibliography.

4 COLLECTION OF DATASET AND EXPERIMENTATIONS

The images exploited for our experimentations have been acquired by using a car equipped by two camcorders. They were controlled by an hodometer (the trigger was set in order to have a sampling equal to a frame every three meters). Moreover, we used a GPS rover to initialize the system and a notebook for data storage. We intentionally did not use inertial sensors for sampling the absolute orientation of the system since our aim is to develop a methodology able to minimize the human intervention within the analysis of acquired data. Moreover, a right georeferring of the same data has not a great importance in this context. Totally, we collected 16 RGB images at a 800x600 pixel resolution (e.g. Figure 3), depicting 30 road signs so divided:

- 17 mandatory direction of travel (MDT) signs (2 turn-left-here mandatory direction of travel, 2 turn-right-here mandatory direction of travel, 3 straight-ahead mandatory direction of travel, 6 straight-ahead or turn-right-here mandatory direction of travel, 4 straight-ahead or turn-left-here mandatory direction of travel);
- 10 yield signs;
- 6 parking signs (3 no parking and 3 no stopping);
- 1 no access;
- 1 stop;
- 1 one way access.

They have been taken into the streets of Reggio Calabria, a city located at the Southern part of Italy, on the Straight of Messina. In a first instance, each image has been passed to the SHT recognition block. This procedure is well described by Figure 4. Let us consider the l^{th} as the input image, which has k road signs. First of all, each image has been decimated in order to speed up

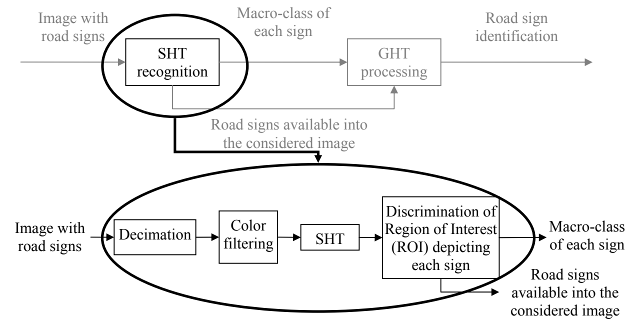


Figure 4: Subschema of the SHT block.

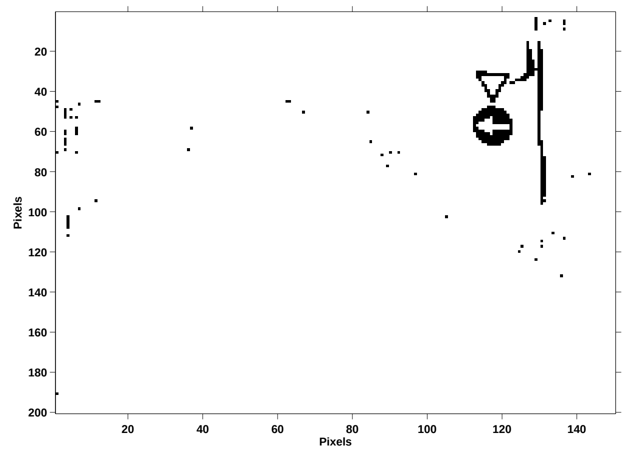
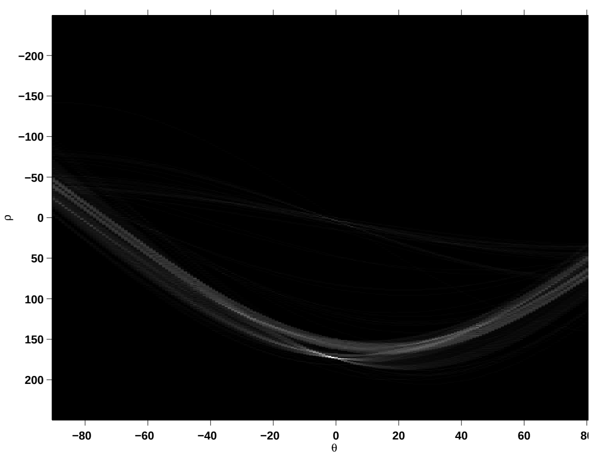


Figure 5: The filtered version of the decimated image shown in Figure 3.

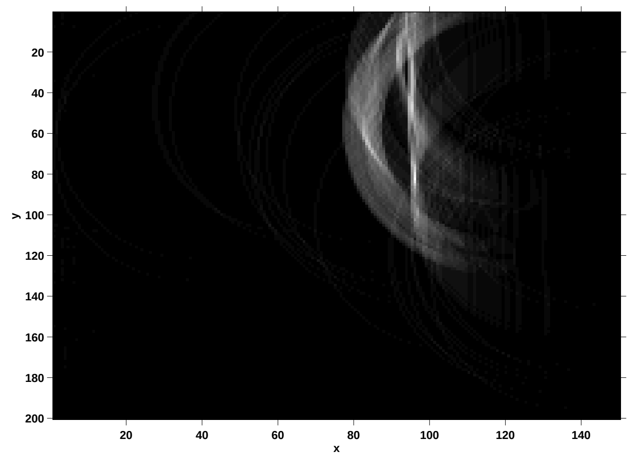
the procedure, so reducing the resolution to 200x150 pixels. In this way, the inspecting area has been reduced, but at the same time the recognition of primitive elements remains practically unchanged since their specificity. Subsequently, the image has been filtered by a self-implemented color filter. In fact, since the borders of road signs usually have the same colors, i.e. red, blue, yellow (we denote this set as ”color domain”), we built a particular filter which compares the color of each pixel of the l^{th} considered image with our ”color domain”. If the pixel's color is a sort of hue of whatever color belonging to the ”color domain”, then the pixel is black colored; otherwise it is white colored. Figure 5 shows the decimated and filtered version of the image in Figure 3. Thus, the SHT can be applied to the considered image, e.g. to frame 4155. Here, it has been possible to recognize the shape of the pictured signs by the following considerations on the accumulator

- for a triangular object, the maximum number of lines that cross 1 point is lower than three times the median value of the number of lines that cross the same point;
- for a squared object, numbers of lines that cross 1 point, 2 points, ..., until n points, in which n is approximately the length of the square, is almost the same;
- for a rounded object, numbers of lines that cross 1 point, 2 points, ..., until n points, where n is approximately the diameter of the rounded object, exponentially increase.

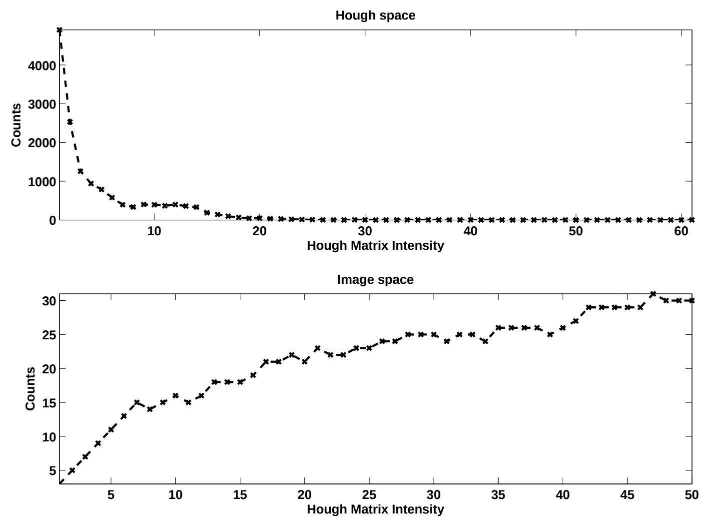
Let us remark how dimensions of objects are unknown a priori, and therefore they must be calculated after the SHT application.



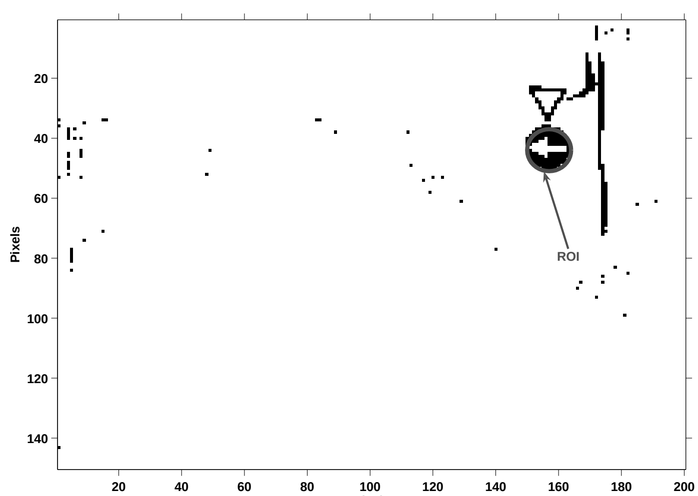
(a) SHT accumulator into the Hough domain



(b) SHT accumulator into the image domain



(c) Intensity of the Hough matrix



(d) Traced ROI for the circular sign

Figure 6: Results obtained by the application of SHT on frame 4155.

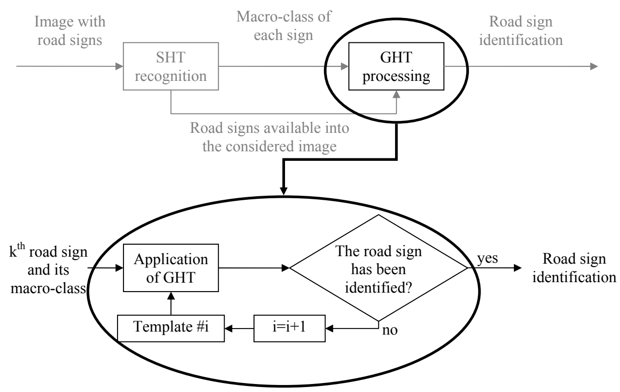


Figure 7: Subschema of the GHT block.

Figure 8: The frame 4157, featuring the misclassified road sign; the error is maybe due to the dimensions of pictured sign.

Results obtained by the application of the SHT to frame 4155 are visible in Figure 6. Thus, it has been possible to discriminate each Region of Interest (ROI) depicting a single road sign within the actually inspected filtered image. Once the shape of the road sign has been detected, the ROI and its macro-class are passed to the GHT sub-block (see Figure 7). Here, the ROI is compared with a set of templates, which are organized and queried according to the macro-class membership. At the end of the procedure, a message identifying the road sign appears to the user.

5 DISCUSSION ABOUT RESULTS AND CONCLUSIONS

In this paper, we deal with the problem of rad sign detection and identification starting from raw images. Actually, it is an open problem with the mobile mapping framework since, even if the image acquisition is automatized now, the interpretation is still deputed to technicians. Consequently, the whole procedure is slowed down with disadvantages in such applications where fast interventions on roadways are necessary, with a particular attention to the expenses. Thus, this kind of matter can be considered as a sort of search problem whose directions of search can be determined either in deterministic or stochastic ways. Our proposed approach, starting from images acquired by commercial camcorders, tends to restrict the search domain by exploiting in a first instance the SHT. In this way it has been possible to recognize the macro-class to which a road sign belongs according to its geometrical shape. Let us remark that the SHT has been ap-

Category	Available signs	Correctly classified	Incorrectly classified
MDT	17	17	0
Yield	10	10	0
Parking	6	6	0
No access	1	1	0
Stop	1	1	0
One way access	1	1	0

Table 1: Results, sorted by kind of signs, of our proposed approach for the road sign classification.

	Available signs	Correctly classified	Incorrectly classified
No parking	3	2	1
No stopping	3	3	1

Table 2: A detail of the parking sign recognition, with the misclassified element.

plied on a filtered image, in which the signs' presence has been enhanced by using a self-implemented filter color. It makes possible to delete the unnecessary information, e.g. buildings, cars, human beings or animals, etc., and consider only the object of interest, i.e. the signs. Subsequently, the usage of the GHT allowed us to identify the specific road sign within its macro-class, by a comparison between the sign detect into the raw image and a collection of templates stored into a previously created ”evaluation archive” (see Figure 7). The whole procedure has been implemented within the Matlab© environment, and so it can be easily compiled as a DLL which can be used in an external GIS tool. The performances of proposed approach are very encouraging: 97% of used road signs have been correctly recognized. Detailed results are shown by Tables 1 and 2. Here, it is possible to denote how a no parking sign has been incorrectly classified as a no stopping sign. Actually we are engaged in optimizing the procedure and linking the obtained DLL with a suitable GIS, in order to evaluate the proposed approach in a real-world application.

REFERENCES

Ballard, D., 1982. Computer vision. Englewood Cliffs, N.J., USA.

Cotroneo, F. and Barrile, V., 2006. Un software per l'aggiornamento automatizzato del catasto strade in ambiente GIS (in Italian language). Bollettino SIFET.

Gonzalez, R. C. and Woods, R. E., 2002. Digital Image Processing. Englewood Cliffs, N.J., USA.

Khotanzad, A. and Zink, E., 2003. Contour line and geographical feature extraction from U.S.G.S. color topographic paper maps. IEEE Trans. on Pattern Analysis and Machine Intelligence 25(1), pp. 18–31.

Morabito, F. C., Simone, G. and Farina, A., 1999. Automated lineament detection in SAR images based on the joint use of wavelet and hough transforms. In: Proc. of 5th International Conference on Radar Systems, Brest, France, p. 1.2.1318.

Simone, G., Morabito, F. C. and Farina, A., 2001. Radar image fusion by multiscale kalman filtering. Technical report, Reggio Calabria, Italy.

Sonka, M. and et al., 1998. Image Processing, Analysis, and Machine Vision. New York, USA.

Wikipedia, 2007. Hough transform. `http://en.wikipedia.org/wiki/Houghtransform`.

Wolf, P. and DeWitt, B. A., 2000. Elements of Photogrammetry with Applications in GIS. New York, USA.