Technical University of Denmark

DTU

An integrated environment for fast development and performance assessment of sonar image processing algorithms - SSIE

Henriksen, Lars

DTU Library
Technical Information Center of Denmark

An Integrated Environment for fast Development and Performance Assessment of Sonar Image Processing Algorithms - SSIE

Lars Henriksen

Institute of Automation
Technical University of Denmark
Building 326, DK-2800 Lyngby, Denmark, lh@iau.dtu.dk

Abstract - The Sonar Simulator Integrated Environment, SSIE is a tool for developing high performance processing algorithms for single or sequences of sonar images. It formed gradually under the design of a sonar system for the AUV MARTIN which is able to detect and track objects based on data from an electrically scanned high resolution imaging sonar. The tool is based on MATLAB providing a very short lead time from concept to executable code and thereby assessment of the algorithms tested. Also the SSIE utilizes the graphical user interface facilities readily available in MATLAB making it conceptually easy and fast to operate. These two properties; short implementation time and swiftness of operation leads to high speed algorithm development. Therefore the final algorithms to be implemented on the application will be of better quality as more iterations can be made within a given time schedule. A problem related to testing and hence optimization of the algorithms is the availability of sonar images. This problem is twofold: First it is cumbersome and expensive to record data in the field. Secondly the scene viewed can only be partly controlled why the value as reference data degrades. To accommodate this problem the SSIE has been equipped with a simulator capable of generating high fidelity sonar images given a scene of objects, sea-bed AUV path etc. In the paper the main components of the SSIE is described and examples of different processing steps are given. The work is in collaboration with partners under MAST-CT90-0059.

I. INTRODUCTION

For a number of simple AUV tasks such as maintaining a certain altitude above the sea-bottom low resolution sonars are sufficient sensors. For more complex tasks like precise navigation, detection of obstacles and mapping of the sea-bed a higher resolution is desirable. Low resolution sonars may still fulfill the task but the cost expressed by e.g. consumption of time and energy increases. For some operations a high resolution sonar may be explicitly required e.g. for completing a task in a complex area while maintaining vehicle integrity. The fact that a wide range of commercially available sonars offers high resolution without much need for external equipment makes this kind of sonar attractive for especially AUV applications.

At the same time being the advantage the basic problem in using high resolution sonars is that the output is much more complex than for low resolution sonars. The high resolution sonar produces an image that can be compared to that of a video camera in complexity. This demands more advanced algorithms for both interpretation of the output as well as for navigation. At the same time the rate of which the data is available is very high. An output data rate of approx. 1 Mb/sec. is not unusual putting a limitation on the complexity of the algorithms for a given hardware configuration. These two facts, a lot of data and a complex processing of these, indicate that considerable effort needs to be put into the development of the processing algorithms consuming both time and economic resources.

Due to these difficulties the algorithms found may not be thoroughly tuned and tested before implemented on the sonar system. This may lead to sub-optimal performance of the target AUV or maybe even to hazardous situations. In order to reduce these hardships the Sonar Simulator Integrated Environment, SSIE has been developed. The paper describes the different components of the SSIE and their respective functions.

A. Target sonar and application

The SSIE is currently configured for the SeaBat 6012 high resolution sonar which is operating at 455 KHz [7]. The image is divided horizontally into 60 beams each with a width of $1.5°$, covering $90°$ in front of the sonar. Each beam is sampled to a resolution of 5 cm. Vertically the image has only one plane covering $\pm10°$. The range of the sonar is adjustable up to 200m. The sampling rate is 0.9 MB/sec which yields an image update frequency of approx. 7 images per second at a range of 100m. Later versions of the sonar will have 3 beams in the vertical direction.

The target application of the object detection system is the AUV called MARTIN which is designed for operation close to the seabed at depths at approx. 600m [1]. See figure 1.

II. FUNCTIONS OF SSIE

The primary objective of the SSIE was to achieve an environment for fast development of sonar image processing algo-
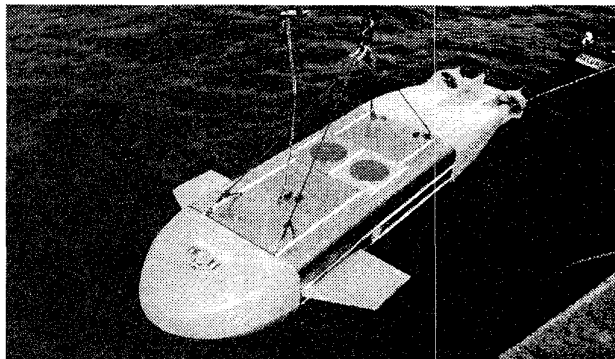
FIGURE 1. Target vehicle MARTIN.

rithms. This was accomplished partly through the establishment of a library of image processing algorithms where the user can add new or modify existing user defined functions. The image processing is described in section III. The different actions in this library are easily accessed through a specially designed graphical user interface, GUI featuring e.g. roll-down menus and slider adjustment of values of variables, see figure 2.
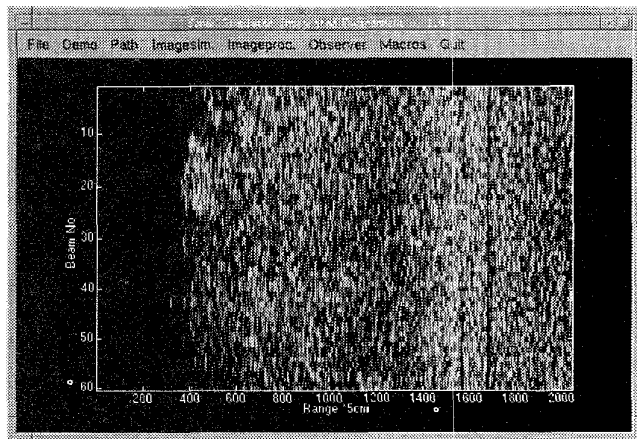

FIGURE 2. The graphical user interface of SSIE.

This facility of operation increases speed of development and the user feels encouraged to perform a more comprehensive evaluation of the algorithms under design leading to better results other things being equal.

Having detected a number of objects in a series of sonar images the precision can be enhanced by an observer. The SSIE suits a library of approaches, choices of source data and different choices for display of results. The observer is treated in section IV.

Besides a library for image processing the SSIE also contains tools for generating synthesized images. The benefit of this is twofold: First the time consuming and costly process of recording of sonar images is reduced. Naturally some real life images are still necessary for validation purposes but the images needed to perform the main part of the development can be generated synthetically. Secondly the scene is known quite accurately which is often not the case in real experimental set-ups. This enables more subtle test such as assessment of the sensitivity of the image processing algorithms to different environmental settings. The image simulation is described in section V.

The SSIE also contains a menu point for demonstration use. This demo- facility basically consist of commands for fast retrieval of sonar images and pre-generated example results. These results covers different steps in the image processing and synthesizing process, movies, example paths of objects and the AUV etc. The demonstration menu point can also be used in the development phase by providing fast access to selected example sonar images. Besides facilitating operation the GUI has a neat appearance making the SSIE even more suitable for demonstration use.

In summary the SSIE offers a wide range of functional capabilities organized as menu points. The four most used are

- Image processing
- Observer
- Image simulator
- Demonstrator

These central functions are supported by a set of utilities providing file handling, definition of the paths of the AUV and the objects, recording of movies for animated documentation, different macros, tools for image presentation (polar/ carthesian), various print commands etc.

III. IMAGE PROCESSING

The raw sonar image is generally of poor quality due to the difficulties in acoustical underwater sensing. As this is inherent to the sensing method, no countermeasures can be taken in the sensing phase:

- The echo is highly dependent on objects material, surface and orientation.
- The echo is highly dependent on sonar position and orientation.

As most of the acoustical energy is typically dispersed in many directions by the contact with the target only little energy is returned to the sonar for detection. The echo of typical objects is therefore generally low. This is inconvenient especially in near-bottom operation where bottom reverberation in many cases will be the dominant signal. When this reverberation arise from a bottom of sand or gravel the amplitude of the

334

signal from the bottom will be especially significant due to diffuse reflection.

These facts indicate that interpretation of the sensed signal is quite difficult and that the possibility of missing objects can not be neglected. Unfortunately the difficulties cannot be overcome completely by the designer of the system as the problems arise due to environmental variables and laws of physics. However there are some points which may and should have influence on the choice and tuning of the image processing algorithms. These can be seen in the example image shown to the right in figure 11:

- Different bias due to bottom reverberation. Bottom reverberation is seen from 18 m.
- Occasional drop-outs at certain ranges due to interference with other sonars or other acoustic sources. Recognized as dark arcs at approximately 78 and 82 m.
- Variations in the time varying gain. Dark belt from 50 to 60m, light belt from 70 to 80m.
- Noise from bottom reverberation. From 18 m.
- Shadow effect is not significant

How to use this a priori information is illustrated by an example application. The SSIE was developed as a part of a project for design of an object detection system ODS which is to provide information about position, velocity, and size of objects in the near surroundings of the AUV. These data should be available sufficiently early for the path planner to generate an evasion manoeuvre. This means that the range of the sonar sensor should be large enough to allow initial detection of the object, processing of several consecutive images to reduce uncertainty on measured values, and to allow room for the AUV evasion manoeuvre with a safe margin between the AUV and the object. At the same time the precision of the required data should be high in order to generate the best path. These two goals are conflicting, as sonar resolution decreases with larger range. The object detection system designed to comply with theses demands consists of four logical successive processing stages:

- Equalization
- Filtering
- Detection
- Observers (if sequence of images. Treated in section IV)

Each of the four steps are in principle independent of the others enabling a modular structure simplifying modification and upgrading of separate steps. The structure also makes pipelined processing possible.

A. Equalization

The equalization reduces the bias effects from bottom reverberation, the effects of interference from other sonars and effects from sonar inconveniences such as different gain in different beams and deviation in the time varying gain. It equalizes the average value of all pixels at each range to a common value.

B. Filtering

The filter is be designed to serve two purposes: First it should minimize the noise that could otherwise disturb the detection algorithm. Secondly it should suppress signals from objects which are smaller than a predefined certain size. This is in order to ignore objects that are not essential to vehicle safety. This minimum size should be selected on basis of objects expected i.e. knowledge about the bottom conditions. The design of this low pass filtering step is crucial to the performance of the overall system as it suppresses the noise which is a primary degrading factor of the sonar image quality. Design of the filter involves a number of choices: filtering in time or space, 2 or 1 dimensional filtering, FIR/IIR, different linear and non-linear types of filter techniques etc. The result of an example set of candidate filters is shown in figure 3. These filters are one-dimensional operating along the beams.
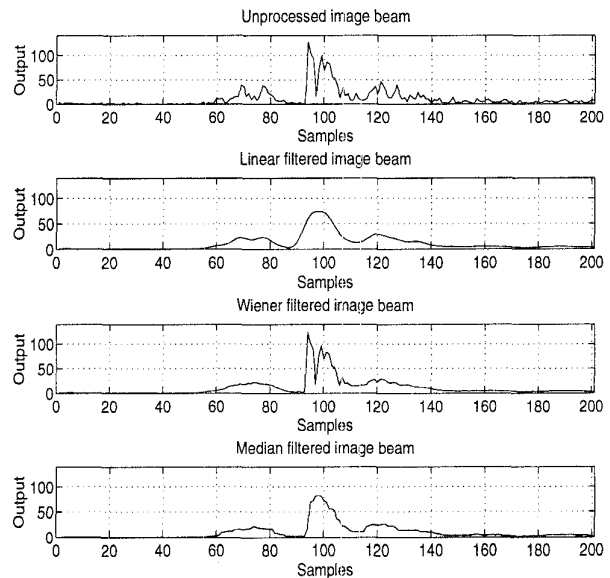

FIGURE 3. Image beam filtered using various library functions of the SSIE: lowpass FIR, Wiener and Median filters.

The appropriate filter can then be selected based on a trade-off between edge preservation, object size preservation, noise suppression and processing time and complexity.

335

C. Detection

The fastest way to detect objects is to use a one level threshold. This method is however relative sensitive to noise why a two level method is employed. The high threshold level is used to detect the existence of an object while the low level is used to determine the extent of the object. The two level approach is computationally more complicated but the gain in output quality reduces the need for successive processing. It is to be noted that an object is detected on its peak/highlight and not the shadow. This is because peaks seem more dominant on the front looking sonar SeaBat6012 than on e.g. side looking sonars.

The objects detected are represented as an array of centres and radii of circles circumscribing each object in the image. As the sonar yields information in one plane only the circles physically represents cylinders normal to the sensing plane of the sonar. The result of an example sequence of equalization, linear filtering and two level thresholding is seen on figure 4.
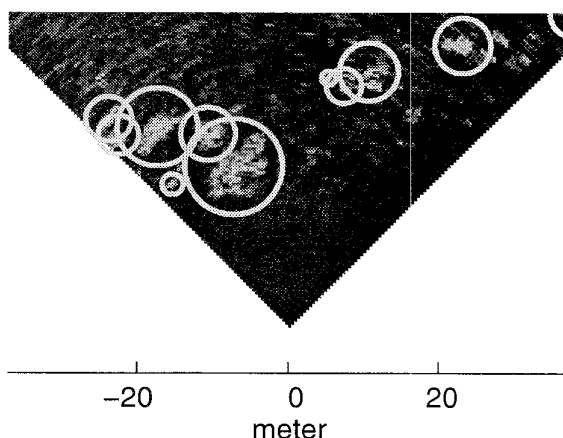

FIGURE 4. Close-up of objects detected in the image shown to the right in figure XX.

Here it is noted that bottom reverberation, drop-outs and inconveniences from the time varying gain are greatly reduced. Furthermore noise is lowered significantly leaving a homogeneous image where the objects are easily visible. Hence the detection step is able to determine the position and size of several objects which are only difficultly recognizable in the original image. It is also able to exclude signals that appear to be objects in the raw image.

IV. OBSERVER AND TRACKER

Due to the difficult sensing technique objects may be missed from sample to sample and the positions of those objects detected are noise corrupted. By use of event driven Kalman filters a stable array of objects can be maintained. Also the speed and heading can be estimated enabling an improved path planning. These filters are relatively complicated to design partly as they need to associate multiple detected objects with objects predicted by the filters. Furthermore they require the position of the AUV in a world fixed coordinate system which can be difficult to extract under water.

A. Structure

Available input for the observer is the object coordinates in a body fixed coordinate system, z_{ob}, obtained from the sonar. The position and orientation of the AUV in a global fixed system, z_{ag} and R is known as well. This is generated by a primary Kalman filter based on various sensors, including a set of transponders. This observer is not to be discussed here. The position of the objects in global coordinates, z_{og}, can be found by equation 1, where R is the rotation matrix describing the orientation of the AUV.

$$z_{og} = z_{ag} + R^{-1}z_{ob} \qquad (1)$$

Representing the objects in world coordinates has the advantage that the observer becomes linear and thus more simple. Another benefit is the possibility to scan a large area of the seabed, generating a map of objects in earth fixed coordinates for use on future missions.

The state vector for each object consists of position and orientation of the object as described in [8]. This is supplemented by an additional parameter S_{kr} which is the radius of the circle circumscribing the object

$$S_{kx} = \begin{bmatrix} x_k \\ \cdot \\ x_k \end{bmatrix} \quad S_{ky} = \begin{bmatrix} y_k \\ \cdot \\ y_k \end{bmatrix} \quad S_{kz} = \begin{bmatrix} z_k \\ \cdot \\ z_k \end{bmatrix} \quad S_{kr} = r \qquad (2)$$

Each of these 4 parameters have their own independent Kalman filter. As objects may dis- and reappear in the image event driven filters are feasible.

It can be argued that velocity is unnecessary to estimate as many obstacles are fixed at the sea-bottom but e.g. AUVs, ships and trawl require that velocity can be different from zero.

B. Extended tracking

A detected object will eventually move out of the field of the sonar due to the movements of AUV and object. At this point the update is switched from Kalman filter (called active state) to time update only (stand-by state). In this way objects can be tracked even when out of contact. This is of importance, if the task of the AUV makes it likely that it will return to the same site later.

In the time update mode the estimation precision of the state of the object naturally decreases with time both due to the initial estimation uncertainty and due to changes in direction and speed of the object. After a number of updates the information

336

is of so little value to the path planner that update is no longer beneficial. At this point the state of the object is hibernated i.e. stored together with at time stamp if velocity is considered close to zero. Another reason for the hibernation of the state is that unrestricted tracking will concern a constantly increasing number of objects each consuming processing power. When to hibernate an object mainly depends on the uncertainty of the state of the object. This can be measured by the covariance matrix exceeding some predefined level or just that a number of updates have passed.
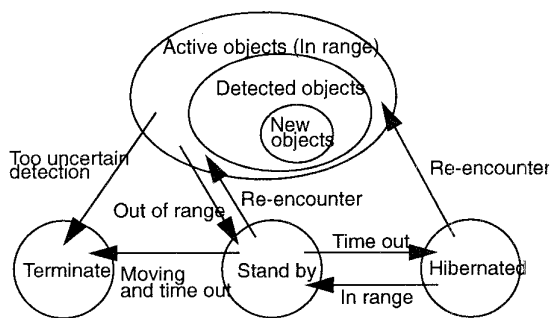

FIGURE 5. The four logical states of an object; Active, stand-by, hibernated and terminated

If speed cannot be considered zero the object is discarded as the object is not suited for fixation in a map through hibernation.

C. Example

The benefit of the observer is illustrated through an example, see figure 6. The AUV starting point was (x,y,z)=(0,0,4)m and the ending point was (50,0,4)m. Roll, pitch and yaw were all 0 degrees. Object no 1 (dark) and object no 2 (light) were both moving as described in figure 7,

At t=4 the measurement noise on object no 1 becomes too large for correct identification why a new object is initiated. At time t=26 the uncertainty of the estimates of the original object grows too large and it is therefore discarded. This shows that the physical object is tracked even though objects may be missed in identification in the internal representation. At t=41 object no 1 turns to stand by state as it is no longer in the field of view of the sonar. In this state the position of the object is extrapolated using time updates in the kalman filter only.

The estimates of the velocities are relatively uncertain due to the heavy noise on both measurements of the position of the objects in the sonar image and on the position and orientation of the AUV, see figure 7.

It is especially noted that the estimated velocity in the z direction is not correct due to the very low resolution of the sonar in the horizontal direction. The change in z position is 8 m on both objects (not shown) but is not caught by the observer
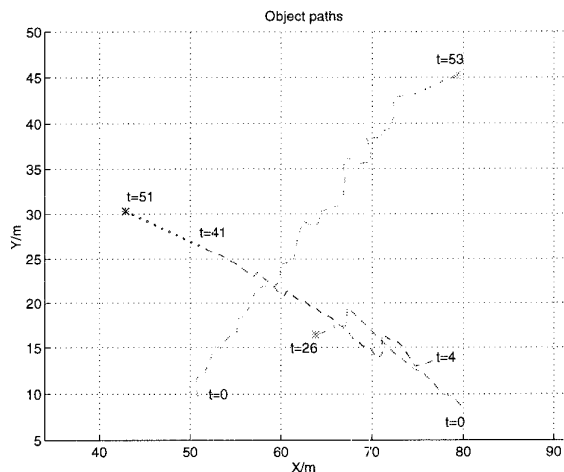

FIGURE 6. The estimated (x,y) path of the objects from time t=0 to t=80 seconds. (x,y) spans a horizontal plane.

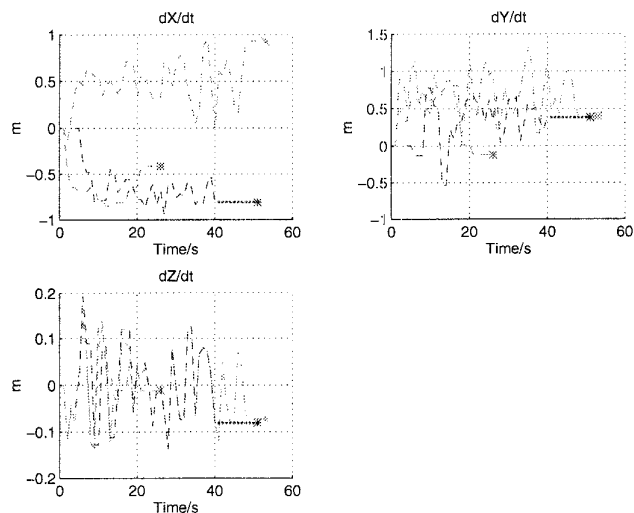

FIGURE 7. Estimated values of velocities: d(x,y,z)/dt. Object no 1 (dark): d(x,y,z)/dt=(-0.7,0.4,0.1). Object no 2 (light): d(x,y,z)/dt=(0.5,0.7,0.1).

(dz/dt=0.1 m/s for 80 seconds). The radii are found with good precision (3 and 6 m).

V. SIMULATION

The SSIE also contains tools for generating high fidelity sonar images based on reflection equations for three generic objects: Cylinders, spheres and planes. These three objects can emulate the most frequent occurring objects encountered in near-bottom operation: the sea-bed, quays, barrels etc. The simulator can also add sonar imperfections and interference from other acoustic sources etc. This simulation feature serves two

337

purposes. First it reduces cost of acquiring sonar image data. Secondly it enables direct evaluation of the result of the image processing as the scene is controlled completely in contrast to a real-life scene. The synthesizing of sonar images also makes it possible to generate a sonar image sequence by defining the paths of a set of objects and of the AUV. In this way it is possible to simulate a mission before it actually takes place. The SSIE can thus be used as a tool in the mission planning phase.

A. Reflection of sound waves

A model of the reflected sound should consist of two components, specular and diffuse sound. The diffuse sound reflection is represented by the Lambertian model scattering the incident sound uniformly in all directions. The reflected sound equals the incident sound which is proportional to $\cos(\alpha)$ where α is angle of incidence. The specular sound is calculated as reflected by a rough surface which is modeled by a number of gaussian distributed facets [4]. The emitted sound from the surface is therefore a lobe symmetric around a pure specular direction. When emitter and receiver is at the same position the angle of incidence and angle of reflection are equal. The relation between reflected intensity and incoming intensity, γ can thus be described as equation 3.

$$\gamma(\alpha) = C_d\cos(\alpha) + C_s G(\alpha)\frac{1}{\cos(\alpha)}\exp\left(-\frac{\alpha^2}{2\sigma_\alpha^2}\right) \quad (3)$$

C_d and C_s denote the percentage of diffuse and specular reflection. σ_α is a factor describing surface roughness. $G(\alpha)$ is a geometric attenuation factor:

$$G(\alpha) = min\left(1, 2(\cos\alpha)^2\right) \quad (4)$$

The surface of the object reflecting the sound wave determines whether the reflection is mainly diffuse or specular. Materials like uncorroded iron, plastic and sand all have irregularities with a very small period compared to the wavelength of sound why they will reflect specularly at almost any angle of incidence. For these not infrequently occurring materials the sound is reflected away from the sonar and are therefore difficult to detect especially when inclined relative to the sonar beam. The more inclination the more difficult detection.

The signal intensity received by the sonar is not only a function of the sound reflected by the object $\gamma(\alpha)$ but does also depend on the emission and reception characteristics of the sonar. For the example front looking sonar the reception characteristic is equal to the emission characteristics in the vertical plane $S(\beta)$ which is depicted on figure 8.

The final relation between emitted and received intensity is described by equation 5.

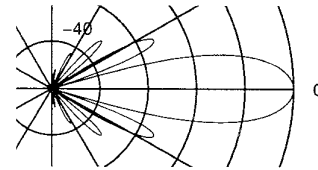$$\frac{I_{out}}{I_{in}} = \gamma(\alpha) S^2(\beta) \quad (5)$$

FIGURE 8. Horizontal sensitivity of SeaBat 6012 sonar; $S(\beta)$ (10 dB/line).

As illustrated by figure 9 side lobes may induce mirror objects as sound is received from angles off-axis. This effect is however moderate if the beam forming is reasonable (-25db max for the SeaBat 6012)
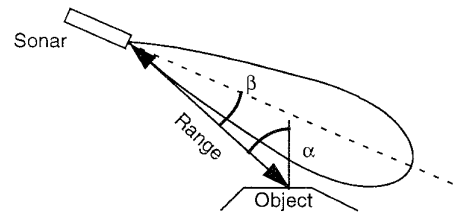

FIGURE 9. Configuration of sonar and object.

B. Bias and noise from bottom reverberation

In near-bottom operation the sonar will receive an amount of reverberations from the bottom. These reverberations consists of a bias and a noise contribution due to speckle phenomena. The bias is a result of the overall reflection of the sound wave and is a function of the emission and reception characteristics of the sonar and the position and orientation of the sonar relative to the bottom according to equation 5. A typical profile of the echo in one beam is shown in figure 10. Note the weak echoes in the range from 4-10 m caused by the side lobes of the sonar emission/reception characteristics.
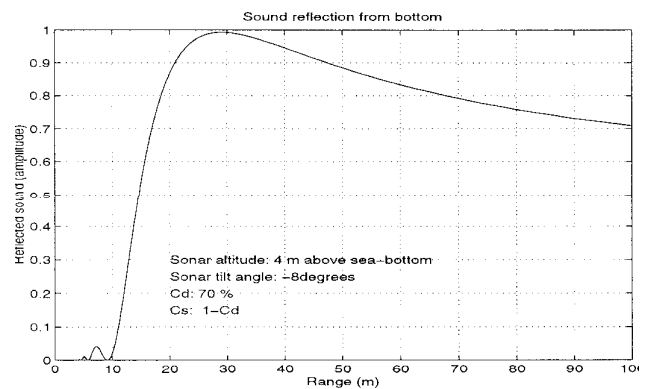

FIGURE 10. Intensity of one beam in sonar image viewing the sea-bottom.
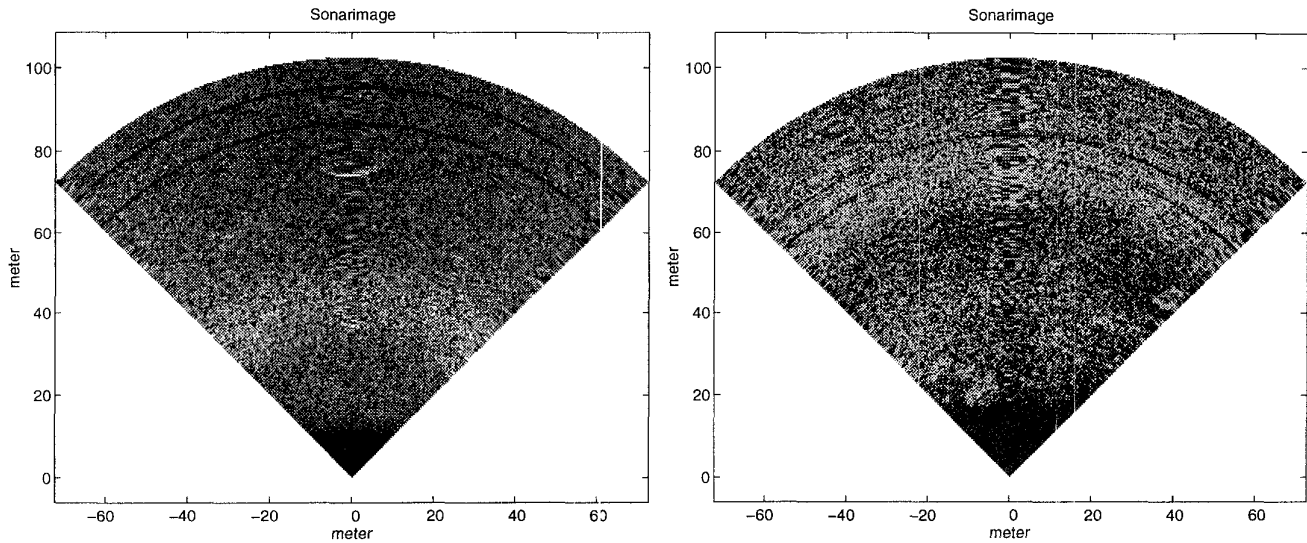
338

FIGURE 11. Left: Synthesized sonar image with three objects, bottom reverberation, speckle noise and sonar inconveniences. Object echoes can be identified at (x,y)=(0,35), (0,75) and (-30,60)m. Right: Real-life sonar image of gravel bottom at Skagen, Denmark.

Figure 10 is for a flat and unobstructed lateral surface. In real applications the sea-bed will consist of slow variations in height and simulation fidelity is thus enhanced by modeling the sea bed as multiple facets.

As the occurring noise is due to speckle phenomena the noise is simulated as random noise. This has shown to be a reasonable trade-off.

C. Sonar imperfections and interference from other acoustic sources

The output signal of the sonar does not only consist of sound intensity received from objects in the scene. As indicated on the right sonar image in figure 11 drop-outs identified by dark arcs at certain ranges clearly disturb the signal. A possible explanation is that other acoustic sources interferes with the sound of the sonar. Also 'drop-ins' due to variations in the time varying gain occur as slow varying intensity at certain ranges which are not related to the shape of the sea bed. Both imperfections are easily simulated. An example of the final synthesized image is found to the left on figure 11. To the right a real image for comparison.

VI. PLATFORM AND EXPERIMENTAL IMPLEMENTATION

The platform of the SSIE was chosen to be MATLAB from Mathworks. This offers several advantages: The SSIE is portable as MATLAB can run on various types of computer systems e.g. DOS or UNIX based. It is easy and fast to modify the code as no time consuming compilation is necessary. Also the MAT-LAB language resembles C to a large degree which facilitates generation of MATLAB-code as no new language has to be learned. Furthermore porting the code to run-time C-code is relatively uncomplicated and thereby fast (was done for the MARTIN project). MATLAB supplies different toolboxes for e.g. signal processing, image processing, optimization problems all speeding up the design process. The MATLAB also provides a powerful tool for easy design of a graphical user interface [5].

A. Simulation and processing in SSIE

In table 1 some typical values of the consumption of resources when generating a synthesized sonar image is shown. The MATLAB code was not subject to optimization with regard to processing time.

Subject	KFLOPs	CPU secs on HP9000/735
3 Cylinders & spheres	1511	85
Sea bed	3846	103
Sonar inconveniences	125	7
Noise	615	29
Total	6097	224

TABLE 1. Typical consumption of resources in sonar image synthesizing.

The table shows that it is possible to achieve a high fidelity sonar image of a user specified scene in less than 4 minutes on the given platform. This is considerably less than the time consumption of acquiring real-life images.

339

B. Processing algorithms: SSIE and C implementation

For the initial experiments and tuning flexible hardware is needed. The MVME 162 Motorola 68040 25 MHz single board computer was selected for this purpose. Apart from its computational and communication capabilities the board was chosen because of its modularity and reliability in a potentially rough environment. In the current configuration the operation system is OS9 capable of real-time execution.

In table 2 the time consumption of selected sonar image processing algorithms is shown for both VME 162 and SSIE

Mode	CPU secs on HP9000/735	CPU secs on MVME 162
Equalization	1.27 s	0.14 s
Filtering	1.76 s	1.06 s
Object detection	22.5 s	0.27 s
Total	25.53 s	1.47 s

TABLE 2. Typical time consumption of selected sonar image processing.

The processing time for an image in SSIE clearly states that fast performance assessment is possible in the design stage. The corresponding numbers of the VME 162 shows that as the board is a general purpose computer data cannot be processed at the high rate they are supplied by the sonar. Therefore different data reducing arrangements mainly based on focus of attention have been employed. One of these is called the multirate/range method, see [3]. This approach assigns processing power to the different areas of the image in accordance to the interest for the application. For ensuring vehicle safety the ranges close to the AUV is the most important. When both multirate/range and subsampling in range data is employed the rate of processing can be increased by at least a factor of 3. This does not comply with the 7 Hz update rate of the sonar but it is sufficient for most surveys as the AUV can move only 1m between update of obstacles.

When the final algorithms for sonar image processing have been fully developed they will be transferred to dedicated hardware. As the algorithms furthermore are suitable for pipelined processing a considerable increase in update rate is expected.

VII. CONCLUSION

The design of the object detection system for the MARTIN AUV has lead to a general purpose tool SSIE for development of processing algorithms for high resolution sonar images.

The tool offers both fast generation and execution of code. It is also conceptually easy and hence fast to operate. This leads to high speed algorithm development and often the final algo-

rithms implemented on the application will be of better quality as more iterations can be made within a given time schedule. The main functions of SSIE are libraries of processing and synthesizing of images, macros, utility functions etc. For sequences of images an observer/tracker can be applied.

Future work will concentrate on the ability to perform surveys in a virtual environment and in this way provide a foundation for optimization of the actual survey.

IX. REFERENCES

[1] Lars Henriksen, Anders Bjerrum and Anders Ishøy. 'Sea Trials of MARTIN - A European Survey AUV'. MTS/IEEE OCEANS '95 San Diego, CA. USA. october 1995.

[2] Lars Henriksen, Morten Lind, Kjeld Lynge Hansen, Allan Theill Sørensen and Ole Ravn. 'Use of sonar for underwater inspection'. Technical report, Institute of Automation, DTU 1995.

[3] Lars Henriksen. 'Real-time Underwater Object Detection on an Electrically scanned High-Resolution Sonar'. IEEE Oceanic Engineering Society's Symposium on Autonomous Underwater Vehicle Technology, AUV '94. Cambridge, MA, USA, July 1994.

[4] D. Langer, M. Hebert. Building Elevation Maps From Underwater Sonar Data. 7'th International Symposium on Autonomous Unmanned Untethered Submersible Technology. University of New Hampshire, September 1991.

[5] 'Building a Graphical User Interface'. MATLAB manual 1993.

[6] B. Shridhar, P. Smith, R. Suorsa, B. Hussein. 'Multirate and event-driven kalman filters for helicopter flight'. IEEE Control Systems, vol. 13 No 4 1993.

[7] C. J. Simonsen. 'Mine locating and disposal, the need for real-time scanning sonars'. Presentation made at UDT '93 in Cannes.

[8] Glen N. Williams, Glenn E. Lagace, Alfred J. Woodfin. 'A collision avoidance controller for autonomous underwater vehicles'. IEEE Oceanic Engineering Society's Symposium on Autonomous Underwater Vehicle Technology AUV '90, Washington, DC, USA, June 1990.

340