BIULETYN INSTYTUTU AUTOMATYKI I ROBOTYKI WAT
NR 15, 2001

Irregular Colour Pattern Recognition Using the Hough Transform[1]

Witold Żorski, Brian Foxon∗, Johnathan Blackledge∗, Martin Turner∗

Cybernetics Faculty, Military University of Technology
00-908 Warsaw, S. Kaliskiego 2, POLAND

∗ Institute of Simulation Sciences, SERCentre,
Hawthorn Building, De Montfort University, Leicester LE1 9BH

Abstract. This paper presents an application of the Hough Transform to the tasks of learning and identifying irregular patterns in a computer vision system. The method presented is based on the Hough Transform with a parameter space defined by translation, rotation and scaling operations. A fundamental element of this method is the generalisation of the Hough Transform for grey-level and colour images. The technique may be used in a robotic system, identification system or for image analysis.

1. Introduction

The Hough Transform was patented in 1962 as a method for detecting complex patterns of points in a binary image [13]. It introduced the possibility of determining a set of parameters circumscribing the searched pattern. The problem of complex pattern detection in an image is converted into one that searches for local maxima in a parameter space. This method has become very popular. In 1981 it was shown that the Hough Transform is a specific case of the Radon Transform [32] known since 1917.

The Radon Transform is affirmed to be equivalent to the Hough Transform only in the case of binary images. In the case of grey-level or coloured images, the issue is more complicated. Equations defining the Hough Transform limit it to the application of binary images. A set of transforms is often applied that aims to convert the initial image into a binary image with minimal loss of information.

If we consider the equation, which defines the Radon Transform, we may state that there are no limitations for its application directly to grey-level or colour images. However, the

[1] This paper was presented at "IMA Third Conference on Imaging and Digital Image Processing: Mathematical Methods, Algorithms and Applications" (Leicester, 13th - 15th September 2000) and is published with the consent of The Institute of Simulation Sciences at De Montfort University.

following statement is raised - how do we modify the Hough Transform for grey-level or colour patterns? The problem lies in the process of accumulation.

This paper presents an application of the Hough Transform to the tasks of learning and identifying irregular colour patterns in computer vision systems. The elaborated method provides correct results without binarisation of the initial image. It is based on the Hough Transform with a parameter space defined by translation, rotation and scaling operations. A fundamental element of this method is the generalisation of the Hough Transform for grey-level and colour images. The technique simplifies the application of the Hough Transform to pattern recognition tasks as well as accelerating the calculations considerably. The CPU time for the calculations is also very important, and in the case of the test images (size 300x400 pixels) was only a few seconds (Celeron 433MHz).

2. Basic definitions

Let us consider binary **digital images**, i.e. images, which are formed with sets of points, which by convention are either black or white. Such binary images may be represented with the following function [18]:

$$B : D \rightarrow \{0,1\}, \quad \text{where} \qquad D = [1,...,W] \times [1,...,K] \subset N^2 . \tag{1}$$

Hence, we may consider a digital image as a matrix with row and column indices identifying a point in the image.

Given an image B, we can define an **object** $b(B)$ as follows:

$$b(B) = \{(x, y) \in D : B(x, y) = 1\} . \tag{2}$$

Let us denote every image that originated from an image B as the result of restricting its domain D, as a fragment Q of image B. Thus, a **fragment** Q of image B is defined as follows:

$$Q : D_Q \rightarrow \{0,1\}, \quad \text{where} \qquad D_Q \subset D = [1,...,W] \times [1,...,K] \subset N^2 . \tag{3}$$

3. The Hough Transform vs. the Radon Transform

In 1981 Deans noticed [9] that the Hough Transform for straight lines was a specific case of the more general Radon Transform [32], which is defined as (for function $I(x,y)$ in two-dimensional Euclidean space):

$$H(\rho,\alpha)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I(x,y)\,\delta(\rho - x\cos(\alpha) - y\sin(\alpha))dxdy\,, \tag{4}$$

where δ is the delta function. This result shows that the function $I(x,y)$ is integrated along the straight line determined by the parametric equation $\rho = x\cos(\alpha) + y\sin(\alpha)$. The Radon Transform is equivalent to the Hough Transform when considering binary images (i.e. when the function $I(x,y)$ takes values 0 or 1). The Radon Transform for shapes other than straight lines can be obtained by replacing the delta function argument by a function, which forces integration of the image along contours appropriate to the shape.

In order to explain equation (4) in terms of the essence of the Hough Transform, we note that $H(\rho,\alpha)$ represents the value of the Hough Transform at point (ρ,α) in parametric space. This results from the fact that the argument of the delta function is constructed in a way that reflects straight lines. The local maxima in the parameter space $H(\rho,\alpha)$ correspond to straight lines in the image.

Using the Radon Transform to calculate the Hough Transform is simple (almost intuitive) and is often applied in computer implementations. We call this operation pixel counting in the binary image.

An (alternative) interpretation of the Hough Transform is the so-called backprojection method. This method, however, is not clear from the point of view of its direct implementation in the form of a computer algorithm but is exact and commonly applied.

The detection of analytical curves defined in a parametrical way, other than straight lines is quite obvious. Points (x,y) of image lying on the curved line determined by n parameters $a_1,...,a_n$ may be presented in the form:

$$\lambda_O = \{(x,y)\in R^2 : \; g((\hat{a}_1,...,\hat{a}_n),(x,y)) = 0\}\,, \tag{5}$$

where $g((\hat{a}_1,...,\hat{a}_n),(x,y)) = 0$ describes the given curve.

By exchanging the meaning of parameters and variables in the above equation we obtain the backprojection relation (mapping image points into parameter space), which may be written down in the following way:

$$\lambda_T = \{(a_1,...,a_n) \in R^n : g((\hat{x},\hat{y}),(a_1,...,a_n)) = 0\}. \tag{6}$$

From equation (6) the Hough Transform $H(a_1,...,a_n)$ for the image $B(x,y)$ is defined as follows [14]:

$$H(a_1,...,a_n) = \sum_{(x_i,y_i)\in B} h(\hat{x}_i,\hat{y}_i,a_1,...,a_n), \tag{7}$$

where

$$h(\hat{x}_i,\hat{y}_i,a_1,...,a_n) = \begin{cases} 1 & \text{if } g((\hat{x}_i,\hat{y}_i),(a_1,...,a_n)) = 0 \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

In order to calculate the Hough Transform digitally an appropriate representation of the parameter space $H(a_1,...,a_n)$ is required. In a standard implementation, any dimension in the parameter space is subject to quantisation and narrowing to an appropriate range. As a result, an array is obtained where any element is identified by the parameters $(a_1,...,a_n)$. An element in the array is increased by 1 when the analytical curve, determined by co-ordinates $(a_1,...,a_n)$, passes through point $(\hat{x},\hat{y})$ of the object in image B. This process is called **accumulation** and the array used is called an **accumulator (**usually marked with a symbol A). The correct selection of quantisation levels within the parameter space is important since it directly affects the precision of localisation of the searched segments of an image.

Thus, we may assume that the Hough Transform is based on a representation of the image B into the accumulator array A, which is defined as follows:

$$A : P \to N, \quad \text{where} \qquad P = P_1 \times P_2 \times \cdots \times P_p. \tag{9}$$

The symbol $P_i \subset N$ determines the range of i-parameters of a p-dimensional space P. Determining array A is conducted through the calculation of partial values for points of an object $b(B)$ and adding them to the previous ones ($\to$ 7) which constitutes a process of accumulation. Initially, all elements of array A are set to zero.

4. The Binary Hough Transform for irregular objects

The Hough Transform may be successfully applied to detect irregular objects [4], [14]. In the generalised Hough Transform, an object is represented by a pattern, which is a list of boundary points $\{(x_i, y_i) : i = 1,..., n\}$ (without a concrete analytical description), and the parameter space is defined for translation $[x_T, y_T]$, rotation α and scale s of the pattern in the image.

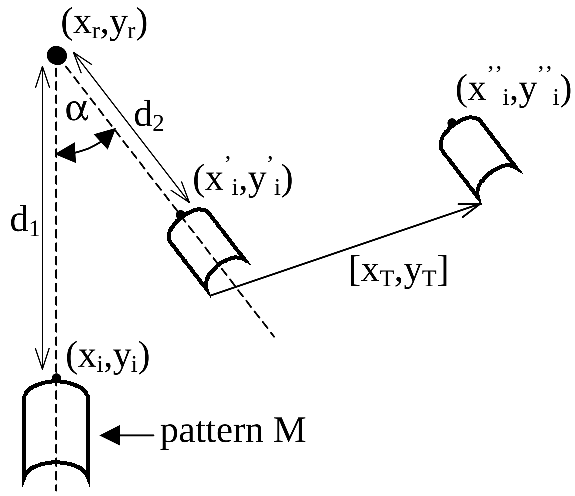

Figure 1. Scaling, rotation and translation of a pattern M with respect to an arbitrary point (x_r, y_r)

Figure 1 shows rotation with respect to an arbitrary point (x_r, y_r) of a given pattern by an angle α with scaling determined by $s = d_2 / d_1$. As a result, a given point (x_i, y_i) of the pattern M is transformed into (x_i', y_i') where the relationships are as follows:

$$\begin{cases} x_i' = x_r + s(x_i - x_r)\cos(\alpha) - s(y_i - y_r)\sin(\alpha) \\ y_i' = y_r + s(x_i - x_r)\sin(\alpha) + s(y_i - y_r)\cos(\alpha) \end{cases} . \quad (10)$$

When a translation operation by a vector $[x_T, y_T]$ is carried out, we have

$$\begin{cases} x_i'' = x_i' + x_T = x_r + s(x_i - x_r)\cos(\alpha) - s(y_i - y_r)\sin(\alpha) + x_T \\ y_i'' = y_i' + y_T = y_r + s(x_i - x_r)\sin(\alpha) + s(y_i - y_r)\cos(\alpha) + y_T \end{cases} . \quad (11)$$

Each point of the image generates an appropriate hypersurface, as a result of backprojection, in parameter space. A number of hypersurfaces that criss-cross a given point (x_T, y_T, α, s) of the parameter space is equivalent to a number of points common for a given object in the image and the fitting pattern. In this case the Hough Transform $H(x_T, y_T, \alpha, s)$ for an image $B(x, y)$ is defined as:

$$H(x_T, y_T, \alpha, s) = \sum_{i=1}^{n} h(x_i, y_i, x_T, y_T, \alpha, s)\,, \qquad (12)$$

where

$$h(x_i, y_i, x_T, y_T, \alpha, s) = \begin{cases} 1 & \text{if } (x_i'', y_i'') \in b(B) \\ 0 & \text{otherwise.} \end{cases} \qquad (13)$$

Fortunately, many problems do not include the issue of pattern scaling. However, direct implementation of equation (12) is inadvisable when scaling must be taken into consideration.

The Hough Transform operation applied directly to irregular objects, which do not require pattern scaling, may be illustrated by means of an example. The parameter space P is defined as [39]:

$$P = P_1 \times P_2 \times P_3 = [1,\ldots,W] \times [1,\ldots,K] \times [0,\ldots,L-1]\,, \qquad (\Delta\alpha = \frac{2\pi}{L})\,. \qquad (14)$$

The example (Figure 2) includes quite a complicated motif. This task is to search for a particular fragment of the motif as indicated. Special attention should be paid to the content of the accumulator. There are many local maxima and when there are a larger number of objects complying with the pattern in the image it is necessary to adopt an appropriate level of acceptance.

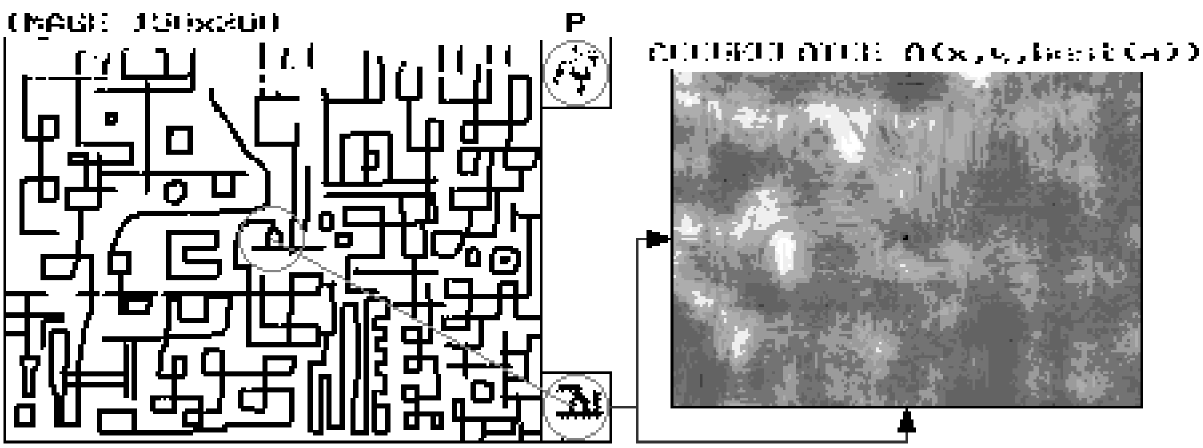

Figure 2. Hough Transform for complex image

5. Generalisation of the Hough Transform for grey-level images

Most existing algorithms apply the Hough transformation to only binary images. Observe that in the case of analysing grey-level images the process of binarisation can loose important information.

The problem lies in the process of accumulation. At the beginning let us try to write the equation (13) in a different variation:

$$h(x_i, y_i, x_T, y_T, \alpha, s) = \begin{cases} 1 & \text{if } B(x_i'', y_i'') = M(x_i, y_i) \\ 0 & \text{otherwise,} \end{cases} \tag{15}$$

where M is the image of the pattern (the task is to identify pattern with an object in image B).

Equation (15) suggests the idea of modifying (13) in the following way:

$$h(x_i, y_i, x_T, y_T, \alpha, s) = 1 - \left| B(x_i'', y_i'') - M(x_i, y_i) \right|. \tag{16}$$

This form tells us what to do in the case of grey-level images. However, we must first define the concept of a grey-level image, an object appearing in such an image and the concept of a grey-level pattern in a computer vision system.

Definitions

An **image with 256 grey levels** means a set of points, which have a value or "shade" from the set $\{0,...,255\}$. Such an image may be presented as:

$$B_G : D \to \{0,...,255\}, \quad \text{where:} \quad D = [1,...,W] \times [1,...,K] \subset N^2. \tag{17}$$

Object $b(B_G)$ in image B_G means any fragment of that image which may be recorded in terms of

$$Q_G : D_Q \to \{0,\ldots,255\}, \quad \text{where:} \quad D_Q \subset D = [1,...,W] \times [1,...,K] \subset N^2. \tag{18}$$

Pattern M_W defines an image (square matrix) of size $N_W \times N_W$ which is

$$M_W : D_W \to \{0,...,255\}, \quad \text{where:} \quad D_W = [1,...,N_W] \times [1,...,N_W] \subset N^2. \tag{19}$$

Generalisation of the Hough Transform

Let us consider first a simpler task than that given by equation (12). To make it simpler we remove the scaling parameter s. Thus, the Hough Transform $H(x_T, y_T, \alpha)$ for a grey-level image $B_G(x, y)$ (equation 17) in the process of identification of pattern M_W (equation 19) is given by (→ 12)

$$H(x_T, y_T, \alpha) = \sum_{(x_i, y_i) \in M_W} h(x_i, y_i, x_T, y_T, \alpha), \tag{20}$$

where

$$h(x_i, y_i, x_T, y_T, \alpha) = 255 - \left| B_G(x_i', y_i') - M_W(x_i, y_i) \right|, \tag{21}$$

provided that values x_i', y_i' are calculated with the following formulas:

$$\begin{cases} x_i' = x_r + (x_i - x_r)\cos(\alpha) - (y_i - y_r)\sin(\alpha) + x_T \\ y_i' = y_r + (x_i - x_r)\sin(\alpha) + (y_i - y_r)\cos(\alpha) + y_T \end{cases}. \tag{22}$$

As the above formulas show, the implementation of this definition of the Hough Transform does not differ from the standard definition. However, it enables us to apply the method directly to grey-level images. This process is illustrated in Figure 3.

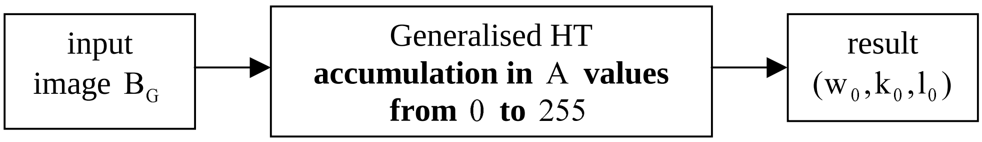

Figure 3. A modified way of conduct for grey-level images

The detection success rate for this method in a large collection of test images is over 80%. This method seems to be promising for any form of object identification.

Application of the histogram function

To improve this elaborated method we wish to find a characteristic of the pattern that is invariant under rotation. The histogram is the obvious characteristic especially for diverse images (of 256 grey levels). The histogram study is introduced into the previous scheme (Figure 3) as shown in Figure 4.

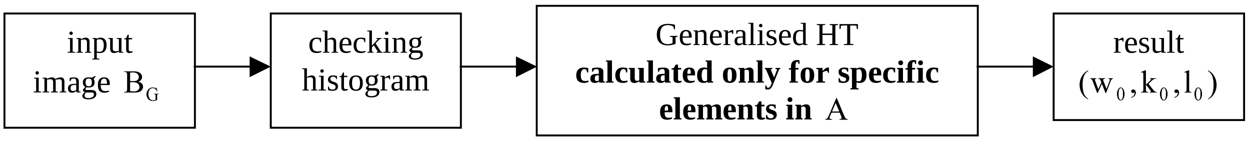

Figure 4. Pattern localisation process taking into consideration a histogram study

The histogram of pattern M_W is determined once only and compared with the histograms of fragments of image B_G, determined at all possible locations of the pattern M_W. A histogram of a grey-level image defines a function that maps for any grey level (from 0 to 255) the number of image pixels that have that level and may be denoted by

$$\Phi : \{0,...,255\} \rightarrow N \quad . \tag{23}$$

ALGORITHM – histogram analysis

Step 1: Determine histogram Φ_W of the identified pattern M_W.

Step 2: Determine histograms $\Phi_{Q(i,j)}$ for all fragments $Q_G^{B(i,j)}$ of size $N_W \times N_W$ of image B_G where $i = 1,..., W - N_W + 1$ and $j = 1,..., K - N_W + 1$.

Step 3: Compare the received histogram $\Phi_{Q(i,j)}$ with the histogram Φ_W using the following value

$$d_{(i,j)} = \frac{4}{\pi \cdot N_W{}^2} \sum_{k=0}^{255} \left| \Phi_{Q(i,j)}(k) - \Phi_W(k) \right| , \tag{24}$$

where the factor $\pi / 4$ results from the relation of a circle area inscribed into a square.

Step 4: If $d_{(i,j)}$ is higher than a threshold value $d_{treshold}$ then it is excluded when calculating the accumulator array A.

This simple method reduces the complexity in terms of the calculation performed for the whole process (more than 50%).

6. The Hough Transform and the scaling problem

There is another important issue that appears while identifying patterns in images and that is pattern scaling. This appears in the case where the scale of the initial image is unknown and occurs when analysing images used in the guidance system of cruise missiles for example. Unfortunately, taking into consideration pattern scaling adds an extra dimension to the parameter space. The Hough Transform $H(x_T, y_T, \alpha, s)$ (which takes into account translation, rotation and scaling) for image $B_G(x, y)$ $(\rightarrow 17)$ in the process of identifying pattern M_W determined by (19) may be defined as

$$H(x_T, y_T, \alpha, s) = \sum_{(x_i, y_i) \in M_W} h(x_i, y_i, x_T, y_T, \alpha, s) , \tag{25}$$

where

$$h(x_i, y_i, x_T, y_T, \alpha, s) = 255 - \left| B_G(x_i'', y_i'') - M_W(x_i, y_i) \right|, \tag{26}$$

and the values x_i'', y_i'' are calculated from

$$\begin{cases} x_i'' = x_r + s(x_i - x_r)\cos(\alpha) - s(y_i - y_r)\sin(\alpha) + x_T \\ y_i'' = y_r + s(x_i - x_r)\sin(\alpha) + s(y_i - y_r)\cos(\alpha) + y_T \end{cases}. \tag{27}$$

Adding a new parameter results in an increase in the calculation complexity and demand for extra memory for the accumulator (4D problem). However, because the scale range is commonly known and it is not too large, only a few values of scale factor s are often enough to realise the process of identification.

If we assume that the following n values of the scale s factor must be taken into consideration

$$\xi_1, \ldots, \xi_n, \qquad n \in N, \tag{28}$$

then the parameter space may be determined in the following way:

$$P = P_1 \times P_2 \times P_3 \times P_4 = [1, \ldots, W] \times [1, \ldots, K] \times [0, \ldots, L-1] \times [\xi_1, \ldots, \xi_n], \qquad (\Delta\alpha = \frac{2\pi}{L}). \tag{29}$$

In order to accelerate calculations (applying the histogram study) the set of patterns

$$\{M_W^1, \ldots, M_W^n\}, \tag{30}$$

must be generated first by scaling a given pattern M_W within a range determined by values $\xi_1, \ldots, \xi_n$. The process shown in Figure 4 for any pattern formed from the set (30) can then be applied. Such an approach can drastically reduce the number of calculations required.

However, this method has one disadvantage that results from having to calculate histograms for an initial image n times (the size of each pattern is different). A solution is to create a new set of patterns $\{\overline{M}_W^1, \ldots, \overline{M}_W^n\}$ of the same size but without losing information connected with the scale of patterns $M_W^1, \ldots, M_W^n$. Note that the size of pattern M_W^1 is $N_W^1 \times N_W^1$. Appropriate patterns $\overline{M}_W^1, \ldots, \overline{M}_W^n$ from patterns $M_W^1, \ldots, M_W^n$ can be obtained by separating their central part of size $N_W^1 \times N_W^1$. As a result, we have patterns that “remember” the scale they were created at, but are of the same size. A graphical illustration of this process is shown in Figure 5.

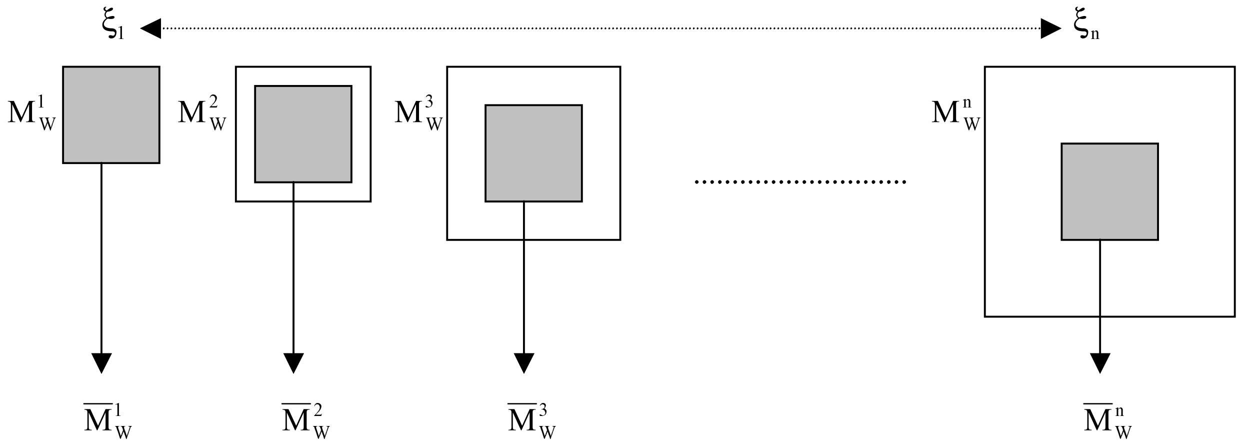

Figure 5. Graphical illustration of the process of patterns $\overline{M}_W^1,\ldots,\overline{M}_W^n$ creation

Since the received patterns are of the same size, it is sufficient to calculate their histograms once and compare them with the calculated histogram of the initial image. As the size of the patterns decreases, the time to create the histograms is reduced. Decreasing the patterns size also results in shortening the CPU time for the accumulator calculation. Unfortunately, patterns $\overline{M}_W^1,\ldots,\overline{M}_W^n$ carry less information than patterns $M_W^1,\ldots,M_W^n$.

The process of an object localisation for any image is shown in Figure 6.

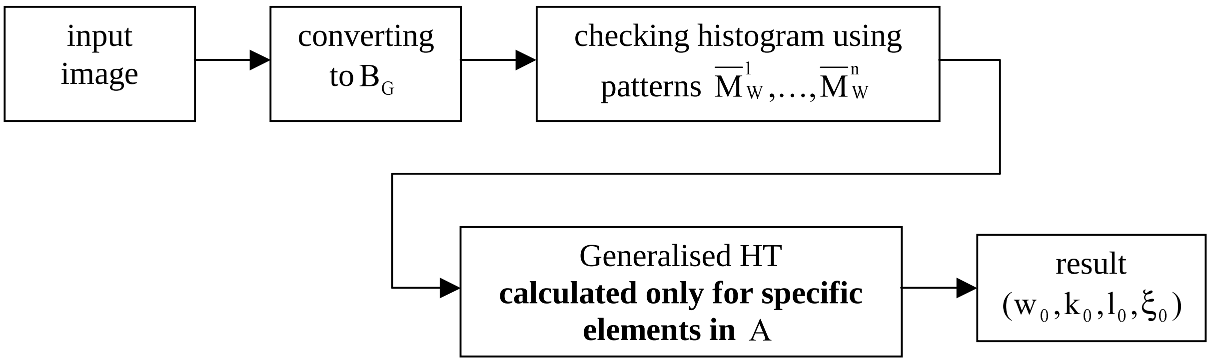

Figure 6. The way of conduct while identifying objects in images of unknown scale

Results based on this method are presented in Figure 7. Because of the difficulty of illustrating a four-dimensional accumulator it has been reduced to a 2D image once the maximal values for rotation angle and scale have been found.

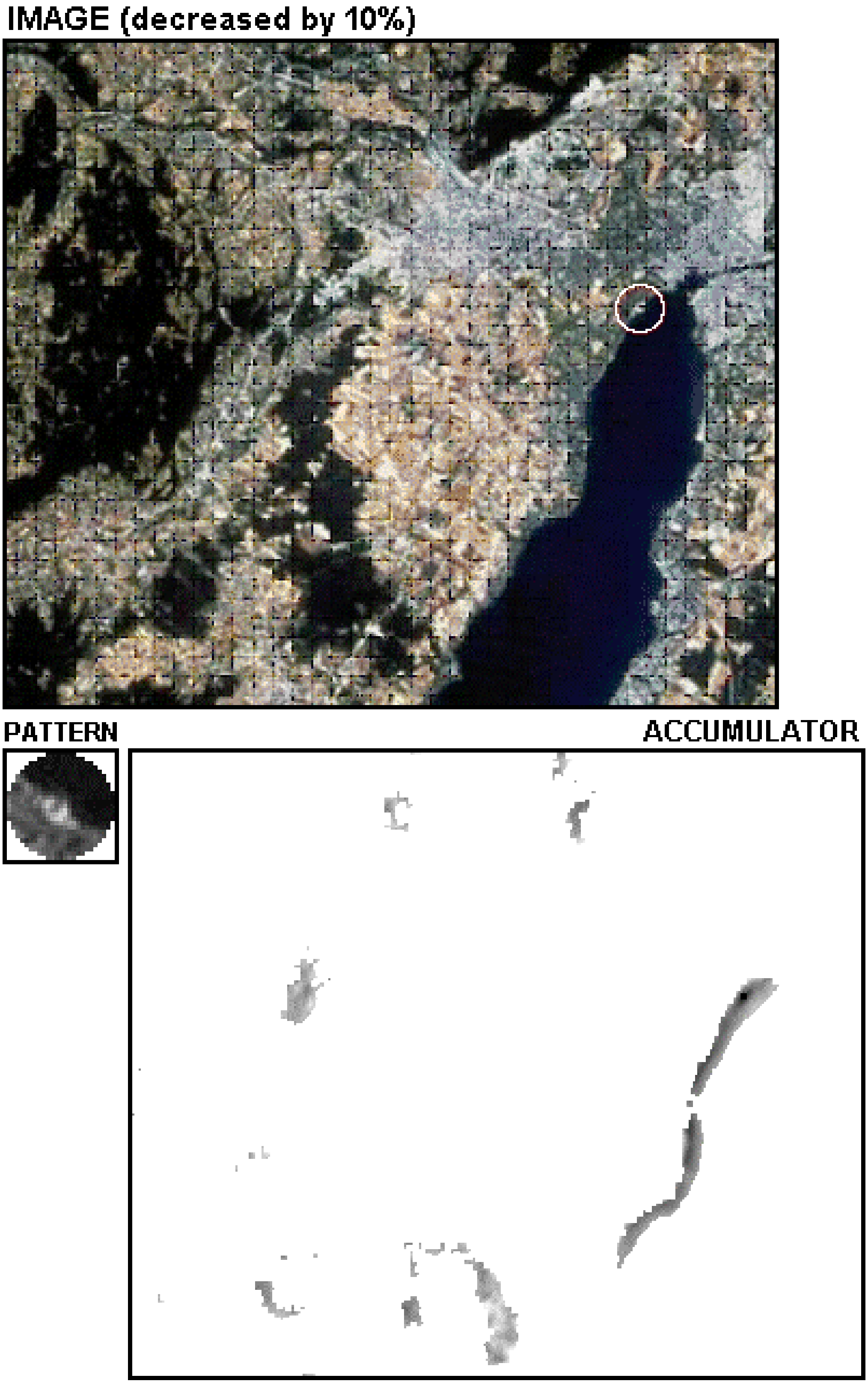

Figure 7. Object location in a satellite image (the pattern "fits" well along a long distance of coastline)

7. Generalisation of the Hough Transform for colour images

The first approach

The question arises - what to do in case of coloured images? Because any pixel of a coloured image has three components (R, G, B) there is a problem with adopting the Hough Transform to identify objects in coloured images. The simplest method is to convert the initial image (TrueColor, HighColor or binary) to a grey-level image B_G. Such a solution has been suggested in Figure 6.

Within most current computer technology we consider TrueColor images (8 bits per each RGB component – i.e., a pallet of 16,777,216 possible colours for each image pixel). Conversion of such an image to grey levels may be done through the projection of points in an RGB cube onto its diagonal (combining whiteness and blackness). There are no crucial calculation problems with converting any image (as regards colouring) to the form of 256 grey levels. There is nothing new in this approach to the problem, nevertheless in many applications it will be an acceptable solution because of its simplicity.

The second approach

The second approach to the problem of a colour pattern location in a colour image is based on calculating the distance between $M(x_i, y_i)$ and $B(x_i', y_i')$ with the following formula ($\rightarrow$ 21):

$$\left|B(x_i', y_i') - M_W(x_i, y_i)\right| = \sqrt{(r_B - r_M)^2 + (g_B - g_M)^2 + (b_B - b_M)^2}\,, \qquad (31)$$

i.e. Euclidean distance between two points (r_B, g_B, b_B) and (r_M, g_M, b_M) in the RGB cube. Unfortunately, this is often not acceptable due to its high computational complexity.

The third, final approach

The third approach assumed at the beginning that an input image and a pattern are given with 256-colour depth. Then it is easy to calculate the distance between $M_W(x_i, y_i)$ and $B(x_i', y_i')$ by pre-calculating a square matrix M_d, that includes all possible distances between any two of 256 base colours. The size of the matrix is 65,536 integer cells, which is acceptable in terms of memory requirement. Such a matrix solves the problem of calculating a distance as well as accelerates the calculations considerably. It allow us to obtain a distance in the following way (→ 21):

$$\left|B(x_i', y_i') - M_W(x_i, y_i)\right| = M_d[B(x_i', y_i'), M_W(x_i, y_i)] . \tag{32}$$

Nevertheless another problem appears, i.e. the problem of creating matrix M_d. There are many colour models for example: RGB cube, CMY or single-hexcone HSV. Thus it is necessary to choice the colour model first and then establish the colour quantisation method, and then finally calculate all possible distances to be stored in matrix M_d.

The process of object location in a colour image is shown in Figure 8.

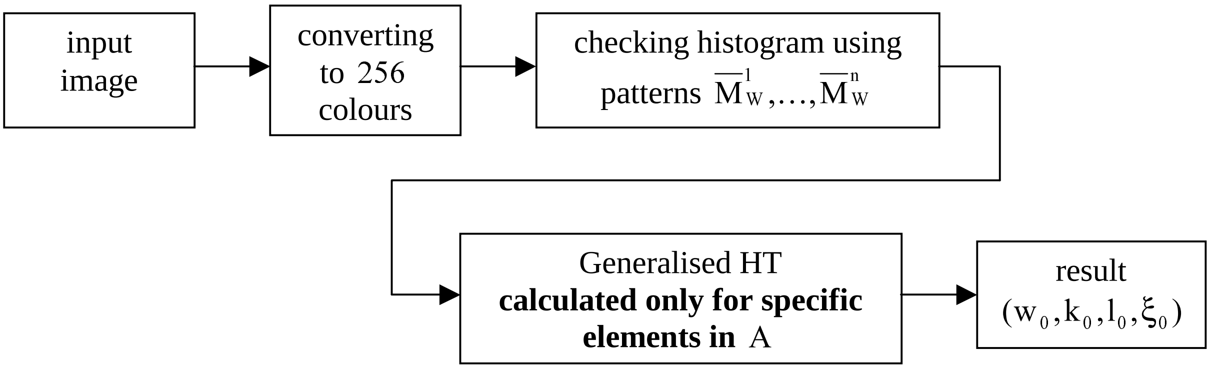

Figure 8. Identifying objects in colour images of unknown scale

Examining result based on this method (with an RGB colour model) is presented in Figure 9.

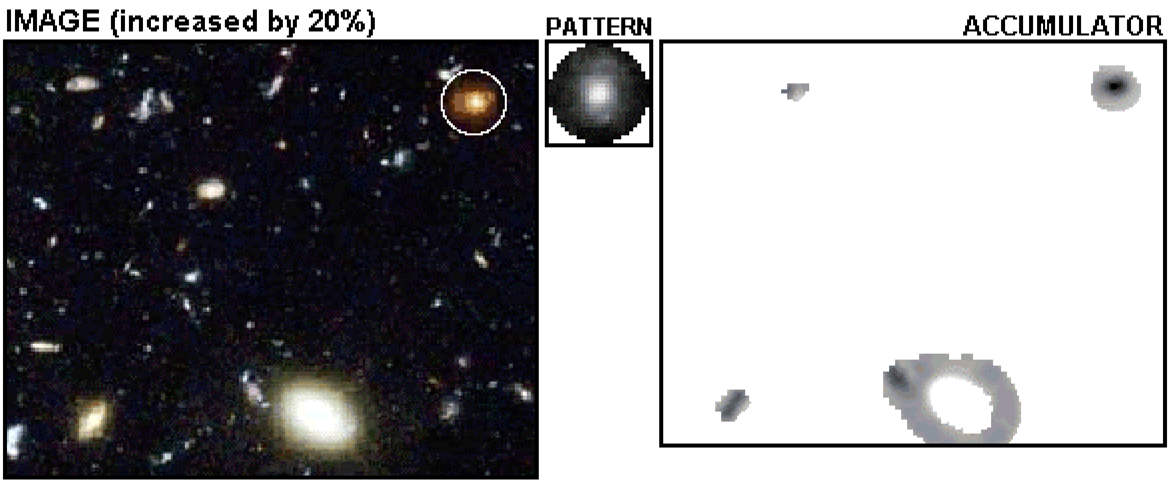

Figure 9. Object location in an image obtained from HST

8. Conclusion

There are a number of military applications that could benefit from the method presented, especially in the area of automatic vision systems. These include automatic surveillance of enemy territory using data remotely collected by satellite or images taken by aircraft. Improvements are required however, which could be based on using ideas from the Fast Hough Transform or the Probabilistic Hough Transform. Further work in these areas is expected to provide better results.

REFERENCES

[1] Anagnou A., Blackledge J. M.: *Research Report - Pattern Recognition using the Hough Transform*. Sciences and Engineering Research Centre, De Montfort University, Leicester 1993.

[2] Atiquzzaman M.: *Multiresolution Hough transform - An efficient method of detecting patterns in images*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 11, 1992, 1090-1095.

[3] Ballard D. H., Brown C. M.: *Computer Vision.* Prentice-Hall, Englewood Cliffs, New York 1982.

[4] Ballard D. H.: *Generalizing the Hough Transform to Detect Arbitrary Shapes.* Readings in Computer Vision: Issues, Problems, Principles, and Paradigms. Los Altos, CA. 1987, pp. 714-725.

[5] Blackledge J. M.: *Spatial data representation for rotation invariant correlation*. Sciences and Engineering Research Centre, De Montfort University, Leicester 1996.

[6] Davies E. R.: *Finding ellipses using the generalised Hough transform*. Pattern Recognition Letters, vol. 9, no. 2, 1989, 87-96.

[7] Davies E. R.: *Minimising the search space for polygon detection using the generalised Hough transform*. Pattern Recognition Letters, vol. 9, no. 3, 1989, 181-192.

[8] Davies E. R.: *Reduced parameter spaces for polygon detection using the generalized Hough transform*. Proceedings, Eighth International Conference on Pattern Recognition (Paris, France, October 27-31, 1986), IEEE Publ. 86CH2342-4, 495-497.

[9] Deans S. R.: *Hough transform from the Radon transform*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 3, no. 2, 1981, 185-188.

[10] Duda R. O., Hart P. E.: *Use of the Hough Transformation to Detect Lines and Curves in Pictures*. Comm.ACM., vol. 15, 1972, 11-15.

[11] Fu K. S., Gonzalez R. C., Lee C. S. G.: *ROBOTICS: Control, Sensing, Vision, and Intelligence.* McGraw-Hill, New York 1987.

[12] Han J. H., Koczy L. T., Poston T.: *Fuzzy Hough transform*. Pattern Recognition Letters, vol. 15, no. 7, 1994, 649-658.

[13] Hough P. V. C.: *Method and means for recognizing complex patterns*. U.S. Patent 3,069,654, Dec. 18, 1962.

[14] Illingworth J., Kittler J., *A survey of the Hough Transform*. Computer Vision, Graphics and Image Processing **44**, 1988, pp. 87-116.

[15] Jain A. K.: *Fundamentals of Digital Image Processing.* Prentice-Hall, New Jersey 1989.

[16] Kierkegaard P.: *A method for detection of circular arcs based on the Hough transform*. Machine Vision and Applications, vol. 5, no. 4, 1992, 249-263.

[17] Kiryati N., Eldar Y., Bruckstein A. M.: *A probabilistic Hough transform*. Pattern Recognition, vol. 24, no. 4, 1991, 303-316.

[18] Kwiatkowski W.: *Zastosowanie techniki Hougha przetwarzania obrazów do pomiaru zmiany położenia obiektu na obrazie rastrowym.* Biuletyn WAT, **4**, 1994, pp. 33-46.

[19] Leavers V. F.: *Shape Detection in Computer Vision Using the Hough Transform*. Springer, London 1992.

[20] Leavers V. F.: *The Dynamic Generalized Hough Transform: Its Relationship to the Probabilistic Hough Transforms and an Application to the Concurrent Detection of Circles and Ellipses*. CVGIP - Image Understanding, vol. 56, no. 3, 1992, 381-398.

[21] Li H. F., Pao D., Jayakumar R.: *Improvements and systolic implementation of the Hough tranformation for straight line detection*. Pattern Recognition, vol. 22, no. 6, 1989, 697-706.

[22] Li H., Lavin M. A., LeMaster R. J.: *Fast Hough transform*. Proceedings of the Third Workshop on Computer Vision: Representation and Control (Bellaire, MI, October 13-16, 1985), IEEE Publ. 85CH2248-3, 75-83.

[23] Li H., Lavin M. A., LeMaster R. J.: *Fast Hough transform: a hierarchical approach*. Computer Vision, Graphics, and Image Processing, vol. 36, 1986, 139-161.

[24] Lot R. C., Tsai W. H.: *Grey-scale Hough transform for thick line detection in grey-scale images*. Pattern Recognition, vol. 28, no. 5, 1995, 647-661.

[25] McLauchlan P. F., Mayhew J. E., Frisby J. P.: *Stereoscopic recovery and description of smooth textured surfaces*. Image and Vision Computing, vol. 9, no. 1, 1991, 20-26.

[26] McLaughlin R. A., Alder M. D.: *Technical Report - The Hough Transform versus the UpWrite.* Tech. Rep. 97/2, The University of Western Australia, Centre for Intelligent Information Processing Systems, Dept. of E.E. Eng., U.W.A., Stirling Hwy, Nedlands W.A. 6907, Australia, Available from http://ciips.ee.uwa.edu.au/Papers/Technical_Reports/, 1997.

[27] McLaughlin R. A.: *Technical Report - Randomized Hough Transform: Improved ellipse detection with comparison.* Tech. Rep. 97/1, The University of Western Australia, Centre for Intelligent Information Processing Systems, Dept. of E.E. Eng., U.W.A., Stirling Hwy, Nedlands W.A. 6907, Australia, Available from http://ciips.ee.uwa.edu.au/Reports/, 1997.

[28] Pao D., Li H. F., Jayakumar R.: *A decomposable parameter space for the detection of ellipses*. Pattern Recognition Letters, vol. 14, no. 12, 1993, 951-958.

[29] Pao D., Li H. F., Jayakumar R.: *Detecting parametric curves using the straight line Hough transform*. Tenth International Conference on Pattern Recognition (Atlantic City, NJ, June 16-21, 1990), IEEE Catalog No. 90CH2898-5, 1990, subconference B, 620-625.

[30] Pao D., Li H. F., Jayakumar R.: *Shapes recognition using the straight line Hough transform: Theory and generalization*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 14, no. 11, 1992, 1076-1089. *Pnvestt Pvestoc(r)74(r)74(n)10v.,6(2((. 121 /F3(put)10v)(d13 1 Tf 5.8072R)9130.0 TD 0.0008 Tc 40.0002 Tw 1(Uber)6*

[31] Pei S. C., Horng J. H.: *Circular arc detection based on Hough transform*. Pattern Recognition Letters, vol. 16, no. 6, 1995, 615-625.

[32] Radon J.: *Uber die BestimcJ0361 0.022.2(Funk.8(sen du die Bch ih die BtterJ /Integ die Balw)4.9(tterJ die Br)7.2(d 6(tgtimcJ0*