Correspondence Rules for Motion Detection using Randomized Methods

Amr Goneid and Howaida Naguib
Department of Computer Science & Engineering, the American University in Cairo, Cairo, Egypt
goneid@aucegypt.edu

Abstract

Parametric domain techniques have proved to be quite successful in the detection of 2-D and 3-D objects and motion. Most noted in these methods is the Motion Detection Randomized Hough Transform (MDRHT). This technique proved to be able to decreases considerably the time consumption and memory requirements of the Hough Transform through the use of a random sampling mechanism in the image space. In such method, the most important problem is the establishment of correspondence between sets of points belonging to the same object in successive motion frames. In previous researches, 2-points correspondence rules are commonly used. In the present paper, we introduce a mathematical analysis to investigate the invariance of such rules under translation and rotation. The present paper also introduces two new 3-points correspondence rules and proves one of them to be invariant under translational and rotational motion. The performance of the different correspondence rules has been investigated through experiments using a randomized algorithm. From these experiments, the newly introduced 3-poin rules proved to outperform the other rules by 3 times and 8 times for translational and rotational motion detection, respectively.

Keywords: *Motion Detection, Randomized Hough Transform, Correspondence Rules*

1. Introduction

The detection and recognition of a moving object in a sequence of time varying images proves to be a very important task in machine intelligence in general and computer vision in particular. Parametric domain techniques can be successfully used with a number of variants. In such methods, the image is transformed into some parameter space and the motion detection process is applied in that space. An efficient parametric domain method is the Randomized Hough Transform (RHT) [1,2] that uses a random sampling mechanism in the image space, scores accumulation in the parameter space, and then bridges between them using a converging mapping. The use of such method for motion detection is known as the Motion Detection Randomized Hough Transform (MDRHT) [3, 4, 5]. Since random sampling is used, the process of establishing correspondence between sets of points belonging to the same object in successive motion frames proves to be the most important problem in this methodology. Improving the accuracy of correspondence rules will improve the performance of the algorithm.

In the present work, motion detection was considered through the analysis of a sequence of time varying gray level image stream using the RHT algorithm that provides an efficient simple non-model based methodology using edge pixels as features. The objective of our work is to construct a set of correspondence rules that would maximize the ability of the methodology to detect motion parameters for both pure translational and pure rotational motions in the case of 2-D rigid objects. Analysis of accuracy and efficiency of correspondence is restricted to the cases of two points and three point pairs to select rules maximizing the performance.

For that purpose, five different correspondence rules are investigated. The first three are 2-point rules that were used in previous researches. They measure correspondence through 2-point x- and y- differences, City Block distances and Euclidean distances, respectively. The present work introduces the two remaining rules for the first time. These are 3-point rules that measure correspondences through 3-point City Block distances and triangular areas, respectively.

We develop a mathematical analysis of the invariance of the five rules given for both pure translational and pure rotational motions. Also, a performance parameter is introduced to measure the capability of peak detection in the RHT space and to compare the performances of the randomized motion detection methodology for the different rules.

For translational and rotational motions, different simulation experiments are conducted in order to investigate the dependence of peak detection efficiency on the correspondence rule and on the size of the random sample. Investigations are also made of robustness of the algorithms under noise conditions, varying angles of rotation, RHT spatial resolution, and correspondence tolerance.

2. The Basic MDRHT Algorithm

The Randomized Hough Transform (RHT) was widely used for object detection [6 - 9], visual control and medical imaging [10, 11], and motion detection [3 - 5, 12 - 15]. The Main idea in using RHT for motion detection is that the parameter space is divided according to the displacement of different points in the image and accumulated in the accumulator space to get the real displacement of the object being tracked.

Motion Detection Randomized Hough Transform (MDRHT) is a feature based method that detects motion of a non-model based 2-D rigid object in a sequence of time varying images using the edge pixels as features. This usage of edge pixels helps to avoid overlapping and to avoid difficulties faced by some other motion detection methods. Many methods that use the feature based approach has been developed using parallel projection, the perspective transformation, and both 3-D points and 3-D lines as features. Usually, in these methods the correspondence between selected features is assumed to be known.

The method uses Hough transform (HT) and its' variants, which is considered to be flexible since it can work even in noisy and complex images, by randomly picking points pair from two images and calculating the transition with them. The use of HT in motion detection is not a new idea, but it is time and memory consumptive and needs a relatively short time

span between images. It is suitable for finding global features like line and curve segments from a static image. An improved method based on the Randomized Hough Transform was later developed for motion detection and called MDRHT [3].

The basic MDRHT algorithm is illustrated as follows:

Algorithm 1: Basic Motion Detection using Randomized Hough Transform
Assume that there are two gray level image frames:

1. Two binary edge pictures are formed, B and C
2. Assuming a 2-point correspondence rule, the correspondence of the edge points in the two sets is examined in the following way:
    - Two points (b_1, b_2) are picked randomly from frame B.
    - A corresponding point pair (c_1, c_2) is sought from frame C.
      The *corresponding pair* is any point pair which satisfies a certain correspondence rule. Also as an extra or alternative a heuristic condition can be included in this correspondence problem.
3. The transition (d_x, d_y) is then calculated between the points of the first and second images.
4. The cell A(d_x, d_y) is accumulated in the accumulator space.
5. Repeat steps 1 - 4 a suitable number of repetitions.

3. Correspondence Rules

The core of MDRHT is the use of random sampling in the image space, score accumulation in the parameter space, and converging mapping as a bridge between the two spaces. Around this core, some issues could be implemented in different ways which give different variants of the algorithm.

It is argued that [8, 9] research effort should focus on addressing accuracy instead of computational or memory aspects, since they are Hardware problems that could be solved if faster and cheaper hardware invented. This implies that we consider the correspondence problem as the most important problem in this algorithm. Improving the accuracy of correspondence rules will improve the performance of the algorithm and will help improving its' throughput.

3.1 Two-point correspondence rules

Point pairs can be used to detect the correspondence. The rules used to detect this correspondence as shown in Table (1) work as follows:

Rule 1 (Displacement): uses point pairs, (b_1,b_2) from first image, and (c_1,c_2) from second image. The C-pair will be considered to be correspondent to the B-pair if the *x* displacement between (b_1,b_2) is equal to the *x* displacement between (c_1,c_2) and the *y* displacement between (b_1,b_2) is equal to the *y* displacement between (c_1,c_2).

Rule 2 (City Block Distance): uses point pairs, (b_1,b_2) from first image, and (c_1,c_2) from second image. The C-pair will be considered to be correspondent to the B-pair if the city block distance between (b_1,b_2) is equal to the city block distance between (c_1,c_2).

Rule 3 (Euclidian Distance): uses point pairs, (b_1,b_2) from first image, and (c_1,c_2) from second image. The C-pair will be considered to be correspondent to the B-pair if the Euclidian distance between (b_1,b_2) equals the Euclidian distance between (c_1,c_2).

Table (1): Two points correspondence rules

No.	No. of points	Rule
1	2	$(dx < \varepsilon) \cap (dy < \varepsilon)$
2	2	Equal city block distance: $equal(y_{b12} + x_{b12}, y_{c12} + x_{c12})$
3	2	Equal Euclidian distance: $equal(\sqrt{y_{b12}^2 + x_{b12}^2}, \sqrt{y_{c12}^2 + x_{c12}^2})$

In the above table,

$(x_{b1}, y_{b1}), (x_{b2}, y_{b2})$ are the coordinates of the selected B-pair,

$(x_{c1}, y_{c1}), (x_{c2}, y_{c2})$ are the coordinates of the corresponding C-pair.

Also,

$dx = |(x_{b2} - x_{b1}) - (x_{c2} - x_{c1})|$, $dy = |(y_{b2} - y_{b1}) - (y_{c2} - y_{c1})|$, ε is a tolerance,

and

$y_{b12} = |y_{b2} - y_{b1}|, x_{b12} = |x_{b2} - x_{b1}|, y_{c12} = |y_{c2} - y_{c1}|, x_{c12} = |x_{c2} - x_{c1}|$

3.2 New three-point correspondence rules

In the present work, we introduce two new correspondence rules using point triples as given in Table (2). These rules as specified as follows:

Rule 4 (Triple Point City Block Distance): uses point triples, (b_1,b_2,b_3) from first image, and (c_1,c_2,c_3) from second image. The C-triple will be considered to be correspondent to the B-triple if the sum of city block distances between (b_1,b_2,b_3) equals the sum of the city block distances between (c_1,c_2,c_3).

Rule 5 (Triangle Area): uses point triples, (b_1,b_2,b_3) from first image, and (c_1,c_2,c_3) from second image. The C-triple will be considered to be correspondent to the B-triple if the area of the triangle formed by (b_1,b_2,b_3) equals to the area of the triangle formed by (c_1,c_2,c_3).

Table (2) Proposed new three-point correspondence rules

No.	No. of points	Rule
1	3	Equal city block distance: $equal(x_{b123} + y_{b123}, x_{c123} + y_{c123})$
2	3	Equal triangle areas: $equal(\lvert \det(A_B) \rvert, \lvert \det(A_C) \rvert)$

In the above table,

$$x_{b123} = |x_{b1} - x_{b2}| + |x_{b1} - x_{b3}| + |x_{b2} - x_{b3}|,$$
$$y_{b123} = |y_{b1} - y_{b2}| + |y_{b1} - y_{b3}| + |y_{b2} - y_{b3}|,$$
$$x_{c123} = |x_{c1} - x_{c2}| + |x_{c1} - x_{c3}| + |x_{c2} - x_{c3}|,$$
$$y_{c123} = |y_{c1} - y_{c2}| + |y_{c1} - y_{c3}| + |y_{c2} - y_{c3}|$$

Also,

$$A_B = \begin{pmatrix} x_{B1} & x_{B2} & x_{B3} \\ y_{B1} & y_{B2} & y_{B3} \\ 1 & 1 & 1 \end{pmatrix} \quad , \quad A_C = \begin{pmatrix} x_{C1} & x_{C2} & x_{C3} \\ y_{C1} & y_{C2} & y_{C3} \\ 1 & 1 & 1 \end{pmatrix}$$

4. Invariance of correspondence rules

In the following, we investigate the invariance of the different correspondence rules under translation and rotation.

4.1 Invariance under translation

When detecting pure translational motion, the rule used to establish correspondence between two shapes must be invariant under translation. This ensures that the rule can establish correspondence whether the shape is stationary or undergoing translational motion. In the following, we investigate such invariance for the five rules given above. To do so, we consider a stationary shape in a frame (B), another stationary shape in frame (C) and then shape (C) translated by a displacement (δ) in frame (D). Let Rule (0) be the correspondence rule applied between (B) and (C), and Rule (δ) be the same rule applied between (B) and (D). If Rule (δ) does not change from Rule (0), then the rule is invariant under translation.

To investigate such invariance, let us consider the points (i) and (j) for the 2-point algorithms, and (i), (j) and (k) for the 3-point algorithms. Let (f) be a frame (b, or c or d) and:

$(dx)_f = (x_j - x_i)_f$ and $(dy)_f = (y_j - y_i)_f$

Also, $(x_{ij})_f = |x_i - x_j|_f$ and $(y_{ij})_f = |y_i - y_j|_f$

Correspondence rule (1) (2-point equal differences in both x and y):

Rule (0) $\equiv \{[(dx)_{bc} < \varepsilon] \cap [(dy)_{bc} < \varepsilon]\}_0$

where $(dx)_{bc} = (dx)_b - (dx)_c$ *and* $(dy)_{bc} = (dy)_b - (dy)_c$

The translation from (c) to (d) gives :

$(dx)_d = (x_j + x - x_i - x) = (dx)_c$, and similarly $(dy)_d = (y_j + y - y_i - y) = (dy)_c$

It follows that $(dx)_{bd} = (dx)_{bc}$ *and* $(dy)_{bd} = (dy)_{bc}$

Hence, *Rule* ($\delta u \equiv \{[(dx)_{bd} < \varepsilon] \cap [(dy)_{bd} < \varepsilon]\} = \{[(dx)_{bc} < \varepsilon] \cap [(dy)_{bc} < \varepsilon]\}_0 \equiv Rule(0)$

This proves that the correspondence rule (1) is invariant under pure translation.

Correspondence rule (2) (2-point equal city block distances):

$Rule\,(0) \equiv \{[((x_{ij})_b + (y_{ij})_b) - ((x_{ij})_c + (y_{ij})_c)] < \varepsilon\}_0$

$But\ (x_{ij})d = (x_{ij})c\ \ and\ \ (y_{ij})d = (y_{ij})c$

Hence,

$Rule\,(\delta) \equiv \{[((x_{ij})_b + (y_{ij})_b) - ((x_{ij})_d + (y_{ij})_d)] < \varepsilon\} =$

$= \{[((x_{ij})_b + (y_{ij})_b) - ((x_{ij})_c + (y_{ij})_c)] < \varepsilon\}_0 \equiv Rule(0)$

This proves that the correspondence rule (2) is invariant under pure translation.

Correspondence rule (3) (2-point equal Euclidean distances):

$Rule\,(0) \equiv \{[((x_{ij})_b^2 + (y_{ij})_b^2) - ((x_{ij})_c^2 + (y_{ij})_c^2)] < \varepsilon\}_0$

$But\ (x_{ij})_d = (x_{ij})_c\ \ and\ \ (y_{ij})_d = (y_{ij})_c$

Hence,

$Rule\,(\delta) \equiv \{[((x_{ij})_b^2 + (y_{ij})_b^2) - ((x_{ij})_d^2 + (y_{ij})_d^2)] < \varepsilon\} = Rule(0)$

This proves that the correspondence rule (3) is invariant under pure translation.

Correspondence rule (4) (3-point equal city block distances):

Let $(x_{ijk})_f = (x_{ij})_f + (x_{ik})_f + (x_{jk})_f\ \ and\ \ (y_{ijk})_f = (y_{ij})_f + (y_{ik})_f + (y_{jk})_f$

$Rule\,(0) \equiv \{[((x_{ijk})_b + (y_{ijk})_b) - ((x_{ijk})_c + (y_{ijk})_c)] < \varepsilon\}_0$

But $(x_{ijk})_d = (x_{ijk})_c$ and $(y_{ijk})_d = (y_{ijk})_c$

Hence, $Rule\,(\delta) \equiv Rule\,(0)$

This proves that the correspondence rule (4) is invariant under pure translation.

Correspondence rule (5) (3-point equal triangle areas):

$Rule(0) \equiv \{[\det(A_b) - \det(A_c)] < \varepsilon\}_0$

But $\det(A_d) = \det(A_c)$

Hence, $Rule(\delta) = Rule(0)$

This proves that the correspondence rule (5) is invariant under pure translation.

4.2 Invariance under rotation

When detecting pure rotational motion, the rule used to establish correspondence between two shapes must be invariant under such rotation. This ensures that the rule can establish correspondence whether the shape is stationary or undergoing rotational motion. In the following, we investigate such invariance for the five rules given before. To do so, we consider a stationary shape in a frame (b), another stationary shape in frame (c) and then

shape (c) rotated by an angle (φ) in frame (d). Let Rule (0) be the correspondence rule applied between (b) and (c), and Rule (φ) be the same rule applied between (b) and (d). If Rule (φ) does not depend on (φ), (i.e. does not change from Rule (0)), then the rule is invariant under pure rotation. Otherwise, the rule in not invariant under rotation and cannot be used to establish correspondence in case of rotational motion.

To investigate such invariance, let us consider the points (i) and (j) for the 2-point rules, and (i), (j) and (k) for the 3-point rules. Let (f) be a frame (b, or c or d) and:

$(dx)_f = (x_j - x_i)_f$ and $(dy)_f = (y_j - y_i)_f$

Also, $(x_{ij})_f = |x_i - x_j|_f$ and $(y_{ij})_f = |y_i - y_j|_f$

For simplicity, consider the shape to be a unit circle and the rotation point (x_r , y_r) to be the center of the circle. In this case, the (x,y) coordinates of a point on the shape are simply $(\cos\theta, \sin\theta)$, where θ is the angle between the radial vector to the point and the x-axis. It follows that:

$(dx)_f = (x_j - x_i)_f = (\cos\theta_j - \cos\theta_i)_f$ and

$(dy)_f = (y_j - y_i)_f = (\sin\theta_j - \sin\theta_i)_f$

When the shape is rotated by an angle (φ), then $\theta \rightarrow \theta + \varphi$. We now investigate the invariance of each correspondence rule under such transformation.

Correspondence rule (1) (2-point equal differences in both x and y):

$Rule(0) \equiv \{[(dx)_{bc} < \varepsilon] \cap [(dy)_{bc} < \varepsilon]\}_0$

where $(dx)_{bc} = (dx)_b - (dx)_c$ and $(dy)_{bc} = (dy)_b - (dy)_c$

Under the transformation $\theta \rightarrow \theta + \varphi$, the rotation from (c) to (d) gives :

$(dx)_d = (dx)_c \cos\varphi - (dy)_c \sin\varphi$, and

$(dy)_d = (dx)_c \sin\varphi + (dy)_c \cos\varphi$

It follows that

$(dx)_{bd} = (dx)_b - (dx)_d \neq (dx)_{bc}$ and

$(dy)_{bd} = (dy)_b - (dx)_d \neq (dy)_{bc}$

We can see that both $(dx)_{bd}$ and $(dy)_{bd}$ are explicit functions of φ.

Hence,

$Rule(\varphi) \equiv \{[(dx)_{bd} < \varepsilon] \cap [(dy)_{bd} < \varepsilon]\} \neq Rule(0)$

This proves that correspondence rule (1) is not invariant under pure rotation.

Correspondence rule (2) (2-point equal city block distances):

$Rule(0) \equiv \{[((x_{ij})_b + (y_{ij})_b) - ((x_{ij})_c + (y_{ij})_c)] < \varepsilon\}_0$

But $(x_{ij})_d = |(dx)_c \cos\varphi - (dy)_c \sin\varphi|$ and

$(y_{ij})_d = |(dx)_c \sin\varphi + (dy)_c \cos\varphi|$

Hence $(x_{ij})_d + (y_{ij})_d \neq (x_{ij})_c + (y_{ij})_c$ and

$Rule(\varphi) \neq Rule(0)$

This proves that the correspondence rule (2) is not invariant under pure rotation.

Correspondence rule (3) (2-point equal Euclidean distances):

$Rule\ (0) \equiv \{[((x_{ij})_b^2+(y_{ij})_b^2)-((x_{ij})_c^2+(y_{ij})_c^2)] < \varepsilon\}_0$

Under rotation, we find that $(x_{ij})_d^2 + (y_{ij})_d^2 = (x_{ij})_c^2 + (y_{ij})_c^2$

Hence, $Rule\ (\varphi) \equiv Rule(0)$

This proves that correspondence rule (3) is invariant under pure rotation.

Correspondence rule (4) (3-point equal city block distances):

Let $(x_{ijk})_f = (x_{ij})_f+(x_{ik})_f+(x_{jk})_f$ and

$(y_{ijk})_f = (y_{ij})_f+(y_{ik})_f+(y_{jk})_f$

$Rule\ (0) \equiv \{[((x_{ijk})_b + (y_{ijk})_b) - ((x_{ijk})_c + (y_{ijk})_c)] < \varepsilon\}_0$

An analysis similar to that for rule (2) shows that:

$(x_{ijk})_d+(y_{ijk})_d \neq (x_{ijk})_c + (y_{ijk})_c$ with an explicit dependence on φ.

Hence, $Rule\ (\varphi) \neq Rule\ (0)$

This proves that the rule for algorithm (4) is not invariant under pure rotation.

Correspondence rule (5) (3-point equal triangle areas):

The triangle formed by the three points (i,j,k) has sides of lengths S_{ij} , S_{ik} , S_{jk}.

Let $S = (1/2)(S_{ij} + S_{ik} + S_{jk})$ so that the area is given by:

$$A = [S (S - S_{ij})(S - S_{ik})(S - S_{jk})]^{1/2}$$

The S lengths are all 2-point Euclidean distances that are invariant under rotation as seen in rule (3). Hence, the area is invariant as well. It follows that:

$$Rule\ (\varphi) \equiv Rule\ (0)$$

This proves that rule (5) is invariant under pure rotation.

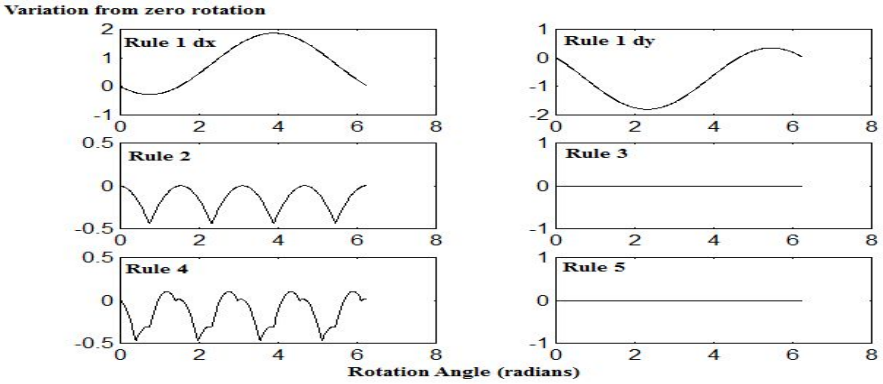


Figure (1): Variations for the different rules

Figure (1) shows an example of the variations computed for the different rules as a result of rotations. The figure shows clearly that only Rule (3) and Rule (5) are invariant under pure rotation.

5. Experiments on MDRHT

We have conducted a number of simulation experiments for motion detection using the MDRHT methodology in order to test the various correspondence rules and to measure their performance for the detection of translational and rotational motions. For this purpose, we have designed an algorithm that allows the variation of the correspondence rule, the multiplicity of point pairs, and the total number of random trials. Using this algorithm, we have investigated appropriate values for the tolerances used in applying the various correspondence rules as well as values to adopt for the accumulator resolution.

5.1 Experimental setup

Two-dimensional accumulators were used for the detection of translational motion in the 2-D plane. Figure (2) shows a typical probable instance in the establishment of correspondence using a point triplet for translational motion. In Figure (3), a typical accumulator display is given for the detection of translation using rule (5).

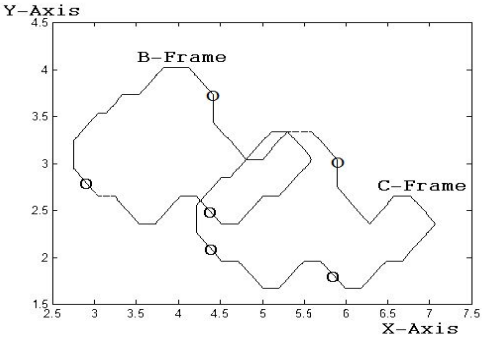


Figure (2): Typical translation instance.

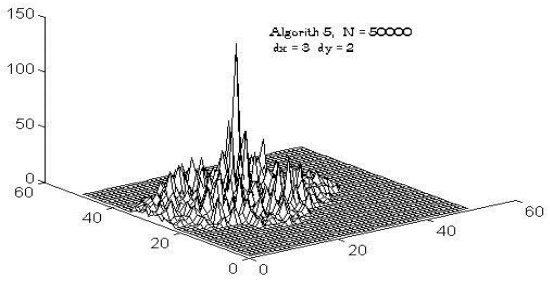


Figure (3): Typical accumulator display

For rotational motion detection, we confine the rotation parameters to the rotation angle (φ) so that a 1-D accumulator is sufficient. We also confine the correspondence establishment to rules (3) and (5) that proved to be invariant under pure rotation. Figure (4) shows a typical probable instance in the establishment of correspondence using a point triplet for rotational motion. In Figure (5), a typical accumulator display is given for the detection of rotation using rule (3).

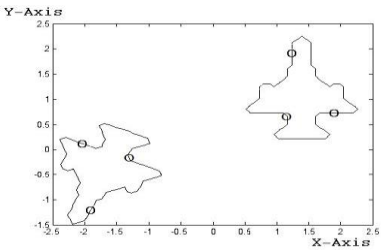


Figure (4): Typical rotational instance.

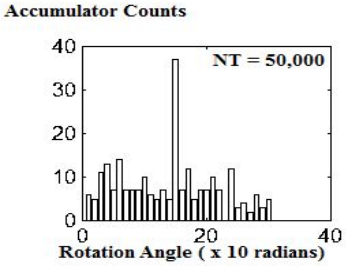


Figure (5): Accumulator display ($\varphi = \pi / 2$)

5.2 Measure of peak detection efficiency

In order to compare the motion detection capabilities between the different correspondence rules, we need to have a measure of the accumulator peak detection efficiency. The accumulator displays shown in Figures (3) and (5) indicate that the peaks detecting the motion can be conveniently modelled by a Laplacian probability density function (PDF). For the rotational motion, it is a 1-D PDF of the form:

$$p(\varphi) = \frac{1}{2b}\exp\left(-\frac{|\varphi - \varphi_0|}{b}\right)$$

where φ_0 is peak location and b is the scale parameter. Similarly, for translational motion, we may use a 2-D distribution in the x-y plane of the form:

$$p(x,y) = \frac{1}{4b^2}\exp\{-\frac{|x - x_0| + |y - y_0|}{b}\}$$

where (x_0,y_0) is peak location. Obviously, for such distributions the maximum probability is at the peak position. Therefore, it is convenient to measure the accumulator peak detection efficiency by a parameter $\rho = p_{\max}$. Therefore, we may use:

$$\rho_1 = \frac{\max(Acc_{xy})}{\Delta x\, \Delta y\, NT}, \quad \rho_2 = \frac{\max(Acc_{\varphi})}{\Delta\varphi\, NT}$$

where $\Delta\varphi$ is the resolution of the rotaion detection accumulator,

$\Delta x\ \Delta y$ are the *x-y* resolutions in the translation detection accumulator,

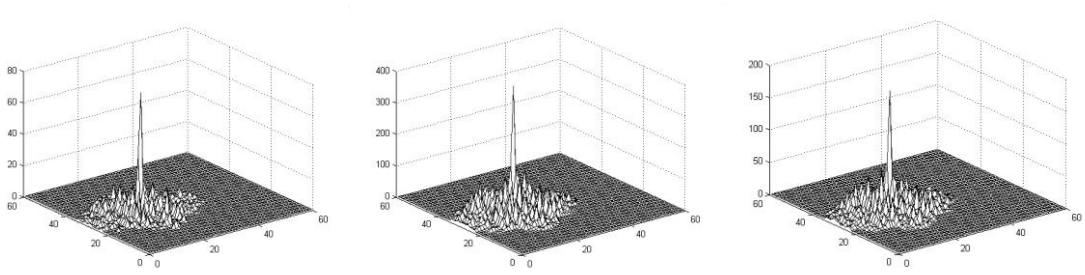
and NT is the total number of trials in the accumulator.

5.3 Results for translational motion peak detection efficiency

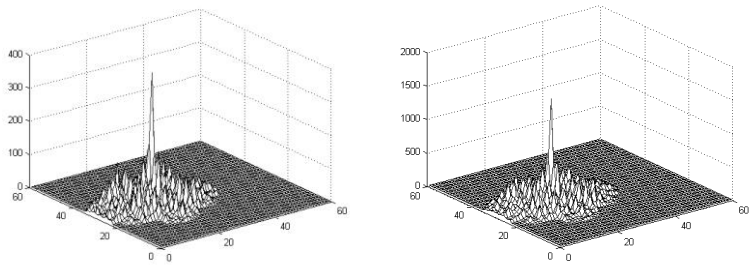
A number of simulation experiments have been conducted to investigate the peak detection efficiency for translational and rotational motions. We have used 5 algorithms corresponding to the 5 different correspondence rules. Typical displays of the 2-D accumulator spaces used for detecting translational motion are shown in Figure (6).

We have also conducted a number of experiments to determine appropriate values for the accumulator resolution, value of tolerance in the correspondence rules and effect of rotation angle on detection efficiency. It is found that the error in the peak positions is minimum for accumulator resolutions in the range 0.03 – 0.12, and values of tolerance $\varepsilon > 0.01$. For rotational motion, it is also found that such error is almost independent of the rotation angle. Accordingly, we have used in most experiments $\Delta x = \Delta y = \Delta\varphi = 0.1 \quad and\ \varepsilon = 0.01$

Table (3) lists the values of ρ_1 for different number of trials for each algorithm for detection of translational motion. These results are also plotted in Figure (7).



(algorithm1, algorithm2, algorithm3) 2-points algorithms



(algorithm4, algorithm5) 3-points algorithms

Figure (6): Typical translation detection accumulator spaces.

Table (3): Dependence of ρ_1 on *NT* for the various algorithms for translational motion (Accumulator resolution = 0.1, $\varepsilon = 0.01$)

NT(x10³)	Algorithm1 Rule (1)	Algorithm2 Rule (2)	Algorithm3 Rule (3)	Algorithm4 Rule (4)	Algorithm5 Rule (5)
5	0.02	0.1	0.04	0.12	0.36
10	0.02	0.11	0.03	0.08	0.32
15	0.0133	0.0867	0.0333	0.08	0.3333
20	0.01	0.1	0.025	0.075	0.295
25	0.012	0.112	0.032	0.072	0.296
30	0.01	0.1	0.03	0.0667	0.2833
35	0.0114	0.0943	0.0314	0.0686	0.28
40	0.0125	0.0875	0.03	0.075	0.29
45	0.0133	0.0867	0.0289	0.0822	0.28
50	0.014	0.088	0.028	0.084	0.286
100	0.021	0.084	0.04	0.082	0.3
200	0.0145	0.0775	0.034	0.0825	0.2925
300	0.016	0.082	0.0353	0.0793	0.3043
500	0.015	0.0798	0.0366	0.0786	0.3

From the experimental results we may observe that all algorithms typically fluctuate in performance at low number of trials (NT) then converge to a stable value of ρ_1 at high *NT*. Algorithms (1) and (3) are both 2-point algorithms and exhibit the lowest performance (ρ_1 values around 2.5%). Algorithms (2) and (4) exhibit almost similar behavior, and at large *NT*, the ρ_1 values are around 8%. Although algorithm (2) is a 2-point algorithm while (4) is a 3-point algorithm, the similarity in behavior is probably due to the fact that both use equal city block distance as a correspondence rule.

Algorithm (5) is a 3-point algorithm and has the highest performance. At large NT, it converges to a ρ_1 value of about 30%. The performance is probably a little better for low number of trials (about 35%) but overall, it outperforms all other algorithms. In general, rule (5) can detect translational peaks with an efficiency that is more than 3 times better than the other rules.

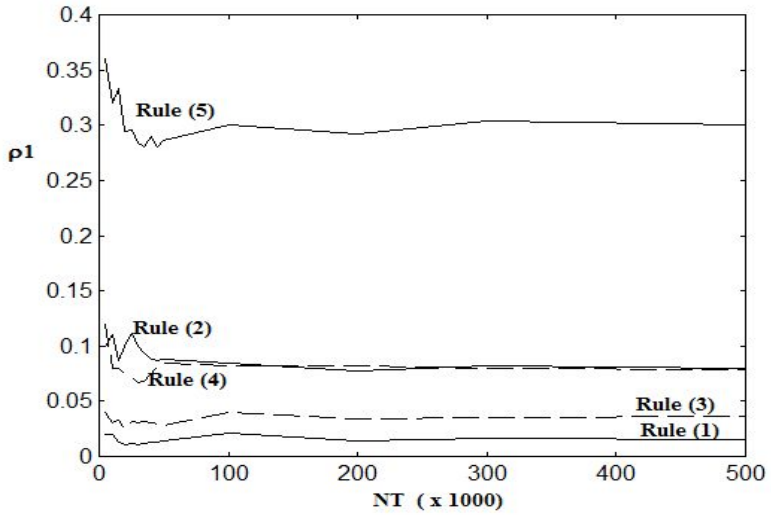


Figure (7): Peak detection efficiency for translational motion.

5.4 Results for rotational motion peak detection efficiency

For rotational motion, the results are confined to Algorithms (3) and (5) since these correspond to the rules that are invariant under pure rotation motion. Table (4) lists the values of ρ_2 for different number of trials for each algorithm for detection of rotational motion. These results are also plotted in Figure (8).

The results show that the two algorithms typically fluctuate in performance at low number of trials (NT) then converge to a stable value of ρ_2 at high *NT*. Algorithm (5) is a 3-point algorithm with a correspondence rule of equal triangular area. It exhibits a significantly higher performance (by a factor of about 8) than the 2-point algorithm (3) with the correspondence rule of equal Euclidian distances.

Table (4): Dependence of ρ_2 on *NT* for the various algorithms for rotational motion. (φ = 90°, Accumulator resolution = 0.1 rad., $\varepsilon = 0.01$)

NT(x10³)	Algorithm3	Algorithm5
	Rule (3)	Rule (5)
5	0.0040	0.0680
10	0.0060	0.0620
15	0.0080	0.0640
20	0.0085	0.0620
25	0.0076	0.0612
30	0.0074	0.0600
35	0.0072	0.0583
40	0.0075	0.0550
45	0.0071	0.0542
50	0.0074	0.0542
100	0.0072	0.0570
200	0.0062	0.0515
300	0.0066	0.0507
500	0.0070	0.0526

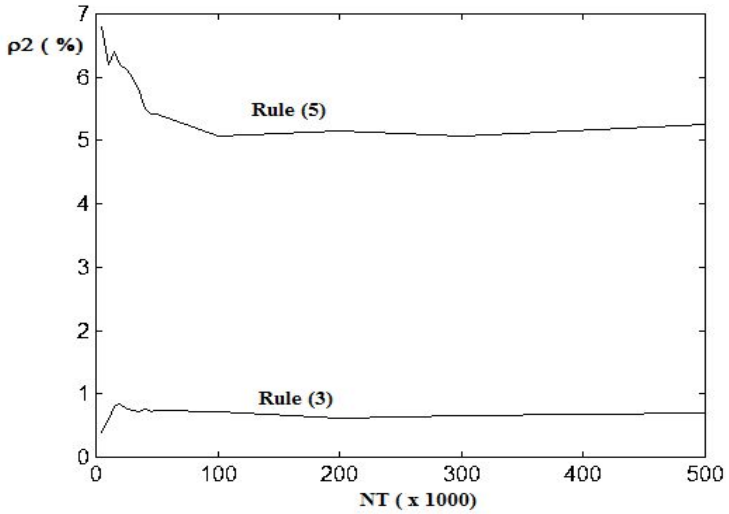


Figure (8): Peak detection efficiency for rotational motion.

5.4 Effect of noise

A number of experiments have been conducted using frames to which a salt and pepper noise component has been added. For noise levels between 2 and 4%, it is observed that the algorithm efficiency has not been significantly affected by the noise level. This shows that the present algorithm is also robust against noisy frames

6. Conclusions

In the present paper, we have investigated the effects of using five different correspondence rules to be used in MDRHT algorithms. The first three are 2-point rules that were used in previous researches. They measure correspondence through 2-point x- and y-differences, City Block distances and Euclidean distances, respectively. The present work introduced the two remaining rules for the first time. These are 3-point rules that measure correspondences through 3-point City Block distances and triangular areas, respectively.

We have developed a mathematical analysis of the invariance of the five rules given for both pure translational and pure rotational motions. We have proved that all five rules are invariant under translational motion, but only the 2-point rule for Euclidian distances and our 3-point rule with triangular areas proved to be invariant under rotational motion.

We have conducted a number of simulation experiments for motion detection using the MDRHT methodology in order to test the various correspondence rules and to measure their performance for the detection of translational and rotational motions. For this purpose, we have designed an algorithm that allows the variation of the correspondence rule, the multiplicity of point pairs, and the total number of random trials. Using this algorithm, we have investigated appropriate values for the tolerances used in applying the various correspondence rules and values for the accumulator resolution. Also, a performance parameter is introduced to measure the efficiency of peak detection in the RHT accumulator space and to compare the performances of the randomized motion detection methodology for the different rules.

For translational and rotational motions, different simulation experiments are conducted in order to investigate the dependence of peak detection efficiency on the correspondence rule and on the size of the random sample. Investigations are also made of robustness of the algorithms under noise conditions, varying angles of rotation, RHT spatial resolution, and correspondence tolerance. The results obtained for the translational motion indicated that our 3-point algorithms are in general superior to the previous 2-point algorithms. In particular, algorithm (5) that uses equal triangle areas gave the highest performance, outperforming the next in performance (2-point City Block distance) by a factor of almost 3 times.

In order to study the affect of noise on the algorithms' performance, a salt and pepper noise with different levels was added to the frames images. The results for translational motion showed that algorithm (5) again has the performance which is three times better than other algorithms and proved to be robust against noisy conditions.

Different simulation experiments were also conducted for the case of pure rotational motion. The two algorithms that proved to be invariant under rotation (using correspondence rules 3 and 5) have been tested for different rotation angles using various numbers of trials NT. Also in this case, our 3-point algorithm proved to outperform the 2-point algorithm by a factor of almost 8 times. Similar conclusions are obtained for the robustness of algorithm (5) under noisy conditions.

References

[1] Kälviäinen, H., "Applications of the Hough Transform for Image Processing and Analysis", *Pattern Recognition and Image Analysis*, 13, 187–190, 2003

[2] Xu, L., and Oja, E., "Randomized Hough Transform (RHT): Basic Mechanisms, Algorithms, and Computational Complexities", *CVGIP: Image Understanding*, 57(2), 131-154, 1993

[3] Kälviäinen, H., Oja, E., and Xu, L., "Motion Detection using Randomized Hough Transform", *Proc. 7th Scand. Conf. on Image Analysis, Aalborg, Denmark,* 72-79, 1991

[4] Kälviäinen, H., "Computational considerations on Randomized Hough Transform and Motion Analysis", *Proc. Conf. On Image Processing: theory and applications, San Remo, Italy*, 87-90, 1993

[5] Kälviäinen, H., Oja, E., and Xu, L., "Randomized Hough Transform applied to Translational and Rotational Motion Analysis", *Proc. 11th Int. Conf. on Pattern Recognition, The Hague, The Netherlands,* 672-675, 1992

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

[7] Leung, M.K., and Huange, T.S., "Detecting Wheels of Vehicle in Stereo Images", *Proc. of 10th Int. Conf. on Pattern Recognition, Atlantic City, USA,* 263-267, 1990

[8] Kittler, J., and Pamler, P.L., "Robust and statistically efficient detection of parametric curves in 2D images", *NSF/ARPA Workshop on Performance and Methodology in Computer Vision, Seattle*, 1994.

[9] Ji, Q. and Haralick R.M., "Error propagation for the Hough Transform", *Dept. of Electrical Computer, and systems Engineering, Rensselaer Polytechnic Institute, Troy, NY, USA,* 2000.

[10] Jean, J.H., and Wu, T., "Robust visual servo control of mobile robot for object tracking in shape parameter space", *43rd IEEE Decision and Control Conf.,* 4016 – 4021, 2004

[11] Behrens, T., Rohr, K., and Stiehll, S.H., "Robust segmentation of tubular structures in 3D medical images by parametric object detection and tracking", *IEEE trans. Systems, Man & Cybernetics, Part B: Cybernetics,* 33(4), 554 – 561, 2003

[12] Kiryati, N., Kälviäinen, H., and Alaoutinen, S., "Randomized or Probabilistic Hough Transform: Unified performance evaluation", *Pattern Recognition Letters*, 21, (13-14), 1157 - 1164, 2000

[13] Nakajima, S., Zhou, M., Hama, H., and Yamashita, K., "Three-Dimensional Motion Analysis and Structure Recovering by Multistage Hough Transform", *Proc. of SPIE Conf. on Visual Communications and Image Processing, Boston, USA*, 1605(2), 709-719, 1991

[14] Heikkonen, J., "Recovering 3D motion parameters from optical flow field using randomized Hough Transform", *Pattern Recognition Letters,* 15, 971 – 978, 1995

[15] Xu, L., and Oja, E., "Randomized Hough Transform", *Encyclopedia of AI, Ed. Dopico, J., et al., IGI Global Publ.,* 1354 – 1361, 2005.