

FUZZY TUNED STOCHASTIC SCANPATHS FOR ROBOT VISION: A DEVELOPMENT REVIEW

M. J. Allen*, G.M. Dimirovski^{††}, N. E. Gough* and Q. H. Mehdi*

* School of Computing & Information Technology,
University of Wolverhampton, Wolverhampton, WV1 1EQ, UK
Email: mike.allen@wlv.ac.uk; Fax: +44-1902-331-776

[†] Faculty of Electrical Engg., SS Cyril and Methodius University
P.O. Box 574, 1000 Skopje, Republic of Macedonia

^{††} Department of Computer Engineering, Dogus University
Acibadem, Kadikoy, 81010 Istanbul, Turkey

Keywords: Autonomous vehicles, fuzzy logic, mobile robot control, robot vision, scanpaths.

Abstract

The real-time processing of frame sequences obtained from cameras mounted on autonomous mobile robots and vehicles is a computationally intensive task. This paper is a review of the work carried out so far in the development of a procedure using Fuzzy Tuned Stochastic Scanpaths (FTSSs) for efficiently scanning images in a frame sequence. A concise explanation of FTSSs is given here followed by a summary of the experimental work that has been undertaken to date. The results show how the technique can reliably locate objects in scene whilst examining only a fraction of the image surface, e.g. 5%. The paper concludes with a discussion of the project so far and proposes ideas for future work.

1. Introduction

This paper documents the work undertaken within the MIST (Multimedia and Intelligent Systems Technology) Research Group at the University of Wolverhampton to design and develop a technique using FTSSs for the efficient frame sequence analysis. When traversing in complex environments, the ability of autonomous mobile robots to assimilate complex visual data in real-time [1]-[3] is an important aid to the navigational process [4], [5]. The aim of the FTSS procedure is to *efficiently* scan the images contained in a frame sequence by radically reducing the surface area of the frame that has to be examined. The procedure achieves this by reasoning with data ascertained

from the examination of previous frames and prior knowledge of the environment. The information gained from this reasoning process is used to direct the scan of the subsequent frame.

The FTSS technique was motivated by the operation of the human eye and its interaction with the brain. In Section 2, an overview of the current work in the field of human vision and visual systems inspired by the human eye is given. Section 3 describes the FTSS procedure and how it functions. This includes a breakdown and explanation of the major elements. Section 4 describes the development of the FTSS procedure, the experimental work undertaken to date and the results obtained. Section 5, contains a conclusion of the work to date and a discussion on future work.

2. Establishing Attention within a Scene

The inspiration for the FTSS technique originated from the operation of the human eye and its interaction with the Central Nervous System (CNS). When examining a scene the eye constantly enacts saccades (rapid eye movements) and uses multiple fixations (periods when the point of regard is relatively still) to build up a visual representation within the mind. A review paper by Henderson and Hollingworth [7] highlights a number of studies, which establish that informative regions of a scene receive more eye fixations than those of lesser importance. The technique described here uses this concept to analyze each frame contained within a frame sequence by determining Regions Of Interest (ROI) that contain salient objects and monitoring these regions from frame to frame.

The FTSS technique uses historical data from previous scans and environmental knowledge to establish which parts of the subsequent frame should be examined. It has been established that the human model combines visual data and experience (expert knowledge) to anticipate future events in order to reduce reaction times. For example, in fast action sports, the expert is able to distinguish between the irrelevant and relevant information and to recognize these cues early in the sequence of events [9]. Hendersen *et al* [7] describe a framework for understanding eye movement control in scene viewing. The method uses a *saliency map* to determine the subsequent fixation point. The *saliency map* contains a set of weights that are initially assigned according to the saliency of a region based on a parse of the entire scene. It is suggested that these initial saliency metrics are assigned by low-level stimulus factors such as colour (visual information as opposed to semantic information). The attention is turned to the region with the highest weight. Over time as the information within the regions of the map are processed and become meaningful, the saliency weights are modified to reflect the semantic importance of the regions. Consequently, the basis for each weight gradually moves from being visually based to semantically based. In this work the FTSS procedure determines the next fixation point(s) using a *saccade map* which shares similarities with the *saliency map* described by Hendersen *et al* [7]. However, the *saccade map* is constructed from an analysis of historical data and a stochastic process is used to determine the subsequent fixation point(s). Norton and Stark [8] coined the term *scanpath* to describe the sequence of fixations used by a particular viewer when viewing a particular pattern. Stark *et al* [10] present a review of current *scanpath* theory for top-down visual perception. Although a more simple approach is adopted in this work, FTSSs create a *scanpath* for analysing the images in a frame sequence.

3. The FTSS Procedure

3.1 Overview

A complete description of the FTSS procedure can be found in Allen *et al* [1]. However, the main elements of the FTSS procedure are shown in

Figure 1 and the main steps of the process are as follows:

- each frame is divided into sub-regions.
- historical data from previous scans (see Section 3.2) regarding each sub-region are analyzed using a FIS to establish how interesting (salient) a sub-region is.
- the output from the FIS is used to construct a *saccade map* (a 2D matrix) where the saliency of each sub-region is represented by a value from 0 (not salient) to 1 (very salient).
- a weighted roulette wheel is used in conjunction with the values contained within the *saccade map* to construct the *scanpath*.
- the *scanpath* is a string of sub-regions that are to be examined in the subsequent frame.
- the data contained within each sub-region of the *scanpath* are analyzed to ascertain if any of these sub-regions are salient, i.e. a ROI.
- the data ascertained from the analysis of these sub-regions is added to the historical data and used when constructing the next *scanpath*.

The size of the sub-regions is determined by the size of the objects being sought. In the early experimental work the size of the sub-regions was determined by the programmer. However, the adaptive procedures have the ability to vary the size of the sub-regions at run-time (see Section 3.3). The FIS is used to infer a crisp output from the historical data and determines the saliency of a particular sub-region. The possibility information is used in conjunction with a weighted roulette wheel to decide which regions should be used to form the *scanpath* that is used to scan the subsequent frame. Consequently, the informative regions are assigned a relatively high possibility and are more likely to be selected for examination. The *scanpath* consists of a string of sub-region locations. In the early work the length of the scanpath (the number of sub-region locations it contained) was predetermined. However, in the adaptive procedure the length of the scanpath could be varied at run-time (see Section 3.3). Finally, the sub-regions contained within the *scanpath* are examined to ascertain if they contain objects of interest. In this work colour data has been used to classify the sub-regions as being salient or otherwise. Artificial Neural Networks (ANNs) have been used for the classification process.

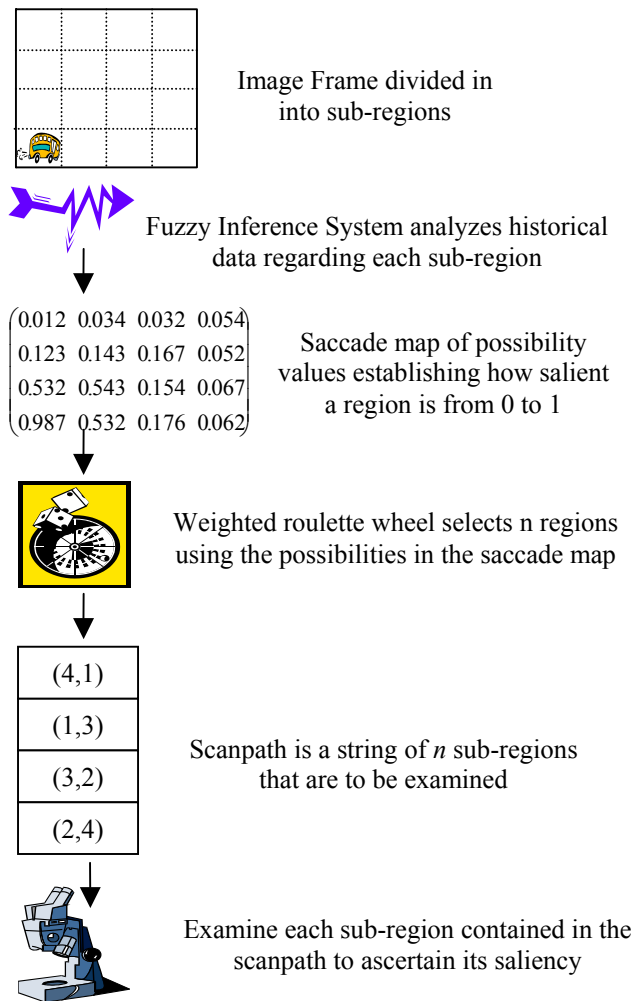


Figure 1: An illustration of FTSS procedure

3.2 Input Metrics to FIS

A *Mamdani* FIS was used exclusively in this project to determine the possibilities contained within the *saccade map*. There is no specific reason why this form of FIS was adopted and other formats, such as the Sugeno FIS, should be examined in future work to identify if they offer any advantages. The input metrics to the FIS are *attention*, *strategy*, *distance* and *bias*, and each metric is normalized between 0 and 1. The *attention* and *distance* metrics have been proposed [6] and were used in the initial work. The *strategy* and *bias* metrics were introduced in later procedures. A detailed explanation of the metrics can be found in [6].

The input metric *attention* determines the possibilities within the saccade map for those sub-regions that are not local to a ROI (an area containing an object). Its function is to promote a

systematic scan of pixels in areas of low significance in order to search for new objects. If only *attention* is used to determine the possibilities in the saccade map, and there are no classifiable objects present, the attention paid to each sub-region should ideally be equal. However, in a random process each sub-region should receive approximately the same amount of attention.

The *strategic* input metric encapsulates environmental knowledge by identifying areas of constant interest. A *strategic map* is created by the program designer during the initialization of the procedure and areas of constant interest can be identified on this map, e.g. the areas around entrances and exits. The *strategic map* is similar to the saccade map in that each element corresponds to a sub-region in each frame. In much of the work an area of strategic interest was denoted by 1 and all other areas are denoted by 0. However, in the latest procedures a mechanism for determining areas of negative strategic interest can also be assigned, e.g. areas where permanent fixtures are located. The degree of *strategic* interest can also be assigned. In these procedures a value in the range 0.5 to 0 would be assigned to areas of negative significance - 0.5 is average significance to 0 being of no significance. An area of average significance might be a corridor whereas an area of no significance may be a restricted area where objects of interest will never enter. Conversely, areas of positive *strategic* interest are assigned values between 0.5 and 1 - 0.5 is average to 1 which is highly significant. An area may be assigned a rating of 0.9 if it is close door a doorway that is a major point of ingress/egress from the area being examined. Sometimes it is desirable to reduce the attention given to sub-regions of positive strategic values. For example, if an object is not identified in a particular frame then the attention paid to sub-regions with strategic significance should be reduced until the object is relocated. This refinement is achieved by introducing a *bias* metric that is described later.

The input metric *distance* determines the distance of a sub-region from the nearest ROI. The metric is calculated in terms of map cells (sub-regions) using the *chessboard* measure. The *distance* metric aims to focus attention on sub-regions of particular

interest by assigning high possibilities to those sub-regions containing or local to objects or agents. The *bias* input metric monitors *system status* and is used to reduce the possibilities in the saccade map assigned to those regions not local to an object if the procedure fails to identify a previously identified object during a particular scan. When the *bias* is invoked, a different set of fuzzy rules are used that assign lower possibilities to sub-regions not local to a ROI. Consequently, more attention is paid to a known ROI in order to re-establish the position of the lost object on the next scan.

3.3 Adaptive FTSSs

Three factors have to be considered when FTSSs are used to scan image frames in order to locate an object(s): the size of the sub-regions (the resolution of the saccade map), the number of sub-regions included in the *scanpath* (the length of the *scanpath*), and the characteristics of the FIS which influences how attention is distributed around each frame. Initially these factors were predetermined by the system designer and remained constant throughout the procedure. The procedure could not respond to environmental factors or changes in object/agent behaviour. The performance of the technique, therefore, relied upon the skill of the designer. In recent work the concept of using Adaptive Fuzzy-Tuned Scanpaths (AFTSSs) for object location has been introduced (Section 4.4).

The procedure was enhanced so that the resolution of the saccade map could be adjusted at run-time. The resolution of the *saccade map* is determined by the relative size of the objects that are to be located. To be identified, an object must predominantly occupy at least one sub-region. Therefore, if an object occupies a relatively small area of each frame then a map of higher resolution is required, i.e. to sub-divide the frame into smaller sub-regions. The next step was to enable the FTSS procedure to adjust the length of the *scanpath* at runtime. A number of factors can have a bearing on how long the *scanpath* should be, e.g. the number of objects in the scene and the degree of positional displacement for each object from frame to frame.

3.4 Object Classification

An object classification technique was required to identify the data contained within the sub-regions

on the image surface. These regions could contain areas of a single object or areas of different objects and these objects could be either stationary or mobile. The use of colour appears to be an obvious feature to use in the classification process because a sub-region can be classified as belonging to one object, or a reduced set of objects, based exclusively on the data compiled from pixels it contains. Features, such as shape, are useless as the sub-region may not contain the entire object. In each of the experiments an ANN was used for the classification process. A brief description of the ANNs used in each experiment is given in this work. A detailed explanation of the ANN and the pre-processing of the colour data can be found in the papers that accompany the experimental work.

4. Development of FTSS Technique

4.1 Conception of FTSS Technique

Griffiths *et al* [6] describe a technique for scanning simple scenes using a *scanpath* that is tuned using a FIS and introduced the term Fuzzy Tuned Stochastic Scanpaths (FTSSs). The technique can be used to establish ROI within a scene and it is proposed that this data can be used to direct a camera, mounted on a pan and tilt head, towards informative areas of the scene. A *saccade map* is used to influence the *scanpath* (see Section 2.1). The information contained within each cell of the *saccade map* represents the *possibility* that a corresponding region is a ROI because a particular object or agent is located in that area. In this process the *possibility* value for each cell is derived using a FIS with the input metrics *distance* and *attention* (see Section 2.2). To evaluate the technique a set of 10x10 binary image matrices were used and the object was represented by a 1 in an otherwise zero matrix. The experimental work did not analyze real image data and the binary matrices were of relatively low resolution.

4.2 FTSSs to locate a real object in a frame sequence

In Allen *et al* [2] the FTSS technique proposed by Griffiths *et al* [6] was adapted to analyze a frame sequence captured using a digital video camera. This new procedure was designed to locate and monitor the position of a single mobile object whilst continuing to systematically scan the rest of

the frame. The environment containing the object is uncluttered. A radial basis neural network was used to classify the nature of each sub-region by analyzing the colour data it contained. The procedure was implemented and experiments undertaken. This work established that FTSSs could be used effectively with real image data captured from a real environment.

The image sequence was captured using a progressive scan. The image frames were downloaded from the camera and initially stored as 640x480, 24-bit colour files. For hardware/software compatibility reasons the images were then converted to 8-bit indexed colour images and stored as bitmap files. A digital image sequence was captured of a yellow ball rolling across the image scene (Figure 2). The ball (approximately 50 pixels in diameter) enters from the left hand side of the image and 38 frames charted its progress across the plane of view before it disappeared at the right hand side. The camera was held stationary during the capture of the sequence and only two object areas were present in each frame, the ball and the blue textured carpet that formed the background. This constrained scenario was chosen to ensure that the objects did not change in size, that the lighting conditions were held constant and to ensure that the objects could be easily classified using colour. In the experimental work a predetermined 20x20 *saccade map* was used to identify 400 sub-regions of 32x24 pixels in each of the 38 frames. For each frame the length of the *scanpath* was fixed at an arbitrary value of twenty sub-regions - 5% of the frame surface. These parameters were chosen based on the size of the object relative to the frame size and some initial experiments. The colour data from each sub-region selected in the *scanpath* was classified to ascertain if it was a ROI, i.e. it contained the object and not background. Four fuzzy membership functions (MFs) were used to fuzzify the input parameter *distance* as zero, small, medium and big, and one MF was used to fuzzify the input parameter *attention*. Five fuzzy rules were used, four for *distance* and one for *attention*. The *distance* rules were all weighted as 1 whereas the *attention* rule was weighted as 0.01 to ensure that greater emphasis was given to the rules related to *distance*. This resulted in the assignment of higher possibilities to regions local to the object. This was



Figure 2: Images 3,19 and 36 from frame sequence with yellow ball on a blue background

the only part of the experimental work where weightings other than 1 were used in conjunction with the fuzzy rules. In subsequent procedures it was found that it was not necessary to add different weightings to the rules to obtain the desired mapping. Having different weightings caused confusion during the design of the FIS. Four MFs were used to form the fuzzy *output* sets that result from implication - zero, small, medium and big. The fuzzy output sets were then aggregated together to form one final set. In order to have a control, the progress of the ball was first analysed using a more intensive procedure designed to accurately calculate the actual position of the ball, i.e. the centroid was calculated to ascertain the *x* and *y* position of the ball in each frame. These were taken to represent the *actual ball positions* that could be compared with those returned by the experimental procedure - *the estimated positions*. The main aim of the procedure is to detect the object in the image and not to accurately track the centroid of the object. However, the error between the actual position of the ball and the estimated position is used to establish that the procedure locates the correct object area.

Mean Error (<i>x</i> axis)	Mean Error (<i>y</i> axis)	Mean Error (Euclidean)
14.638	11.545	20.519

Table 1: Summary of results for rolling ball

Since the FTSS procedure uses stochastic methods it was necessary to repeat the experiment a number of times and present the results statistically. The results contained in Table 1 were acquired by performing the test 50 times and comparing the estimated positions with the actual positions. During each run the vertical and horizontal errors were calculated, in terms of pixels, for each of the 38 image frames, i.e. the absolute difference between the *actual* and *estimated x* and *y* positions for the centroid of the ball.

In terms of efficiency these results were obtained whilst examining only 5% of the surface of each frame. The first experiment thus provided promising results and formed an encouraging foundation for further work. However, it was clear that a more rigorous testing process was required to determine: how many times, in terms of frames, the system fails to locate the object; how the system copes when focussing attention on more than one object; and how computationally intensive are the system overheads e.g. the scan process and the production of the *saccade map*. It was also concluded that the method could be improved by incorporating further input metrics in the fuzzy system to accommodate changes in environmental conditions by changing the parameters of the FIS.

4.3 FTSS procedure to locate two objects

This work was originally published in Allen *et al* [1]. In this procedure the *bias* and *strategic* input metrics were added to the existing *distance* and *attention* metrics (Section 2.2). Two objects were monitored in a constrained environment and a more comprehensive set of results was collected from the test procedure. The test procedure counted the number of frames in which the procedure failed to locate a particular object and, on these occasions, the number of times the procedure located the lost object on the following frame was also noted.

In the FIS, thirteen fuzzy rules were used with each rule carrying the same weighting of 1. Triangular MFs were used throughout. The fuzzy inputs for *distance* and *attention* were each defined using sets of four MFs. The inputs *bias* and *strategy* just act as switches which were either on or off, i.e. 0 or 1. The fuzzy output set that results from implication is defined using seven MFs. As in the first experiment, the image frames were sub-divided into 400 equally sized sub-regions using a 20x20 matrix and consequently each sub-region was 32x24 pixels. During one iteration of the scan cycle, a sub-region was selected and classified based on the colour of the pixels it contained. Twenty sub-regions were contained within each *scanpath* - 5% of the frame surface. These parameters were chosen based on the size of the objects and some initial experimentation.

The frame sequence was captured and stored on the hard disk as in the first experiment. The sequence contained 56 frames and, as shown in Figure 3, three objects were contained within each frame: the textured blue, white and grey background, a green pyramid and a yellow ball. The pyramid was static whereas the ball was always mobile. As the background never changed the aim was to focus attention on the ball and pyramid in each frame.

The scene was illuminated from one side to create areas of shadow so that the experiment was more representative of the conditions encountered in a real environment. Some parts of the background are in shadow and one side of the green pyramid is completely in shadow (see Figure 3).

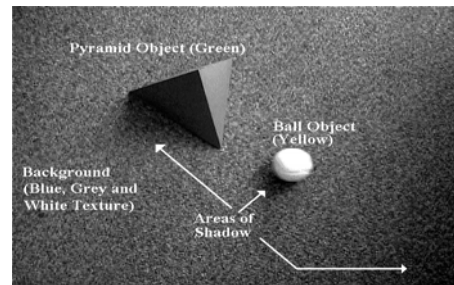


Figure 3: Sample frame from the image sequence

The actual positions of the ball where analyzed using the same procedure described in the previous section so that a comparison could be made with the experimental technique. The centroid of the pyramid was calculated from the known position of the three points on its base. These *actual positions* were used with the experimental findings to find the resulting *experimental errors*. These results are summarized in Tables 2 and 3. The results show the mean results extracted from 40 test runs.

A number of different measures have been employed to grade the success of the method. In addition to measuring the location error along the horizontal, vertical and Euclidean error, the procedure also measures the *Identification Failure* and the *Instant Recovery*. *Identification Failure* measures the number of times during one test (56 frames) the procedure fails to locate the object in a particular frame. If the system fails to locate an object or object during the examination of a particular frame the system puts forward the location of that object in the previous frame as its estimated position. *Instant Recovery* is the number of times after failure that the procedure relocates

Measures	Analysis of all Data	Analysis (Outliers Removed)
Longitudinal Error	18.56 pixels (3.9%)	12.44 pixels (2.3%)
Latitudinal Error	74.70 pixels (11.7%)	17.20 pixels (2.7%)
Euclidean Error	80.75 pixels (10.1%)	23.59 pixels (3.0%)
Identification Failure	30.75 iterations (54.9%)	25.82 iterations (46.1%)
Instant Recovery	8.63 iterations (28.1%)	10.30 iterations (40.0%)

Table 2: Scan results for the rolling ball

Measures	Analysis of all Data	Analysis (Outliers Removed)
Longitudinal Error	17.82 pixels (3.7%)	16.10 pixels (3.4%)
Latitudinal Error	24.30 pixels (3.8%)	23.03 pixels (3.6%)
Euclidean Error	33.67 pixels (4.2%)	31.58 pixels (3.9%)
Identification Failure	7.23 iterations (12.9%)	6.46 iterations (11.5%)
Instant Recovery	5.50 iterations (76.1%)	5.56 iterations (86.1%)

Table 3: Scan results for the green pyramid

the object on the next iteration. The percentage error is calculated as proportion of the identification failure.

Due to an inconsistency in the object classification process a *rogue threshold* was included to eliminate the opportunity for misclassifications. The threshold ensured that sub-regions classified as a particular object were within 3 cells of that object's expected position. However, if the object was not located within 5 iterations of the start the procedure could never locate that object. This inconsistency was eradicated in subsequent experiments. In 7 of the 40 tests, the system failed to locate the ball within the first 5 iterations. Therefore, the two sets of results presented in

Table2 evaluate the system with these seven results included and without them respectively. With regard to the pyramid, one set of data was very different to the rest and therefore Table 3 also has two sets of results, one with the rogue result included and the other with it removed.

In conclusion, when only examining 5% of the frame surface the procedure was able to locate a static object with an overall accuracy of 24 pixels and mobile object with an accuracy of 32 pixels (Euclidean measure). The *Identification Failure* (see Tables 2 and 3) of the ball was nearly four times greater than it was for that of the pyramid: the average failure rate on the ball was about 46% compared with about 11.5% for the pyramid. As the position of the ball changed from frame to frame it presented the procedure with a more challenging problem and the results reflected this. However, on average the procedure located the centroid of the ball with more accuracy than that of the pyramid. This was attributed to the relationship between the size of the objects (the pyramid occupies approximately four times as much of the image than the ball) and the size of sub-region of the image. The results of this experiment suggest that a self-tuning system, using the *bias* as a variable control rather than a switch, should be developed. These experiments were carried out in carefully controlled conditions in a simple scenario and hence the complexity of the experimental environments should be gradually increased to reflect the complexity of a real environment. These should include different lighting scenarios to identify how well the system can handle factors such as changes in the ambient light source, the presence of multicoloured objects and variations of illumination intensity.

4.4 Adaptive FTSSs

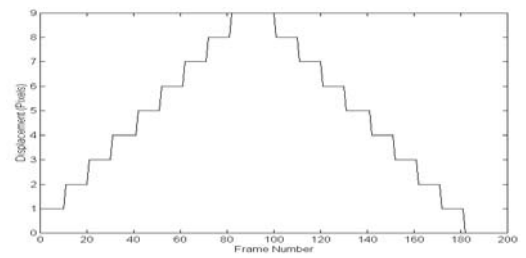
The adaptive technique presented in Allen *et al* [3] (see Section 3.3) has the ability to change the size of the sub-regions, alter the length of the *scanpath*, accommodate any number of objects up to a predetermined maximum and does not assume that any of the objects are present within the scene until they have been positively identified. The procedure also assesses whether an object has been lost and discontinues monitoring its position until it is relocated. In addition, areas within the environment

of negative and positive strategic importance could be specified. Up until now only areas of positive strategic importance could be specified.

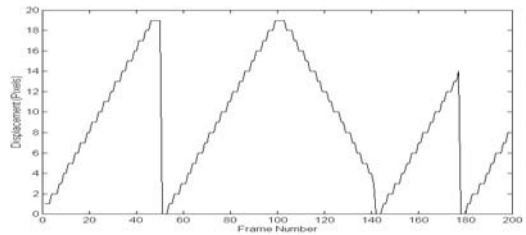
In order to clearly observe the function of this new technique, more test data was required about the procedure's operation at run time. For example, the size of the sub-regions selected by the procedure and the length of the *scanpath* were needed in order to assess whether the technique was displaying the desired behaviour. Consequently, the first test was carried out using an artificial frame sequence because the parameters could be accurately controlled and recorded, i.e. the exact size, displacement and position of the objects on the screen. Data about the behaviour of the objects on the screen can be represented graphically and compared with the behaviour of the procedure. For example, when all the objects on the screen are moving very quickly, the length of the *scanpath* needs to be increased to accommodate this factor.

Although the use of an artificial frame sequence simplified the object classification procedure the classification procedure still had to generalize if more than one object was present within a particular sub-region. A multi-layer perceptron having a single hidden layer with four units and an output layer also consisting of four units was chosen to classify the image data.

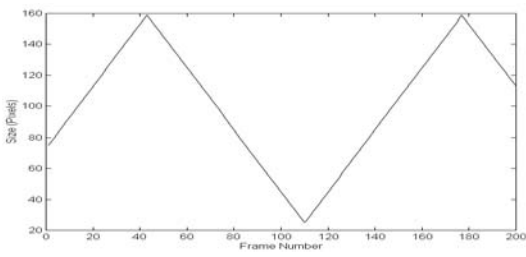
The frame sequence consisted of 200 frames, each 640 x 480 pixels, with a white and grey background and two mobile objects – one red and one green. Both objects changed size at the same rate but the size of Object 1 was allowed to vary over a wider range (see Figures 4c and 4d). Object 2 was able to increase and decrease the rate of displacement from frame to frame (0.4 pixels/frame) more quickly than Object 1 (0.1 pixels/frame) and over a wider range (see Figures 4a and 4b). The objects could move freely around the white area of the background but could not enter the grey area. This grey area was located in the bottom right hand corner of each frame and was 81 x 81 pixels in size. The experimental procedure was able to step the resolution of the saccade map between 20 x 20 cells and 40 x 40 cells. The length of the *scanpath* could be varied to scan between 5% and 20% of each frame in increments of 2%. Decisions to adjust either of these parameters were based on the



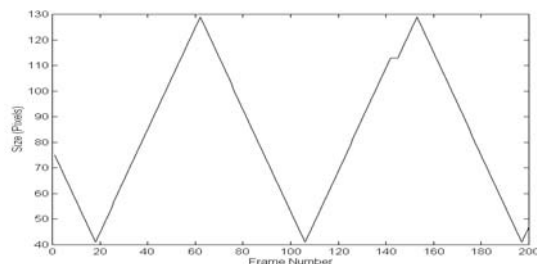
(a) Rate of Displacement of Object 1



(b) Rate of Displacement of Object 2



(c) Size of Object 1 ((x,y) dimensions)



(d) Size of Object 2 ((x,y) dimensions)

Figure 4: Characteristics of objects

performance of the procedure over the preceding 3 scans. The resolution of the saccade map was increased if the average number of locations r for the most unreliably located object of the previous three scan was less than 1 and the resolution was decreased if r exceeded 3. The length of the *scanpath* was increased or decreased if r fell below or exceeded 1 and 2.1 respectively. The initial resolution of the *saccade map* was 20x20 cells and the length of the *scanpath* was 10%. No initial indications of the objects positions were given.

The results given in Table 4 provide information regarding the reliability of the procedure in locating each object from frame to frame and its overall accuracy. In addition, the table gives the mean,

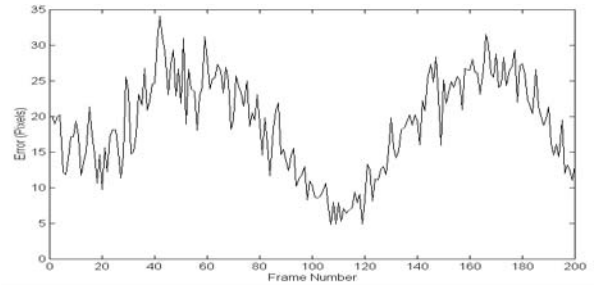
maximum and minimum areas of each frame examined by the procedure in order to estimate the position of the objects. Figures 4a and 4b display the rate at which the objects changed position and Figure 4c and 4d establish the size of each object over the 200 frames in the sequence. Figure 5a and 5b show the degree of error (the accuracy) for the procedure when locating the two objects. Figure 5c and 5d show the resolution of the saccade map and the length of the *scanpath* at each frame. The information in Table 4 shows that the procedure reliably located both objects throughout the test (90.8% and 87%) and to within approximately 12 pixels along the horizontal and vertical axes for both objects. This was achieved whilst only analyzing an average of 7.5% of each frame. There

	Object 1	Object 2
Location Reliability (%)	90.8	87
Vertical Error (pixels)	12.5	12
Horizontal Error (pixels)	12.2	11.8
Euclidean Error (pixels)	19.3	18.6
Mean area of frame examined (%)	7.5 (min area 6.0, max. area 14.2)	

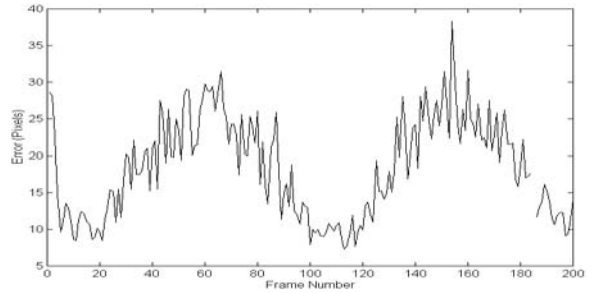
Table 4: Mean results for two objects over 20 tests

is a clear correlation between the size of the objects (Figures 4c and 4d) and the accuracy of the procedure (Figures 5a and 5b), since an increase in the size of the object triggers a reduction in accuracy and vice versa. Between frames 100 and 120 the objects were at their smallest (at their most difficult to locate) and both were exhibiting a relatively high level of displacement from frame to frame as seen in Figure 4. At this point the procedure radically increased the length of the scanpath to compensate as shown in Figure 5d (Frames 100 to 120). After this period the scanpath returned to its previous length.

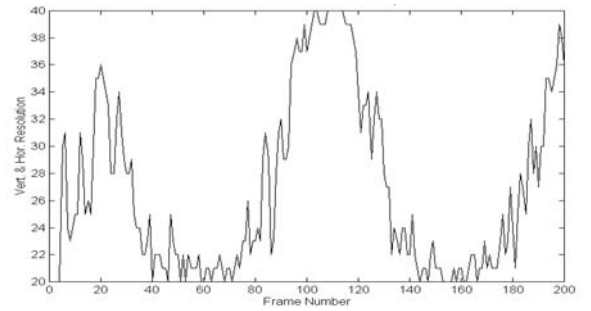
In conclusion, this modified procedure differs from those that preceded it in that it can adapt the resolution of the saccade map and the length of the *scanpath* to suit the requirements being made upon it by changes in the environmental conditions. This makes the system more accurate, more reliable and



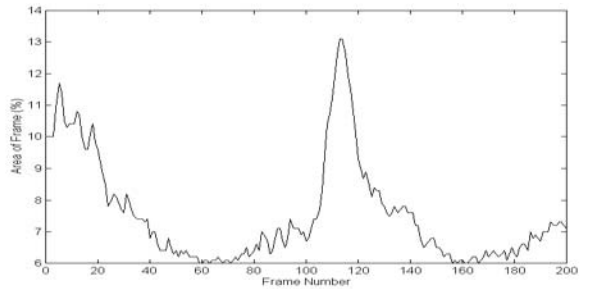
(a) Errors for Object 1



(b) Errors for Object 2



(c) Mean resolution of Saccade Map



(d) Mean Area examined of each frame

Figure 5 Performance of experimental procedure

makes it less reliant on the skill of the designer to set these parameters at the outset. The procedure can still be tuned by selecting the threshold at which the procedure implements changes.

5. Conclusions and Further Work

In this paper the development of the FTSS procedure has been discussed and the important elements of the experimental work have been documented. Building on the work of Griffiths *et al* [6] the FTSS procedure has been refined so that it

can analyze real image data, locate and monitor more than one object and adapt to changing circumstances within the environment. The new procedure described in Section 4.3 is more advanced than the earlier procedures. It is more flexible in that it can accommodate a variable number of objects and can recognize when an object has been lost or has left the scene and act accordingly. However, this procedure could still benefit from further refinement. Two objects of the same colour could cause problems to the technique and, at present, there is no means of ranking the importance of different objects so that different levels of attention can be applied to each. There are still parameters that must be predetermined by the operator, i.e. the strategic data. The procedure, over time, should be able to identify the areas of strategic importance as negative or positive. A method for intelligently assessing the strategic importance of different areas would be a useful addition. If the *strategic* parameter could be linked to the world map of a robot the procedure could predict where a visual system on a mobile robot should be looking based on its current position. The fuzzy system has been a powerful tool for constructing the *saccade map*, in that its behaviour has been carefully defined using the rules. It would be rather interesting to investigate other methods in this role, i.e. ANNs. The process of analyzing the sub-regions has not been covered in depth in this work. However, this element of the process is vital and further research in this area is currently being undertaken so that the FTSS procedure can be used in more complex environments. Currently, the procedure runs in MATLAB. The procedure has proved that it can locate objects efficiently in terms of image surface examined. The next step is to examine how the procedure can be programmed on a real time platform to test its efficiency in terms of meeting hard real time targets.

Acknowledgements. Professors Dimirovski and Gough would like to acknowledge the respective academic and state institutions in the R.M. and the U.K. for supporting in part their long lasting academic and scientific co-operation.

References

- [1] M. J. Allen, Q. Mehdi, I. J. Griffiths, N. E. Gough, I.M. Coulson. "Object location in colour images using fuzzy-tuned scanpaths and neural networks", In *Proceedings of the 19th SGES Int. Conf. on Knowledge Based Systems and Applied Artificial Intelligence*, Cambridge, UK, pp 302-314 (1999).
- [2] M. J. Allen, Q. Mehdi, N. Gough. "Stochastic scanpaths for efficient frame sequence analysis", In *Proceedings of the 12th SCS European Simulation Symposium*, The Society for Computer Simulation International Publication, pp 185-189. (2000).
- [3] M. J. Allen, I. J. Griffiths, I. M. Coulson, Q. Mehdi, N. E. Gough, "Efficient tracking of coloured objects using fuzzy-tuned scanpaths", In *Proceedings of the Workshop on Recent Advances in Soft-Computing*, de Montfort University, Leicester, UK, pp.225-230. (1999).
- [4] G. M. Dimirovski, N. E. Gough, O. L. Iliev, A. T. Dinibutun, O. Kaynak. "A contribution to fuzzy-logic analytical-simulation methods for AGV motion navigation", In O. Kaynak, M. Ozkan, I. Tunay (eds.) *Proceedings ICRAM95 on Recent Advances in Mechatronics*, Bogazici University, Istanbul, vol. II, pp. 678-683 (1995).
- [5] G. M. Dimirovski, Z. M. Gacovski, K. Schlacher, O. Kaynak. "Mobile non-holonomic robots: On trajectory generation and obstacle avoidance", In P. Kopacek (ed.) *Robot Control (SYROCO 2001)*, Pergamon Elsevier Science Ltd., Oxford, pp. 545-554 (2001).
- [6] I. J. Griffiths, Q. H. Mehdi, N. E. Gough. "Fuzzy tuned stochastic scanpaths for AGV vision", In *Proceedings of the Int. Conf. on Artificial Neural Networks and Genetic Algorithms*, Norwich, UK, pp. 88-92 (1997).
- [7] J. M. Hendersen, A. Hollingworth. "Eye movements during scene viewing: An overview", In *Eye Guidance in Reading and Scene Perception*, Elsevier Science Ltd, Oxford, pp. 269-294 (1998).
- [8] D. Norton, L. Stark. "Scanpaths in eye movements during pattern perception", *Science*, **171**, pp. 308-311 (1971).
- [9] G.C. Paull, D. J. Glencross. "Expert perception and decision making in baseball", *Int. J. of Sport Psychology*, **28**, pp. 35-56. (1997).
- [10] L. W. Stark, C. M. Privitera, H. Yang, M. Azzariti, Y. F. Ho, T. Blackmon, C. Dimitri. "Representation of human vision in the brain: How does human perception recognize images", *J. of Electronic Imaging*, **10(1)**, pp. 123-151 (2001).