2011-09

An effective noise filtering method for mine detection

Hong, Bryan Y.

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

AN EFFECTIVE NOISE FILTERING METHOD FOR MINE DETECTION

by

Bryan Y. Hong

September 2011

Thesis Advisor:	Peter C. Chu
Second Reader:	Ronald E. Betsch

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704–0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE September 2011	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE An Effective Noise Filtering Method for Mine Detection	5. FUNDING NUMBERS N6230610PO00223
6. AUTHOR(S) Bryan Y. Hong	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA93943–5000	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) Naval Oceanographic Office Stennis Space Center, MS 39529 - 6000	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.IRB Protocol number _____N/A_____.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	12b. DISTRIBUTION CODE

13. ABSTRACT (maximum 200 words)

Automatic detection of sea mines in coastal regions is difficult due to highly varying sea bottom conditions present in the underwater environment. Detection systems must be able to discriminate objects that vary in size, shape, and orientation from naturally occurring and man-made clutter. Additionally, these automated systems must be computationally efficient to be incorporated into Unmanned Aerial Vehicle (UAV) sensor systems characterized by high sensor data-rates and limited processing abilities. Commonly used noise filters largely depend on the window (or neighborhood) size, which makes the mine detection ineffective. Using the bi-dimensional empirical mode decomposition (BEMD) analysis, an effective, robust sea mine detection system can be created. A family of decomposed images is generated and applied to optical lidar image data from the Rapid, Overt, Airborne, Reconnaissance (ROAR) experiment supplied by Naval Surface Warfare Center, Panama City. These decompositions project key image features, geometrically defined structures with orientations, and localized information into distinct orthogonal components or feature subspaces of the image. Application of the BEMD method to the analysis on side scan sonar data is also provided.

Accurate detection and classification of mines is time consuming and requires divers or Autonomous Underwater Vehicles (AUV) in the water. The navy continues to pursue more expedient methods in mine countermeasures, and with airborne lidar, a surf zone (SZ) and landing zone can be quickly surveyed for possible mines. In the near surf zone, all possible mines can be quickly neutralized by dropping guided munitions, eliminating the need to send divers or AUVs to verify contacts. Still, the need for improved methods of detection and classification is needed. BEMD, a relatively new method of signal analysis developed about fifteen years ago, was tested on lidar imagery from the ROAR experiment to look for any improvements in detecting and classifying mines.

14. SUBJECT TERMS Bi-dimensional empirical mode decomposition, filter, mine detection, physical oceanography	15. NUMBER OF PAGES 94
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UU

NSN 7540–01–280–5500 Standard Form 298 (Rev. 2–89)
Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

AN EFFECTIVE NOISE FILTERING METHOD FOR MINE DETECTION

Bryan Y. Hong
Lieutenant, United States Navy
B.A., Thomas Edison State College, 2001

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN METEOROLOGY AND PHYSICAL OCEANOGRAPHY

from the

NAVAL POSTGRADUATE SCHOOL
September 2011

Author: Bryan Y. Hong

Approved by: Peter C. Chu
Thesis Advisor

Ronald E. Betsch
Second Reader

Jeffrey Paduan
Chair, Department of Oceanography

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Automatic detection of sea mines in coastal regions is difficult due to highly varying sea bottom conditions present in the underwater environment. Detection systems must be able to discriminate objects that vary in size, shape, and orientation from naturally occurring and man-made clutter. Additionally, these automated systems must be computationally efficient to be incorporated into Unmanned Aerial Vehicle (UAV) sensor systems characterized by high sensor data-rates and limited processing abilities. Commonly used noise filters largely depend on the window (or neighborhood) size, which makes the mine detection ineffective. Using the bi-dimensional empirical mode decomposition (BEMD) analysis, an effective, robust sea mine detection system can be created. A family of decomposed images is generated and applied to optical lidar image data from the Rapid, Overt, Airborne, Reconnaissance (ROAR) experiment supplied by Naval Surface Warfare Center, Panama City. These decompositions project key image features, geometrically defined structures with orientations, and localized information into distinct orthogonal components or feature subspaces of the image. Application of the BEMD method to the analysis on side scan sonar data is also provided.

Accurate detection and classification of mines is time consuming and requires divers or Autonomous Underwater Vehicles (AUV) in the water. The navy continues to pursue more expedient methods in mine countermeasures, and with airborne lidar, a surf zone (SZ) and landing zone can be quickly surveyed for possible mines. In the near surf zone, all possible mines can be quickly neutralized by dropping guided munitions, eliminating the need to send divers or AUVs to verify contacts. Still, the need for improved methods of detection and classification is needed. BEMD, a relatively new method of signal analysis developed about fifteen years ago, was tested on lidar imagery from the ROAR experiment to look for any improvements in detecting and classifying mines.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ALMDS	Airborne Laser Mine Detection System
AUV	Autonomous Underwater Vehicle
BMED	Bi-dimensional Empirical mode decomposition
COBRA	Coastal Battlefield Reconnaissance and Analysis
COMINEWARCOM	Commander, Mine Warfare Command
CMSS	Compact Modular Sensor and Processing Suite
DBT	Doctrinal Bottom Type
EMD	Empirical mode decomposition
EPMA	Environmental Post-Mission Analysis
ESLRF	Eye-Safe Laser Range-Finder
IED	Improvised explosive device
IMF	Intrinsic mode function
ISDT	Integrated scanner, detector, and telescope
LIDAR	Light detection and ranging
LCS	Littoral Combat Ship
MCM	Mine Countermeasures
MIW	Mine Warfare
MEDAL	Mine Warfare Environmental Decision Aid Library
NAVO	Naval Oceanographic Office
NRL	Naval Research Laboratory
NOMBO	Non-Mine Mine-like object
OMCM	Organic Mine Countermeasures Future Naval Capabilities
ONR	Office of Naval Research
PFM	Pratt's Figure of Merit
ROAR	Rapid, Overt, Airborne, Reconnaissance
ROIC	Read out integrated circuit
S/MSE	Signal to mean square error
SNR	Signal to noise ratio
SONAR	Sound navigation and ranging
SZ	Surf Zone
TCS	Tactical Control Software
TEGRWS	Tactical Exploitation Group Remote Work Station
TUAV	Tactical UAV
UAV	Unmanned Aerial Vehicle
UUV	Unmanned Underwater Vehicles
VSW	Very Shallow Water
WBIED	Waterborne Improvised Explosive Device

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

My deepest gratitude goes to Professor Peter Chu at NPS for his patience, unselfishness, and abiding support. I would also like to thank Mr. Ron Betsch of NAVOCEANO for being a second set of eyes on this thesis. To Chris Coward of NAVOCEANO, I thank you for being so helpful with EPMA to conduct this thesis. I would especially like to thank Mr. Chenwu Fan for sharing his exceptional programming knowledge in MATLAB.

Also, I must thank God for making this possible. And I would like to thank my parents for their love and support.

THIS PAGE INTENTIONALLY LEFT BLANK

I. MINE WARFARE IN NAVAL OPERATIONS AND HOMELAND SECURITY

A. BACKGROUND

The naval mine is one of the most cost effective weapons in the naval force and is an efficient force multiplier (Ocean Studies Board 2000). An underwater minefield has the ability to bring a highly advanced and capable navy to a halt and deny access to territorial waters. A delay of days to weeks could provide the enemy with crucial time. In the Korean War, conflict offensive mining by communist forces immobilized U.S. naval amphibious forces for more than a week during the landing at Wonsan (Zwolski 1998). Approximately 3,000 Russian-made contact and magnetic mines caused a fleet of 250 ships to wait off the coast while 10 American minesweepers tried to sweep a clear channel through the minefield (Morison 1995). Two minesweepers were destroyed and sunk, and many were damaged. Minesweeping operations took two weeks rather than the estimated one week, and as a result, the amphibious forces' window of opportunity was gone.

> "We have lost command of the sea to nation without a navy, using weapons that were obsolete in World War I and laid by vessels that were laid at the birth of Jesus Christ." – Rear Admiral Smith, Commander, Amphibious Task Force, Wonsan, Korea, 1950.

The cost in time and money to sweep a minefield is about ten times the cost of the mines. Damage per mine can cost at least 2,000 times as much. The USS Samuel B. Roberts (FFG-58) on escort duty during Operation Desert Storm was nearly sunk due to a mine strike. The ship underwent major repairs, costing almost $90 million from a mine that cost $1,500 (Figure 1a).

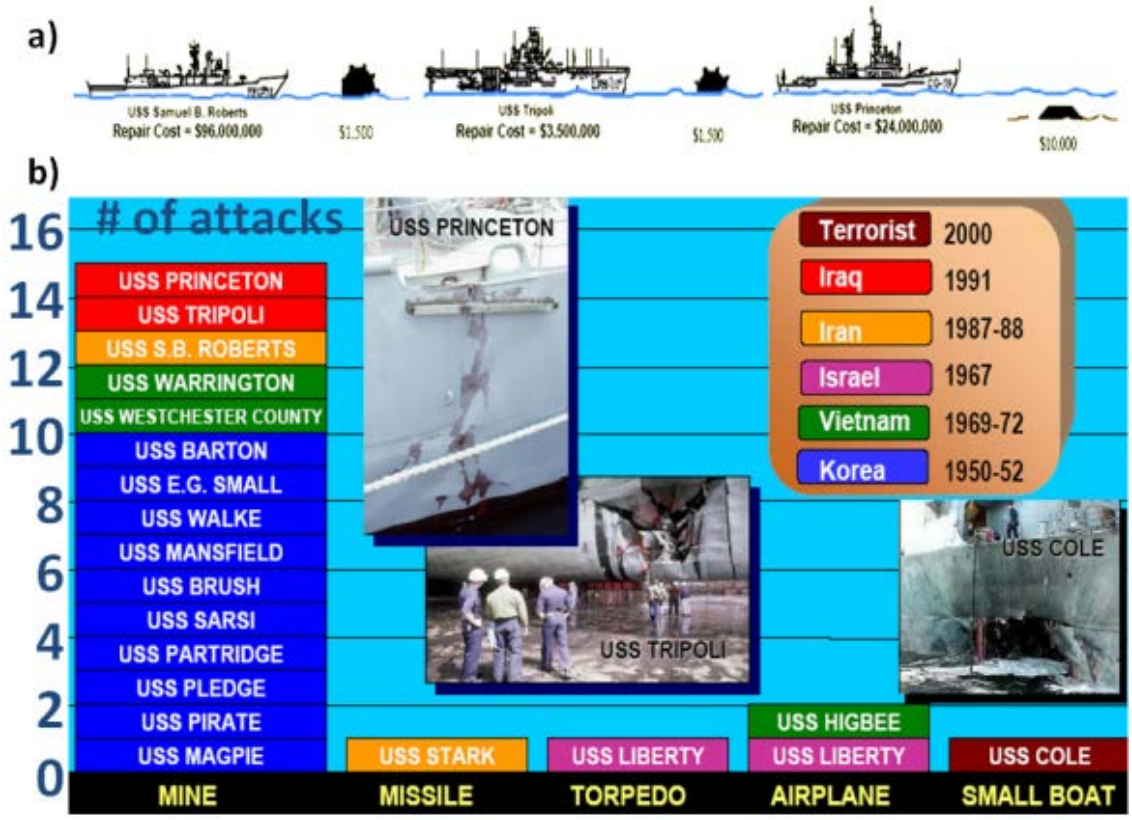


Figure 1. a) Cost of ship damage and cost of mine b) attacks on U.S. Navy vessels since the Korean War (From Avery, 1998).

Even in present times with advanced weapons and technology, the naval mine continues to be a standard and frequently used weapon. Since WWII, mines were the most-used weapon to carry out attacks on U.S. Navy vessels (Figure 1b). The U.S. Navy currently has 14 mine countermeasure (MCM) ships commissioned and deployed throughout the world. These MCM ships are assets dedicated to detecting, classifying, and clearing mines. Organic mine warfare assets, which are combatant ships that are equipped with some mine countermeasure capability, due to MCM ships unable to be everywhere at all times. Transit of fleet assets into foreign waters or any littoral waters requires the surveying and clearing of mines, which depending on the density of the minefield and the number of minesweeping assets, can cause a significant delay of operations (Figure 2) as shown in the parabolic relationship, a dense minefield can take up to 160 days.

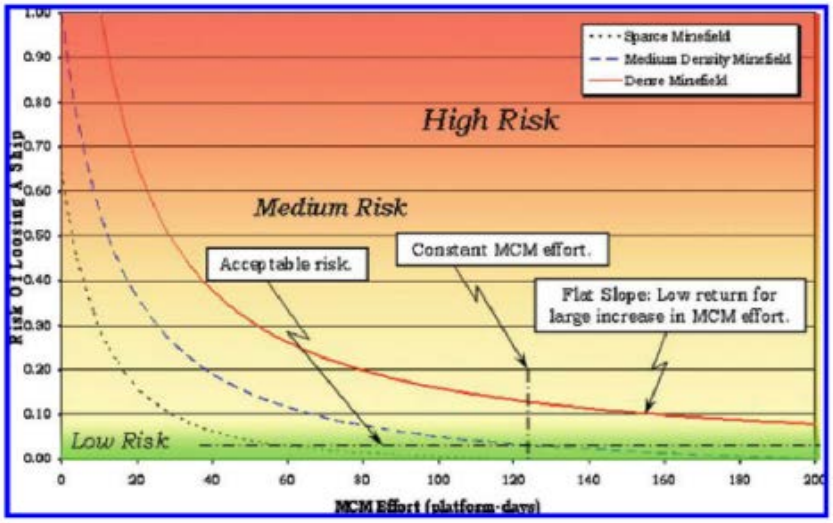


Figure 2. The relationship of the risk of losing a ship based on the MCM effort in days is parabolic (After Holmes, 2006).

This time is increased as the number of mine-like objects increases, as it requires minesweeping assets on location and investigation, then at which the objects is detonated or cleared as non-threat. Of course, the best mine warfare (MIW) strategy is early interdiction and prevention of deployment, which is difficult because mines can be quickly and clandestinely deployed. The capability exists for searching and hunting of mines, but it is very time consuming.

B. CURRENT MINE WARFARE THREAT

With much of the navy's efforts pushing to the littoral region, the timely and accurate detection of mines is even more important. The water column is shallower and more influenced by the environment which causes degradation of sensor performance.

> "The tactical advantage will probably depend not on who has the most expensive, sophisticated platforms – but rather on who can most fully exploit the natural advantages gained by a thorough understanding of the physical environment." – Rear Admiral W.G. "Jerry" Ellis, U.S. Navy, Oceanographer of the Navy, 1999

Mines are not just a threat to our naval forces in foreign waters. Mines can be laid in our own waters, which would threaten not just naval and commercial vessels, but would threaten trade and economy. The patriotic sea diver incident in January 1980 was a

terrorist threat incident that claimed that mines were laid in the Sacramento River, causing all shipping movement to stop. The threat caused a four-day halt to shipping while the USS Gallant (MSO-489) intensely conducted mine hunting operations. The impact to merchant shipping was estimated to be around in the hundreds of thousands of dollars. Threats to major ports and waterways would cause even more significant economic impact. In 2005, more than $700 billion of goods passed through our nation's ports.

April 2004, Lake Ponchartrain, Louisiana: A suspicious floating bag was spotted by a tugboat operator and he notified the Coast Guard. It turned out to be a waterborne improvised explosive device (WBIED) with a few pounds of explosives, encased in pipes with a timer. A dignitary was scheduled to take a campaign trip on the lake. With the success of the improvised explosive device (IED) in recent conflicts, and the ease in which they can be made and deployed, improvised waterborne mines are a very credible threat. Our maritime security program now involves frequent port surveys and more law enforcement involvement.

II. MINE DETECTION

A. DETECTION OF OBJECT VS. ENVIRONMENT

Successful detection of a mine depends on the complexity of the environment that it lies in: seabed features, natural and man-made, and clarity of the water. Flat ocean bottom environments are much more favorable for mine detection; rough and rocky bottom types make detection much more difficult and more time consuming. Some bottom types are highly susceptible to change due to environmental effects. Sandy bottom types can be smooth one day and be rough on another day as ocean currents cause ripples on the bottom. Therefore, the seabed in the littoral environment is highly complex and is a result of many environmental processes and human activities. The mine that is most difficult to detect is one that is at the bottom and buried, intentionally or naturally. The bottom of the ocean is affected by waves and other ocean processes, and can shoal and cover up bottom mines. Sediments can be transported by tides and currents. A re-survey of the same location can be quite different as the seabed terrain might have changed and other sediments which may be transported in. The characteristic of the seabed for U.S. Navy MCM is described in doctrinal bottom type (DBT) values, which is determined by roughness, mine burial amount, and bottom composition.

1. Clutter

Clutter is categorized by density (or amount per square nautical mile) of non-mine, mine-like bottom objects (NOMBO). A NOMBO has a high signal return in contrast to its environment just like mines, and requires divers in the water or survey equipment to verify objects as mine or non-mine. Clutter can be natural objects such as rocks or coral, or man-made objects. They can be put into the waters as ships dispose trash or large objects, or be brought in from a major storm or flood. Areas that are denser in shipping traffic are more susceptible to clutter. According to Naval Warfare Publication 3.15–41, clutter has three categories as described in Table 1. Clutter category 1 is described as low clutter, and has less than 15 NOMBOs per square nautical mile. Clutter Category 2, is

described as medium clutter, and has with 15 to 40 NOMBOs per square nautical mile. Clutter Category 3, is described as high clutter, and has over 40 NOMBOs per square nautical mile.

CLUTTER		
Category	Description	NOMBO/sq nm
1	Low	< 15
2	Medium	
3	High	> 40

Table 1. Clutter categories from NWP 3.15–41

2. Roughness

Strong bottom currents in sandy littoral regions can cause the surface of that seafloor to change into a ridged texture, and the ridges can cause shadow zones. The height of the ridge above the seafloor is what determines the category of bottom roughness, as described in NWP 3.15–41, and is a measure of difficulty of mine detection. Shadow zones may cause contacts lying behind the ridge to be missed determined by the height of the ridge from the seabed. Higher roughness will cause larger shadow zones and can cause more mines to be hidden in the shadow zones behind these ridges which would require more survey time to cover the shadow zones. Roughness categories from NWP 3.15–41 is described in Table 2. Roughness Category 1, described as smooth, rises less than 0.2 meters above the seafloor. Roughness Category 2, described as moderate, rises between 0.2 to 0.3 meters above the seafloor. Roughness Category 3, described as rough, rises more than 0.3 meters above the seafloor.

ROUGHNESS			
Category	%	Description	HEIGHT(M)
1	< 5	Smooth	< 0.2
2	5 to 15	Moderate	
3	> 15	Rough	> 0.3

Table 2. Roughness categories from NWP 3.15–41

3. Burial

Burial is the predicted amount burial of a mine due to the sediment moving around and settling back down. The type and thickness of the bottom sediment, the strength of the ocean dynamics, whether it is waves or ocean currents, play a factor in the chance of mine burial. Planning tools like Mine Warfare Environmental Decision Aid Library (MEDAL) combine bottom type and thickness information from databases and oceanographic parameters to derive burial amount. NWP 3.15–41 describes in Table 3 burial Category 1 as no chance of burial, category 2 as 0 to 10 percent burial, Category 3 as 10 to 20 percent burial, Category 4 as 20 to 75 percent burial, and Category 5 as 75 to 100 percent burial. Complete burial would make the mine undetectable from a side-scan or bottom mapping sonar, and would require bottom profiling sonar survey which can penetrate into the seabed.

BURIAL		
BOTTOM TYPE	NUMBER	PERCENT BURIAL
Rock	1	0%
Mud or Sand	2	
Mud or Sand	3	
Mud or Sand	4	
Mud or Sand	5	

Table 3. Burial categories from NWP 3.15–41

MEDAL offers up to date information on DBT as well as decision aids which allow for planning of MIW operations. As the environment becomes more complicated, detection becomes more difficult, therefore knowledge of the environment is essential in determining the time required to clear an area of mines. Clarity of the water is important as well for successful and timely mine detection. Noise is an always present factor in environmental data which must be dealt with. In lidar imagery sediments and sea surface roughness adds to noise and reduced sensor performance. Resolution is also important to mine detection, with higher resolution, mine shape and features are more distinguishable. Rough seas will also make matters difficult in terms of safety and accuracy of data.

B. DOCTRINAL BOTTOM TYPE

DBT values is what the operators use as their planning tool for MCM operations. The DBT values in Table 4 will give them a rough estimate as to the complexity of MCM operations. Category A is characterized by a sandy or muddy bottom with 0 to 10 percent burial, and smooth bottom surface. This is the easiest bottom environment in which to detect mines. Classification values increase as burial percentage becomes greater and as roughness becomes greater. The most difficult is category D, which is characterized by 75 to 100 percent burial, indicates the most difficult bottom type.

Knowledge of the environment in advance is important to successful MCM and survey operations. As DBT category may provide the operator an idea of difficulty and time required to complete the mission, knowledge of the ocean bottom characteristics like bottom composition, predicted burial amount, bottom roughness, clutter, ocean and environmental conditions also provide valuable information as to performance of sensors.

Bottom Composition	Predicted Mine Case Burial	Bottom Profile Group	Bottom Category
Mud or Sand	0 – 10	SMOOTH	A
		MODERATE	B
		ROUGH	C
	10 – 20	SMOOTH	A
		MODERATE	B
		ROUGH	C
	20 – 75	SMOOTH	A
		MODERATE	B
		ROUGH	C
	75 – 100	ALL	D
Rock	0	SMOOTH	A
		MODERATE	B
		ROUGH	C

Table 4. Doctrinal bottom type categories from NWP 3.15–41

C. CHANGE DETECTION ANALYSIS

Change detection analysis is conducted as follows. A follow-on survey and prior survey of the same area are compared. If they match exactly, and all objects in the previous survey were cleared as a non-threat, then no new threats are present. If there are a few locations that are different, then there is a potential that these locations may have mines present. All the changed areas must be investigated, but the unchanged locations need not be investigated due to previous survey and it reduces time to classify these objects.

D. SENSOR SYSTEMS

1. Klein 5000

The Klein 5000 side scan sonar was developed by L3 Communications for military and commercial use, and is approximately 2 meters long and weighs 155 lbs. The Klein 5000 uses multibeams at a frequency of 455 KHz to map the seafloor. The newest Klein allows for faster tow speeds of up to 10 knots (L3 Communications, Klein Associates, Inc. 2009). This enables the survey vessels to spend less time at sea surveying which in turn saves money. The Klein 5000 is what NAVO ships are using for port surveys, whose data is input into EPMA to output bottom type information.

2. NAVO ODEC BATHY-2000

This bottom and sub-bottom profiler sonar is versatile and can penetrate into the ground to obtain sediment type and thickness. The acoustic sensor operates at 2 kHz–6 kHz and up to 23 dB and is towed under or behind a research vessel. It can operate as deep as 11,000 meters and can profile the bottom as deep as 100–200 meters. Other uses are for hazard and mining surveys, installation and maintenance of pipes and cables, dredging and geophysical surveys. Sediment classification data is used to make burial predictions.

3. AN/SQQ-32

The AN/SQQ-32 was developed and designed by Raytheon and is used onboard MCM class ships. This highly capable sonar allows MCM ships to detect and classify

objects both moored and at the bottom (Figure 3). Sensors onboard the minesweeper MCM class ships are used strictly for operational detection and not for bottom mapping. The data is not included in Naval Oceanographic Office's (NAVO) MIW database.

Figure 3. AN/SQQ-32

4. AIRBORNE LASER MINE DETECTION SYSTEM (ALMDS)

ALMDS is a laser emitting MIW sensor and was recently developed to be fitted on the MH-60S helicopter to detect, classify, and localize floating and near-surface moored mines. This system is a part of the overall Littoral Combat Ship (LCS) MIW package designed for quick assessment of mine threats in littoral waters, choke points and confined regions, and amphibious landing areas. The ALMDS pod is fixed onto the MH-60S via a bomb rack (Figure 4) and is able to operate at high speeds for high search rates, and can operate day and night.

Lidar began with applications to topographic mapping of surfaces above water, and then it evolved with applications to underwater mapping of channels for bathymetry. Now it is applied to operational use for detection of mines in the upper layer of the oceans. Lidar is limited only to the upper layers of the water. Another MIW sensor called the AQS-20 is a sonar sensor that is towed from a MH-60S to detect mines in the lower layers of the ocean and allows for complete coverage of the water column (Figure 4).

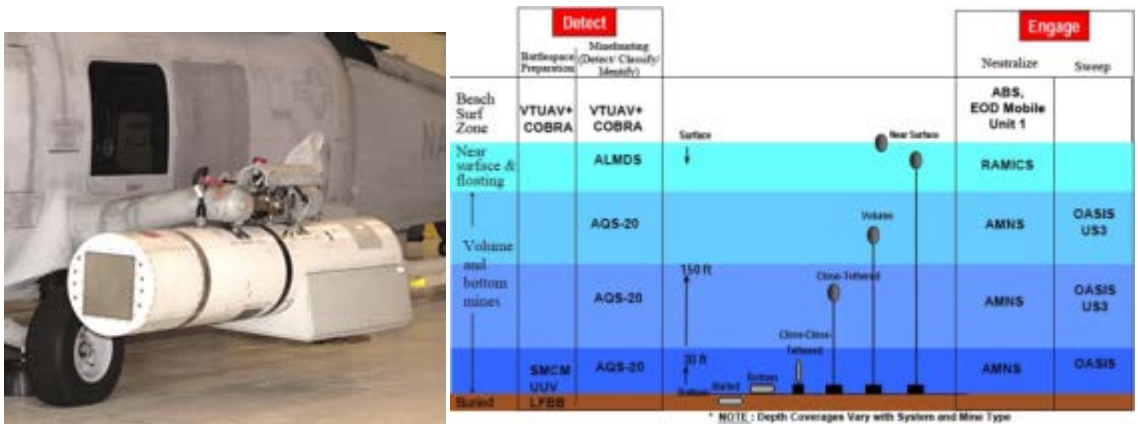


Figure 4. LCS MIW package. b) ALMDS covers the surface waters to detect near surface and floating mines.

E. ENVIRONMENTAL POST-MISSION ANALYSIS SYSTEM

Scientists at the Naval Research Laboratory (NRL) and NAVO developed the Environmental Post-Mission Analysis (EPMA) system for the MIW community to ensure data visible, available, and usable for war fighter, when needed and where needed, to accelerate decision-making. The system ingests and processes field data and formats it into products that can be used in analysis too land prediction models to allow informed and timely decision-making (Lin et al. 2009).

The EPMA system has the capability to process environmental data with different formats such as converting raw side-scan sonar imagery (in proprietary format) to a common processing format, and then exporting it as a geo-rectified TIFF image. Each of these manifestations of the data has a specific purpose. To manage and integrate disparate data outputs, EPMA defines a unified, extensible data type (i.e., EPMA data type) system to help discriminate into what processes a data set can be fed.

In this study, EPMA is used to produce bottom type information from the raw data collected from sensors. DBT is a bottom type classification primarily used by the MIW operators for operational decisions for MCM missions. Further detailed information of seabed characteristics is also kept in a database at NAVO.

F. BACKGROUND AND PROBLEMS

The UAV is the key technology in reducing the dangerous work of surveying and clearing underwater mine fields. The ONR has the spiral development of the Coastal Battlefield Reconnaissance and Analysis (COBRA) system (Almquist 2009): Block-1 (day-time operations, surface mines and obstacles, and detection in beach zones), Block-2 (night operations, detection in the SZ), and Block-3 (buried mines, near real-time processing). The ROAR airborne 3D lidar system is the major component for COBRA Block-2. Technical UAV sensor for detecting minefields in the beach and SZ is the key component in Block-3. Without automated detection and classification algorithms, all image data collected in all the three COBRA blocks must be reviewed by a human operator prior to marking and removing mine-like objects.

The COBRA system will allow naval expeditionary forces to conduct airborne standoff reconnaissance and automatic detection of minefields in the SZ and inland. COBRA will consist of three primary components—the COBRA Airborne Payload, the COBRA Processing Station, and the Tactical Control Software (TCS). The COBRA Airborne Payload will consist of a multi-spectral sensor system that will be placed on a UAV to conduct reconnaissance, detect minefields, obstacles, and camouflaged defenses. The Tactical Control Software that is loaded onto the UAV Ground Control Station will control the COBRA Airborne Payload. Analysis of the imagery collected by the COBRA Airborne Payload will be conducted at the COBRA Processing Station. The COBRA Processing Station includes a Tactical Exploitation Group Remote Work Station (TEG RWS) with enhanced algorithm processing.

In COBRA block-2, the ROAR system will be developed to use an innovative Lite Cycles Incorporated (LCI) proprietary integrated scanner, detector, and telescope (ISDT) receiver architecture. The ISDT tightly couples all receiver components and lidar electronics to achieve the system compaction required for tactical UAV integration, while providing a large receiver aperture and a programmable scanning function for wide area search with temporally displaced multiple looks on the fly for clutter reduction (Figure 5). The ISDT incorporates a 128 x 128 3D camera using Readout Integrated Circuit (ROIC) technology developed by an LCI teammate, an LCI-developed large-array range-gated

ICMOS camera, and a precision rangefinder that incorporates technology from LCI's eye-safe laser range-finder (ESLRF) product line. New processing algorithms for mine detection in the very challenging SZ clutter environment, which offer the potential for significant processing gains in comparison to the legacy approaches, are under development.

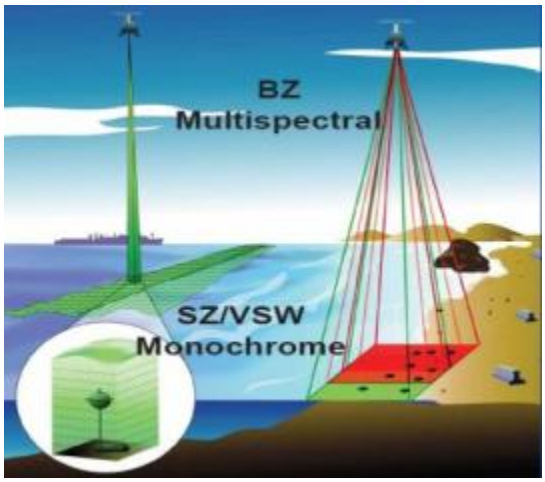


Figure 5. UAV mounted lidar search area

G. IMPROVEMENTS ON CURRENT CAPABILITEIS

The Office of Naval Research (ONR) is looking to improve detection, localization, and classification, and drift prediction of surface and near-surface drifting mines. Specifically they are making efforts to improve the Compact Modular Sensor and Processing Suite (CMSS) integrated into a tactical unmanned airborne vehicle (TUAV), which will also be able to provide in-situ information about the environment to predict mine drift trajectories. Also there is continued effort in the development of target recognition algorithms to provide detection, classification, and localization of near-surface mines. Another effort is to have sensors configured to record environmental conditions such as surface waves, currents, optical properties of the water column, and more.

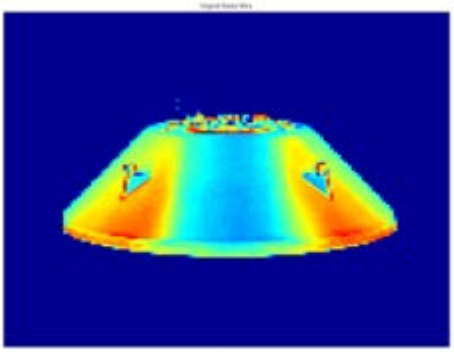
THIS PAGE INTENTIONALLY LEFT BLANK

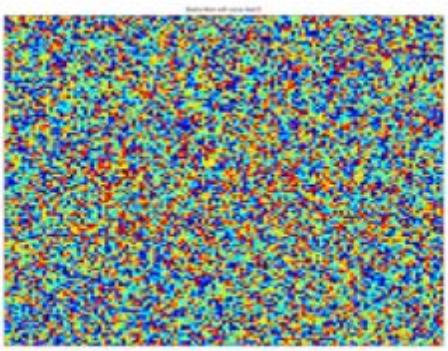
III. CURRENT NOISE FILTERING METHODS

Detection of sea mines in coastal regions is difficult due to the highly variable sea-bottom conditions with uncertain underwater environment. Detection systems must be able to discriminate objects that vary in size, shape, and orientation from naturally occurring and man-made clutter. Usually, noise filtering technique is needed to locate the mines in various images from side-scan sonar, forward-looking sonar, lidar, and others. Linear and non-linear filtering techniques are available. Usually, the linear filtering is conducted in the frequency domain; and the non-linear filtering is conducted in the spatial domain.

The theoretical base of the linear filter methods (e.g., ideal filter, and Wiener filter) is the Fourier transform. Frequency domain is space defined by the values of Fourier transform and its frequency variables. Let $x(i, j)$ represent an input image with (i, j) the pixel location, and size $M \times N$ $(i.e., i = 1, 2, \ldots, M; j = 1, 2, \ldots, N)$. $X(u, v)$ be its discrete Fourier transform with (u, v) the frequencies. The linear filtering is represented by the multiplication

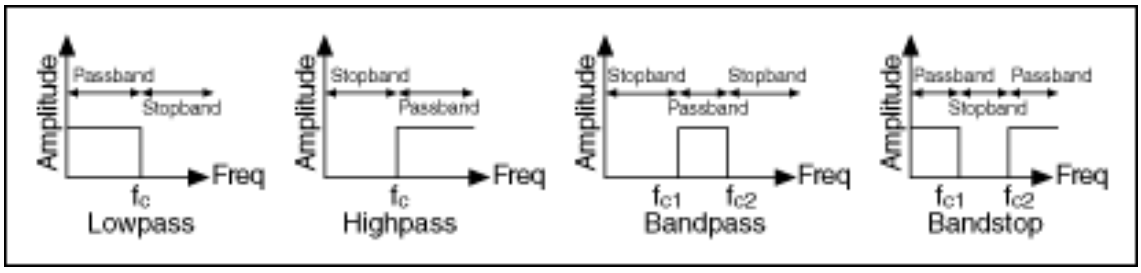
signal processing methods have their origin in statistics. In fact, the median filter was first introduced in statistics for smoothing economical time series. It soon became evident that the median filter performs very well especially in image processing applications where sharp transitions are common. Another non-linear filtering approach that has received considerable attention, and for which much theoretical study has been conducted, is that of so-called rank-order filtering, a method whose filtering effect is obtained by rank-ordering the input data. Rank-order filtering is also known as order-statistics filtering. The non-linear filtering is conducted in the spatial domain, i.e., it directly operates on the pixels,





A. IDEAL FILTER

Ideal filter is a linear filter with the filter function *H(u,v)* having the following features. The set of frequencies ω = (*u*, *v*) such that



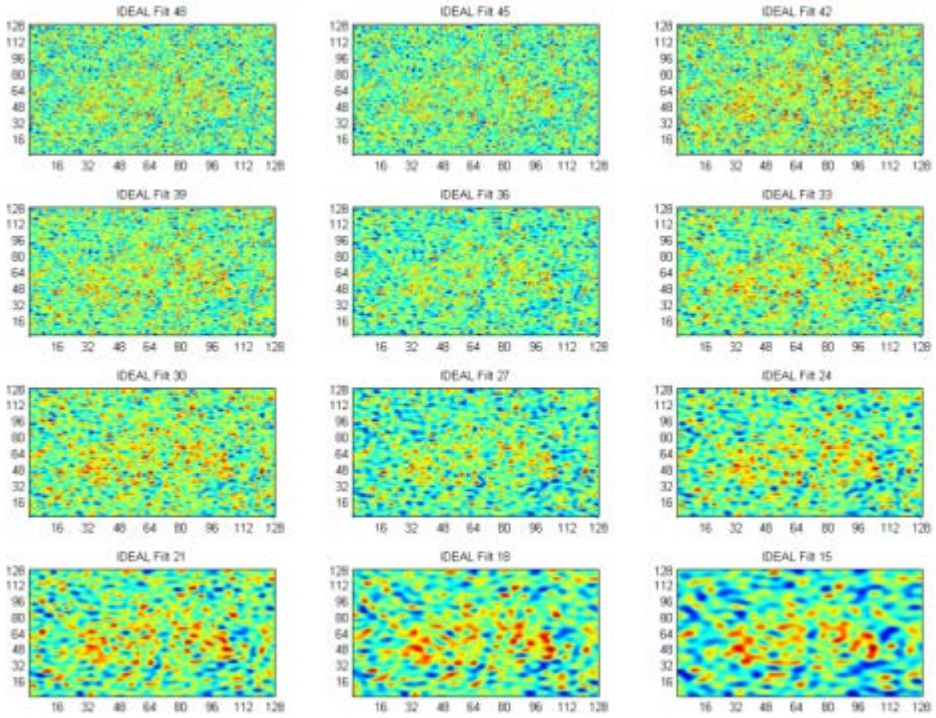


Figure 8. Noise filtration on Fig. 6b using the ideal filter (3) with large threshold frequencies ($f_c > 12$). The noises filtration depends on the threshold frequency.

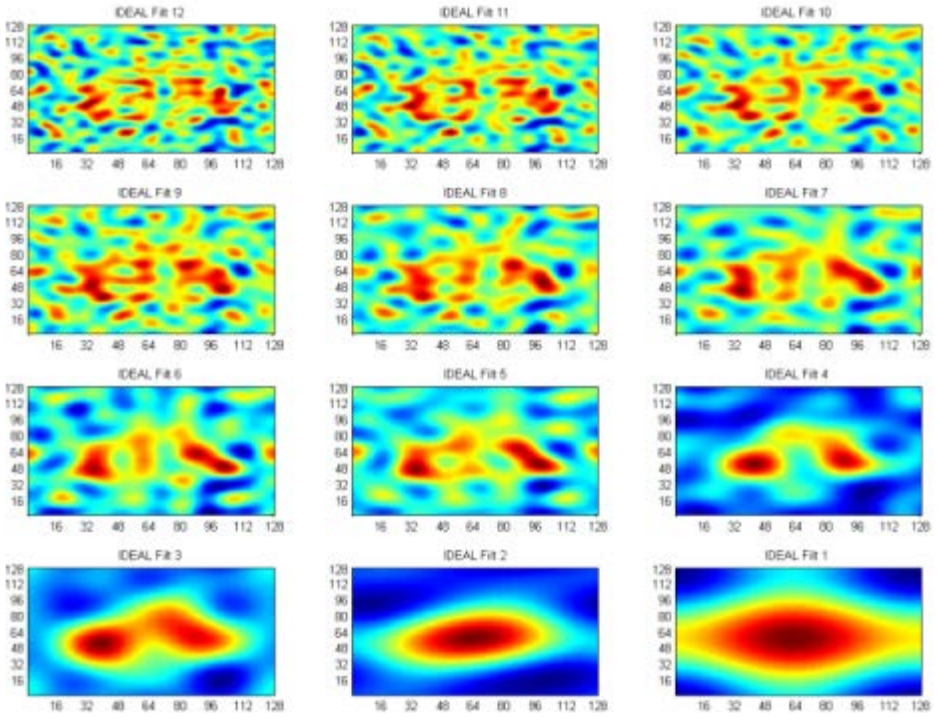


Figure 9. Noise filtration on Fig. 6b using the ideal filter (3) for small threshold frequencies ($f_c \leq 12$). The noise filtration depends on the threshold frequency.

B. WIENER FILTER

The Wiener filter is the least square solution to the problem of signal recovery in the presence of noise. Use of the least square error leads to the filter function $H(u, v)$ of the Wiener filter

signal power spectrum is a little more challenging in principal. To get the signal-to-noise ratio

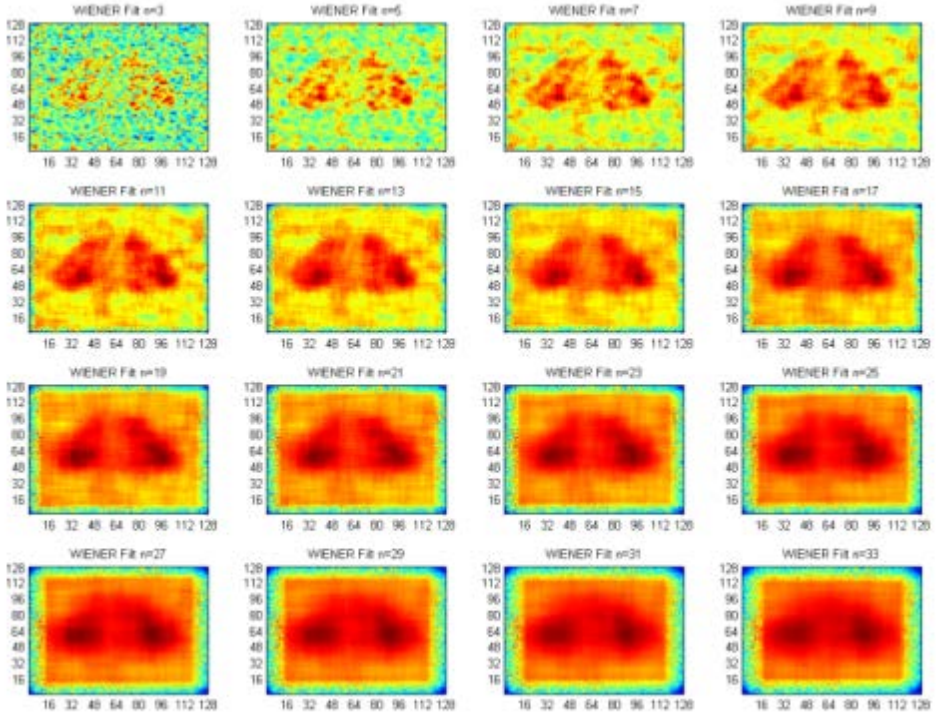


Figure 10. Noise filtration using the Wiener filter (4) with various odd-numbers of neighborhood size *(n*n)* for the estimation of the signal-to-noise ratios. It is noted that the noise filtration largely depends on the value of *n*.

C. MEDIAN FILTER

The median filter is a non-linear filter and more effective than convolution when the goal is to simultaneously reduce noise and preserve edges. Neighborhood averaging can suppress isolated out-of-range noise, but the side effect is that it also blurs sudden changes (corresponding to high spatial frequencies) such as sharp edges. The median filter is an effective method that can suppress isolated noise without blurring sharp edges. Specifically, the median filter replaces a pixel by the median of all pixels in the neighborhood:

rectangle, or a cross. The pixel at the center will be replaced by the median of all pixel values inside the neighborhood. The commonly used neighborhood is a square with size *n* * *n* (*n* is an odd number). Large values of *n* are comparable to low-pass filters. Figure 11 shows the performance of the median filter (6) on the noisy image (shown in Figure 6b) with *n* values varying from 3 to 33. The median filter at *n=3* shows slight signal improvement through the noise, but still remains noisy. At *n=11* two cone shapes come together in a triangular shape correlating to the manta mine in the original image. Window sizes larger than *n=11* result in a decreased contrast of object and background. Although the median filter is able to effectively filter out noise and separate the object against the background, it results in less contrast of the mine shape from the background compared to the previous Wiener filter.

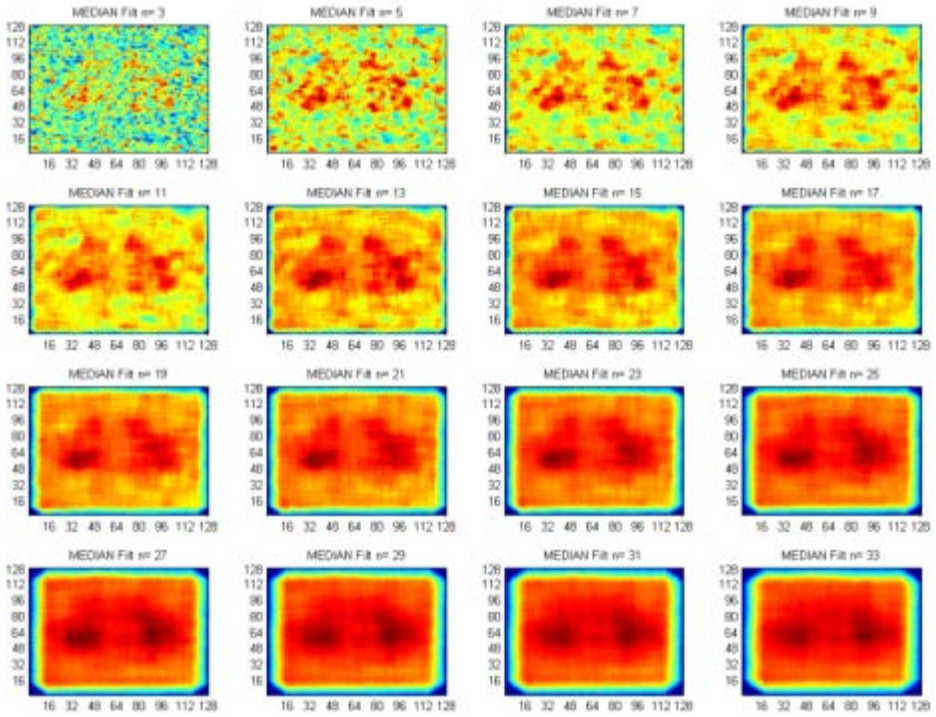


Figure 11. Noise filtration on Fig. 6b using the median filter (6) with various odd-number of neighborhood size (*n*n*). It is noted that the noise filtration largely depends on the value of *n*.

D. ORDER STATISTIC FILTER

The order statistic filter is most common form of image filtering. It replaces a pixel by statistical features of all pixels in the neighborhood centered at that pixel. To use this type of filters, the noisy image is divided into blocks of pixels called windows, comprising $n * n$ pixels, where n is an odd integer greater than 2. Appropriate statistical methods are then applied to the window to remove noise. These techniques are based on order statistics as the spatial values of every window are ranked and the median/mean of the ranked set is given as the output for further processing. Similar to the Wiener filter and median filter, the central pixel $x[i, j]$ of neighborhood size $n * n$ is assumed in the order statistic filter to be a corrupted pixel and is treated for noise removal. In that neighborhood, only $K= (n^2- 1)$ pixels, excluding the central pixel $x[i, j],$ are processed to find are placement for $x[i, j],$ The two-dimensional sequence of pixels is changed into one-dimensional, by taking the pixels left-to-right and top-to-bottom,

with the mine object very slightly distinguishable. At *n=9* the triangular mine shape becomes apparent with the higher valued cone shapes becoming better defined at *n=13.* Higher window sizes results in further filtering of noise in the background up until *n=17.* Further filtering causes the contrast of the object to the background to be stronger, but the shape of the mine becomes more oblong and rounded and so further filtering beyond *n=17* is not considered beneficial.

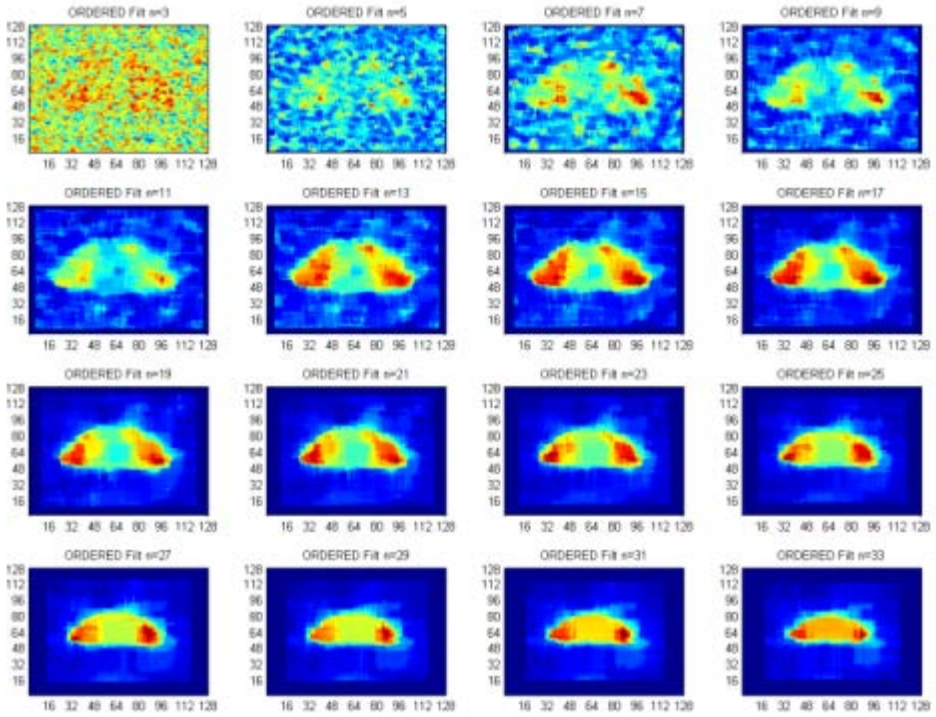


Figure 12. Noise filtration on Fig. 6b using the order statistic filter (10) with various neighborhood sizes (*n*n*) 3 to 33. It is noted that the noise filtration largely depends on the value of *n*.

E. WEAKNESS IN THE EXISITING NOISE FILTERS

As described in this chapter, all the linear filters need window size or frequency threshold; and all the non-linear filters need neighborhood size. The noise filtration largely depends on these sizes.

Noise filters can effectively reduce noise to the point that objects can be detectable, but knowing when filtering is sufficient enough to separate object from background is difficult. Determining the effective threshold depends on the object shape, size, and signal strength, resolution of the image, noise type (random Gaussian, linear, non-linear), and background features. The operator's ability to distinguish the object from a filtered image is important, as some automated detection systems are designed to model detection methods performed by people, which always outperforms automated systems in terms of detection and false alarms.

Effective noise filtering is needed and knowledge of the environment would allow operators to know how much filtering is required. Knowledge of sensor placement and height above the seabed gives us an idea of image resolution. Forecast winds, ocean roughness, and scattering objects in the ocean can be parameterized into a noise level that is to be expected in imagery. This can narrow the guess as to the window size and frequency threshold needed to effectively filter. Instead of producing eight filtered images with different thresholds, this can be reduced to half the amount.

Lower computational time for mine detection algorithms and filters is the main goal. Many algorithms and filters exist and are used in various combinations to improve mine detection.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. EMPIRICAL MODE DECOMPOSITION

In signal processing, filtering and decomposition of a signal has limitations when dealing with real world data and signals. Real life natural processes are never linear or stationary. Fourier methods apply well to stationary data, but it does not suffice in the decomposition of a signal that is non-linear and non-stationary, as such in the littoral environment which is affected by many non-linear physical processes.

A. EMPIRICAL MODE DECOMPOSITION

Empirical mode decomposition (EMD) is a method of signal decomposition designed for application to non-linear and non-stationary signals. And although it is relatively new and lacks a firm theoretical foundation, it has been successfully applied in many areas in signal analysis which Fourier analysis could not do. EMD is a data driven multi-stepped empirical process of decomposing a signal into various components called intrinsic mode functions (IMF), by using a technique called sifting.

B. PROCESS OF DECOMPOSITION

Huang et al. (1998) developed the EMD method to analyze non-linear and non-stationary data and to filter out noises. EMD is a non-parametric data-driven analysis tool that decomposes non-linear non-stationary signals into Intrinsic Mode Functions (IMF). An IMF is a function that must satisfy two conditions according to the algorithm originally developed: (a) the difference between the number of local extrema and the number of zero-crossings must be zero or one; (b) the running mean value of the envelope defined by the local maxima and the envelope defined by the local minima is zero. The algorithm to decompose a signal into IMFs is then the following (Huang et al., 1998): First, the local minima and maxima of the signal $x(t)$ are identified. Second, the local maxima are connected together by a cubic spline interpolation (other interpolations are also possible), forming an upper envelope $e_{max}(t)$. The same is done for local minima, providing a lower envelope $e_{min}(t)$. Third, the mean of the two envelopes are calculated.

$$m_1(t) = [e_{max}(t) + e_{min}(t)]/2. \quad (12)$$

Such procedure is shown in Figure 13. Fourth, the mean is subtracted from the signal, providing the local detail

$$h_1(t) = x(t) - m_1(t), \tag{13}$$

which is then considered to check if it satisfies the above two conditions to be an IMF. If yes, it is considered as the first IMF and denoted.

$$x_1(t) = h_1(t). \tag{14}$$

It is subtracted from the original signal and the first residual,

$$r_1(t) = x(t) - x_1(t) , \tag{15}$$

is taken as the new series in step 1. If $h_1(t)$ is not an IMF, a procedure called "sifting process" is applied as many times as necessary to obtain an IMF. In the sifting process, $h_1(t)$ is considered as the new data, and the same procedure applies. The IMFs are orthogonal, or almost orthogonal functions (mutually uncorrelated). This method does not require stationarity of the data and is especially suitable for non-stationary and non-linear time series analysis. An IMF is a function that satisfies the following two conditions: In the whole data set, the number of extrema and the number of zero-crossings must either be equal or differ at most by one.

By construction, the number of extrema decreases when going from one residual to the next; the above algorithm ends when the residual has only one extrema, or is constant, and in this case no more IMF can be extracted; the complete decomposition is then achieved in a finite number of steps. The signal $x(t)$ is finally written as the sum of mode time series $x_i(t)$ and the residual $r_m(t)$:

$$x(t) = \sum_{i=1}^{m} x_i(t) + r_m(t). \tag{16}$$

where $x_1(t)$ has the highest temporal variability and $x_m(t)$ has the lowest variability. The functions

$$c_k(t) = x(t) - \sum_{i=1}^{k} x_i(t), \quad k = 1, 2, ..., m \tag{17}$$

show the filtration of high-frequency variability (or noises) from the signal $x(t)$ with $c_1(t)$ filtering out of $x_1(t)$, $c_2(t)$ filtering out of $x_1(t) + x_2(t)$, …, and $c_m(t)$ is $r_m(t)$.

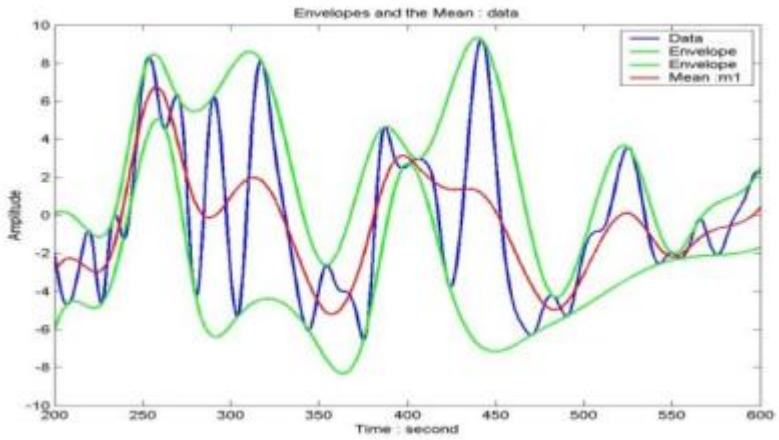


Figure 13. Illustration of the EMD process with the blue curve denoting $x(t)$, the two green curves representing envelopes of local maxima $e_{max}(t)$ and minima $e_{min}(t)$, and the red curve referring to $m_1(t) = [e_{max}(t) + e_{min}(t)]/2$.

C. STOP CRITERIA FOR SIFTING

Although conditions were defined for an IMF, reaching the second condition may require many sifts, and thus could cause a loss in amplitude variation and physical meaning. Depending on the user, or time demands, the first IMF may just require one sift. The stopping condition imposed in Huang's paper [Huang et al. (1989)] is to limit the standard deviation computed from two consecutive results in the sifting process:

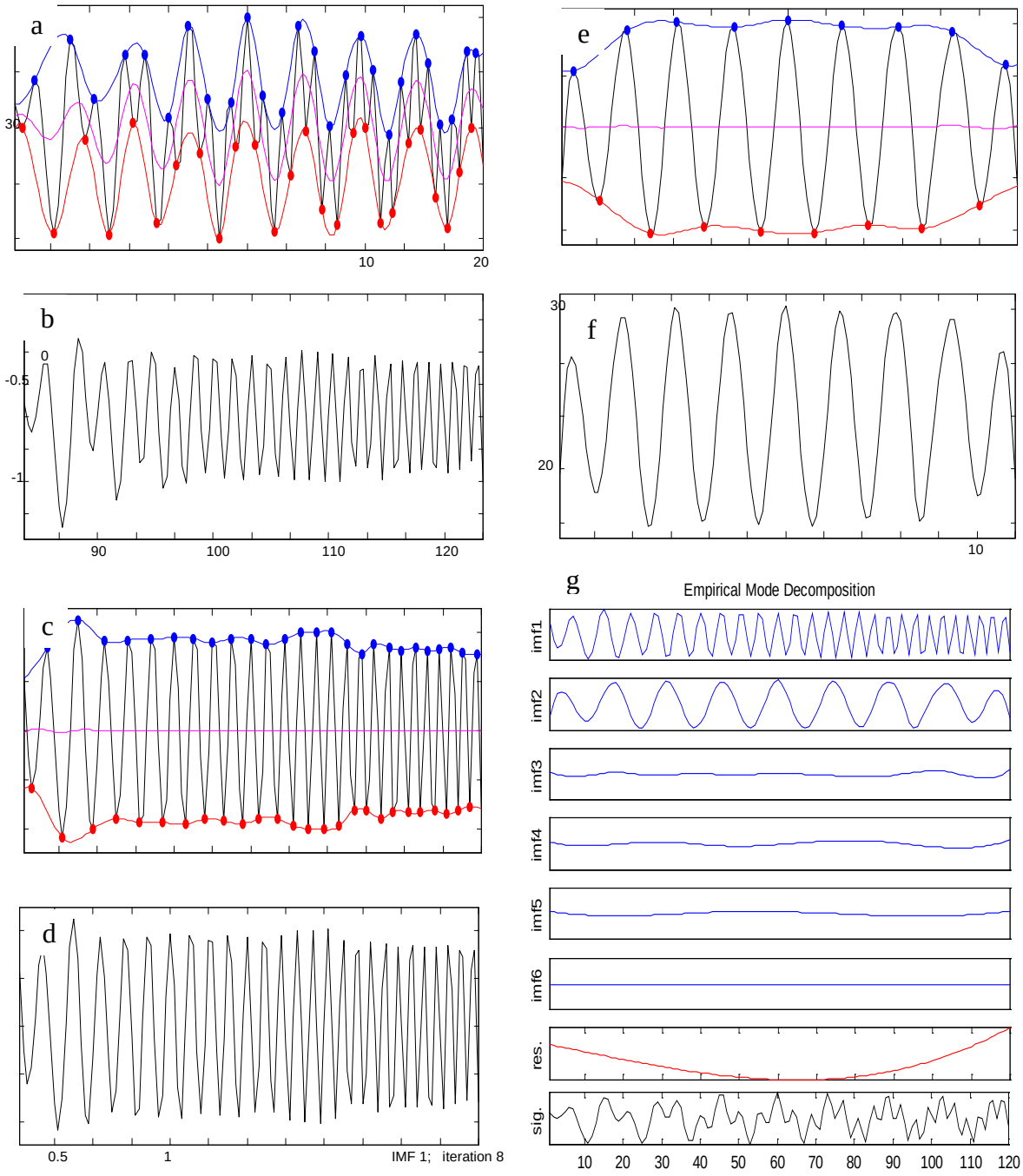


Figure 14. a) Original signal (black), with upper (blue) and lower (red) envelopes, and the mean of the envelopes (magenta). b) Result of the first sift,

D. BEMD FOR NOISE FILTRATION

EMD, which was developed initially for analyze time series applications, has also been applied to a two-dimensional space. A process called BEMD which involves a two-dimensional spatial application of the EMD. Liu and Luo (2010) compared results of the EMD applied filter to other filters applied to an acoustic image. Comparison of the signal to mean square error (S/MSE) and Pratt's Figure of Merit (PFM) clearly showed that the EMD applied filter outperforms the rest in noise suppression. In this case the object is an airplane underwater and is much larger than a mine. Reduced resolution causes a reduction in filter capability.

BEMD is a process similar to EMD, but is applied in two-dimensional space rather than a time signal. EMD is performed for each row and each column until the mean of the envelopes is within the prescribed SD of 0.2 which result in the criteria for IMF 1. And sifting continues for each row and column until the number of modes designated by the program is met. MATLAB was coded to decompose an image into 4 modes. Application of the BEMD on the noisy image (shown in Figure 6b), is decomposed into various IMFs,

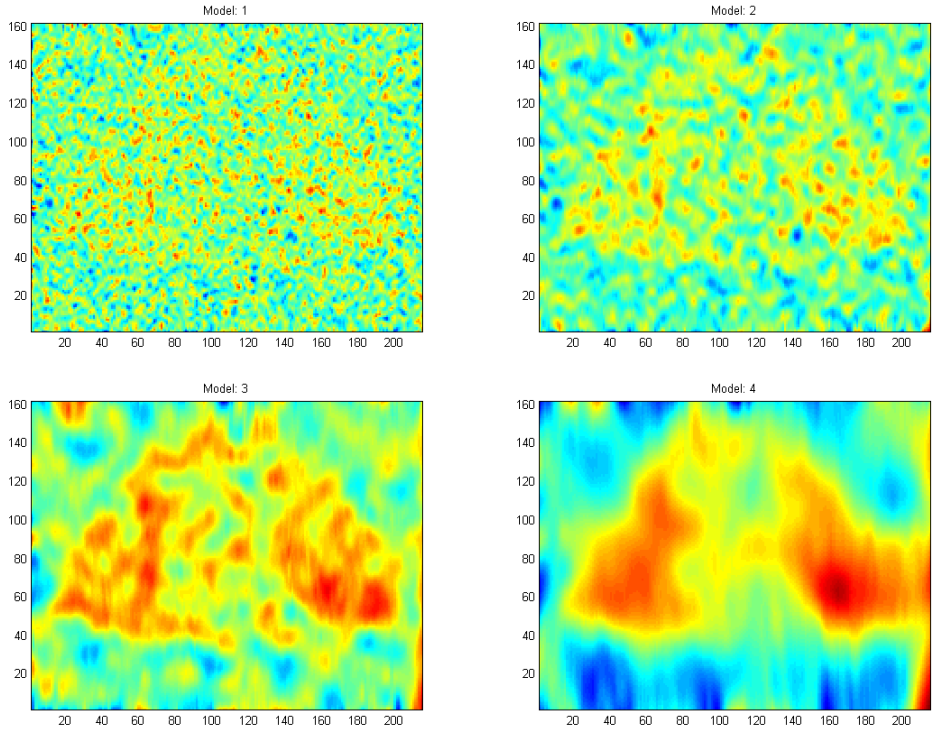


Figure 15. Step-by-step noise filtration on Fig. 6b using BEMD: (a) c_1, (b) c_2, (c) c_3, and (d) c_4. It is noted that this method is very efficient to filter out noises.

V. BEMD FOR NAVY'S RAPID, OVERT, AIRBORNE, RECONNAISSANCE (ROAR)

A. DESCRIPTION OF ROAR

Under the Office of Naval Research's Organic Mine Countermeasures Future Naval Capabilities (OMCM FNC) program, LCI is developing an innovative and highly compact airborne active sensor system for mine and obstacle detection in very shallow water (VSW), through the SZ and onto the beach (Almquist 2009). The system uses an innovative LCI proprietary integrated scanner, detector, and telescope (ISDT) receiver architecture. The ISDT tightly couples all receiver components and lidar electronics to achieve the system compaction required for TUAV integration while providing a large aperture. It also includes an advanced compact multifunction laser transmitter; an industry-first high-resolution, compact 3-D camera, a scanning function for wide area search, and temporally displaced multiple looks on the fly over the ocean surface for clutter reduction. Additionally, the laser will provide time-multiplexed multicolor output to perform day/night multispectral imaging for beach surveillance. New processing algorithms for mine detection in the very challenging surf-zone clutter environment are under development, which offer the potential for significant processing gains in comparison to the legacy approaches.

The objective is to develop an innovative fast, high resolution 3-D flash lidar imager to detect targets in the SZ and VSW from a TUAV. In particular, a high resolution, high dynamic range, fast gate, high integration/sampling rate 3-D Flash lidar imager for underwater imaging

The current 3-D Flash lidar imaging systems have shown the usefulness of 3-D imaging and many lessons have been learned such as the need for narrow gate widths and higher resolution. The ability to detect underwater targets in the SZ with a 3-D volumetric imaging system has been demonstrated. This effort is to develop a 3-D Flash lidar imager to detect targets in the SZ and VSW from a Tactical UAV. This development will require innovations at the chip level to achieve improved pixel count (100,000 pixels or greater), high number of narrow time samples (40 or more), faster gate times (less than

3 nanoseconds), and better signal to noise ratio (SNR). These advancements will provide airborne, ground-based, and sub-surface systems with imaging capabilities beyond those currently available all with reduced cost, size, weight, and power requirements.

The ability to enhance detection of littoral zone targets would also be useful in natural disaster assessment and would also benefit coastal zone surveys, environmental assessments, and search and rescue systems.

B. FT WALTONSURF ZONE EXPERIMENT

1. Survey Environment

ROAR experiment was conducted at Ft. Walton, Florida, around 30.383 N 86.817 W. Lidar survey was conducted along the SZ parallel to the coastline. A shallow sloping seafloor is typical of Gulf Coast waters, yet surf zone is typically rough with significant nearshore processes. This location had a moderate amount of clutter in certain areas of the survey region.

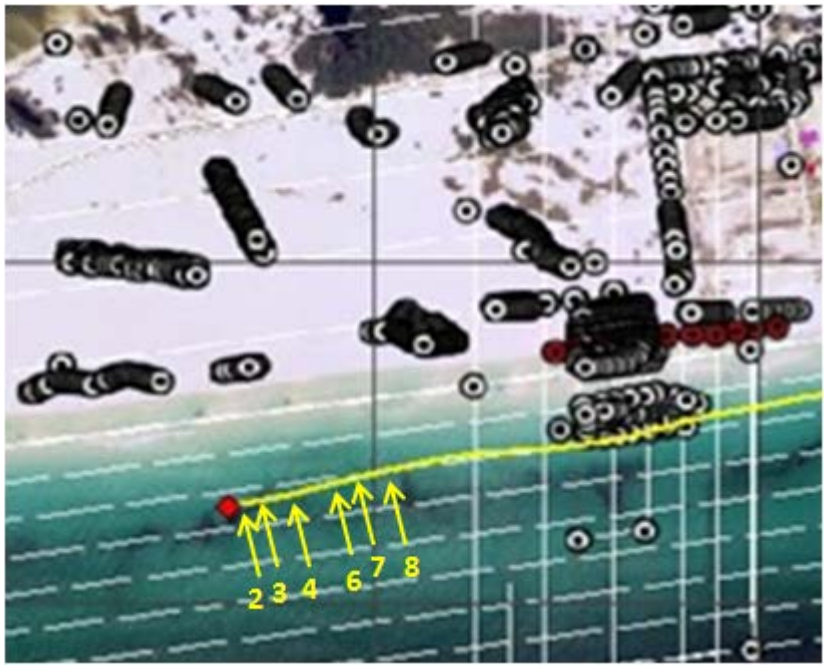


Figure 16. ROAR Survey in the surf zone environment. Sampled images are numbered.

2. Survey Method

Using the Flash lidar imager, an altitude of 3000ft and speed of 70 knots was used throughout the survey. Night and day surveys were completed. Survey track was parallel along the coastline (Figure 16).

3. Survey Data

The surf zone was sampled in cubes encompassed the horizontal and the vertical dimension (Figure 5). Vertical layers were determined by signal return by range gating. Horizontal area was 200 x 200 meters represented by a 128 x 128 pixel horizontal image, with horizontal resolution being 1.5 meters per pixel. The range gate had 44 levels, or frames, which is the lidar pulse returning to the camera in 44 different time sections, representing depth of the water column. Most of the levels were noisy signals unrelated to the seafloor with about only four levels that captured the seafloor. These levels were combined to form one image and then filtered through MATLAB.

The HFT2Dsub function (Appendix B) was set up to filter only one level with TMN = 1, thus only the high wavenumber signals were removed. Further filtering of lower wavenumber signals may have provided better clarity of the object but was not performed due to time constraints which occur in a real operational environment.

4. Findings

After the signal was decomposed, the first IMF was observed to be a relatively noisy signal (Figure 17), with the bottom features only one or two pixels in size. This is taken from the survey track in Figure 16, numbered 3. Higher value returns from the lidar are shown scattered and many in the bottom left corner. IMF 1 and IMF 2 still show an abundant number of contacts on the seafloor. Bottom objects in the IMF 2 image appear larger in size at around 10 pixels, and for the most part correlate to the bottom objects in IMF 1. Moving on into higher modes, the size of the signal becomes larger and thus the signal is representative of larger scale objects, but can also be representative of smaller objects with have strong signal return. Features in the IMF 3 image are even larger scale and the two large contacts in the bottom left corner correlate with the strongest signal

return in the IMF 2 image. Strong signal returns appear in the higher IMF images possibly due to scattering effect. IMF 4 shows one object which stands out against the background and other objects. The residual image can be seen as the overall reflectivity of the seafloor that the objects lie on. The total of the IMF 1 – IMF 4 plus the residual gives us the original image.

In Figure 18, with the first IMF removed the signal of the bottom features are noticeably larger in size and stand out from the background more clearly. The presence of mine-like contacts is clear in the imagery with the first IMF filtered out. The sensor resolution was insufficient to provide clarification on features but does provide enough to determine that there is an object that is possibly mine-like. Appendix A shows before and after images of lidar imagery with IMF 1 filtered out similar to that in Figure18. NOTE: Some of the survey data had consistent impurities which were attributed to problems in the camera sensor. The along track resolution was further degraded due to a high velocity survey speed. Cross track resolution was 1.5 meters.

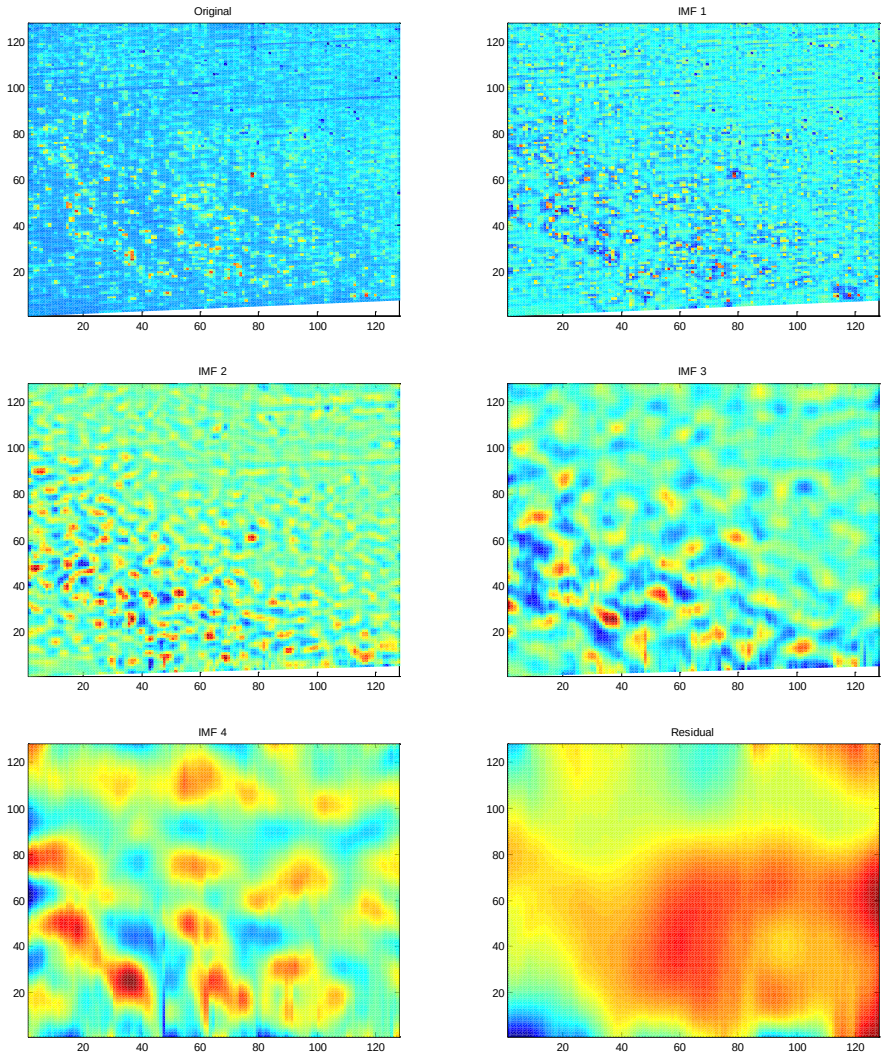


Figure 17. ROAR lidar survey from top left to bottom right. Original lidar, IMF 1, IMF 2, IMF 3, IMF 4, Residual

BEMD can also be applied to lidar imagery to filter out unwanted noise or unwanted small scale signals. The image with IMF 1 removed is the original image with the first IMF taken out, and it clearly shows improvement in defining bottom objects. About five bottom objects can be visually determined as significant, compared to the original, which appears to have over 15 objects speckled on the seafloor. There is also at $x = 40$, $y = 20$ one object, which has a stronger signal return that the other objects of interest. Further filtering by removal of IMF 2 shows that there is clearly one object that

stands out. This corresponds to the same object at *x=40*, *y=20*. With IMF 3 removed, signals seem to somewhat represent objects on the seafloor, but the signal is larger scale and not well defined. Further filtering into the low band pass range causes removal of any clear definition of the object of interest, but it is shown here to demonstrate how BEMD decomposes a signal from high to low wavenumber.

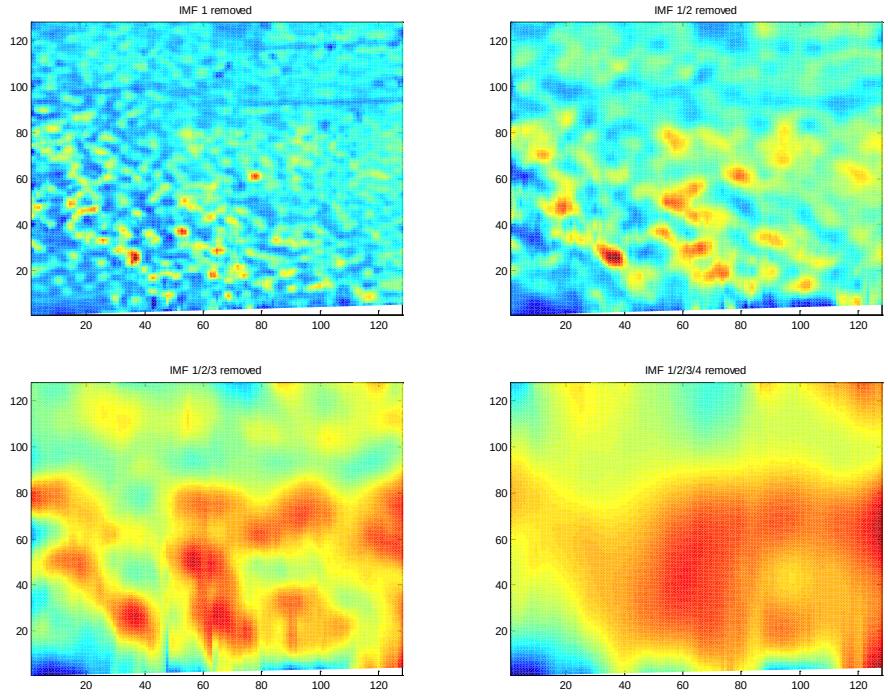


Figure 18. ROAR lidar image from top left to bottom right: IMF 1 removed, IMF1/2 removed, IMF 1/2/3 removed, IMF 1/2/3/4 removed.

Another lidar image taken from the same survey in Figure 16, numbered 4, shows the upper right corner of the image scattered with contacts (Figure 19). The original image shows some high signal return objects, but they do not contrast against the background well. IMF 1 also shows contacts scattered on the seafloor as well, with 10–15 possible contacts, but objects in the image still lack significant contrast. IMF 2 shows about seven contacts, which show red pixels in the center, those contacts are well defined and stand out against other contacts. IMF 3 shows two objects with black centered pixels located around *x=85* and *y=100, y=120*. One of those objects correlates with the same object of interest in the IMF 2 image.

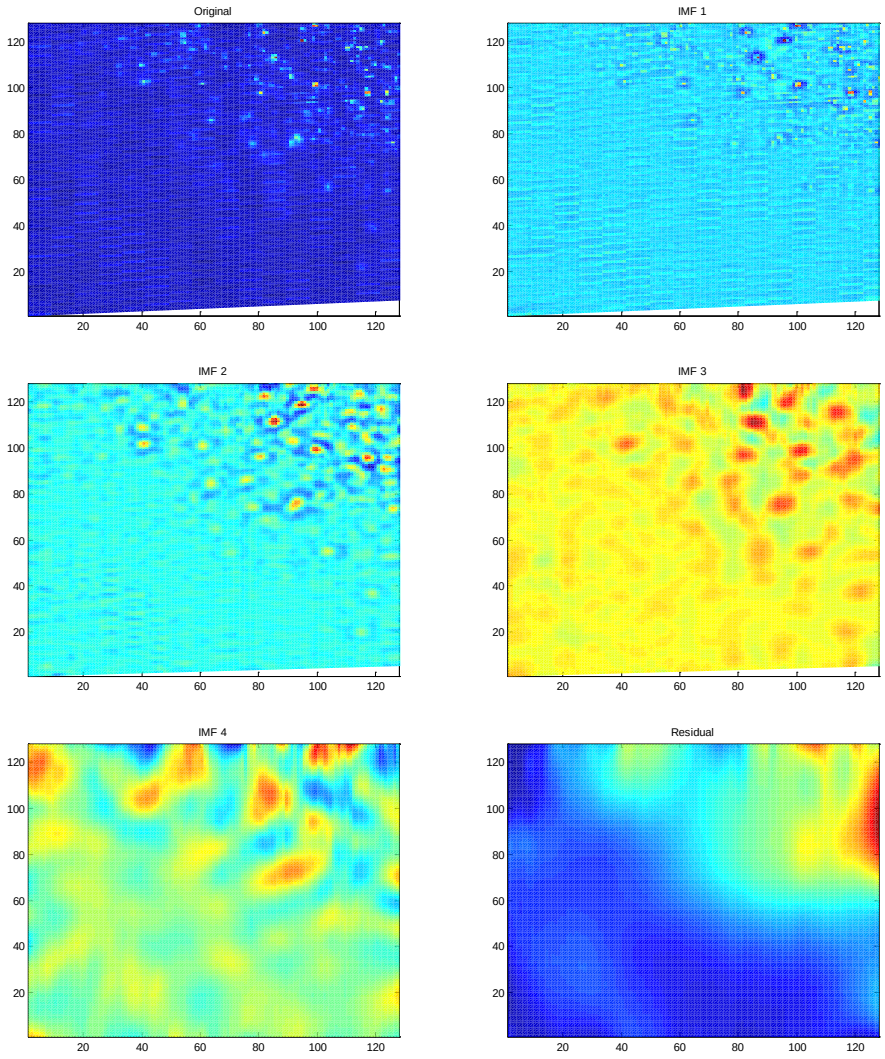


Figure 19. ROAR lidar survey from top left to bottom right: Original lidar, IMF 1, IMF 2, IMF 3, IMF 4, Residual.

IMF 4 shows larger scale contacts but at different locations not associated with the previously defined contacts. This is representative of a signal return related to background features. These stronger return spots in the higher IMF are not likely to be mine-like contacts.

In Figure 20, with IMF 1 removed, seven mine-like contacts appear in the image as red pixels. Other mine-like contacts do appear but are weaker in signal strength. With IMF 2 removed the objects that appeared with the IMF 1 removed appear as larger scale

objects with slightly weaker signal return. Further filtering out of higher IMFs did not expose any more mine-like objects.

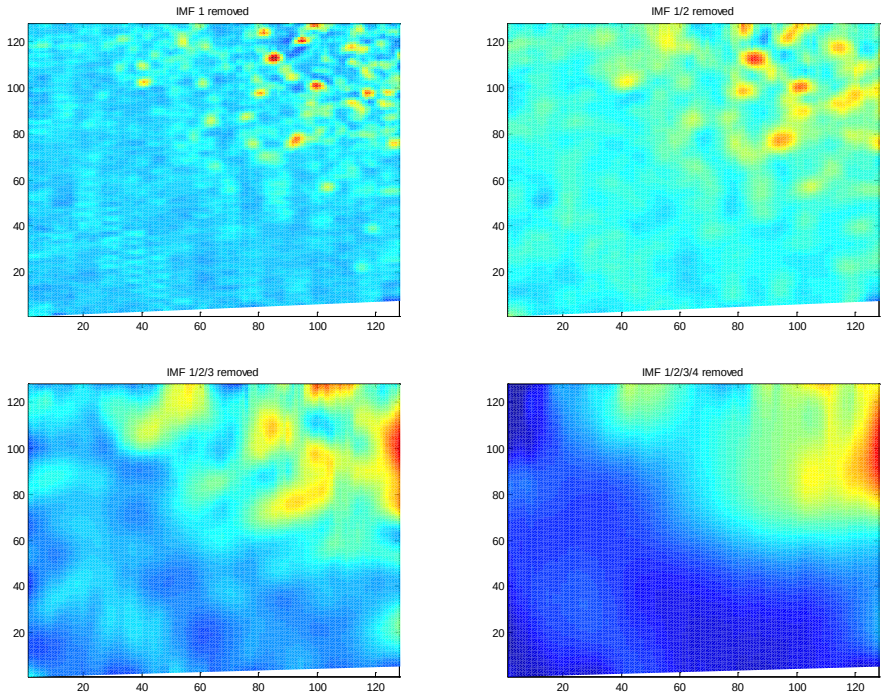


Figure 20. ROAR lidar image from top left to bottom right: IMF 1 removed, IMF1/2 removed, IMF 1/2/3 removed, IMF 1/2/3/4 removed.

In all cases where BEMD was performed on lidar imagery, filtering out just the first IMF is sufficient to produce an image that shows mine-like objects amplified against the background. Although there is still some noise and some clutter that has been left unfiltered, objects with higher signal return from the original image are preserved and show up on the filtered image as well defined objects in Figure 21.

The first original image taken from the survey track in Figure 16, numbered 2, shows strong lidar return signals in red pixels scattered about the image from the top left to the bottom right. The signal strength is weak in terms of pixel size, and causes difficulty in discerning the more prominent signal returns from the lesser prominent ones, so it appears that there is over a dozen contacts. Weaker lidar return signals in light blue pixels are scattered throughout the image, causing clutter and also making detection of the stronger red pixel values more difficult. It appears as about a dozen objects In the

BEMD filtered image on the right hand side, about five objects in the bottom right hand corner stand out as significant objects. Higher value red pixels are shown to be larger in size and can be distinguished from the slightly lower value orange pixels to the right. Distinguishing the higher value objects in the original image is more difficult and would require more time.

The second original image taken from the survey track in Figure 16, numbered 6, is nearly similar to the first original image in number of higher return signals, orientation and background. The BEMD filtered image shows about five contacts that are significant.

The third original image taken from the survey track in Figure 16, numbered 7, shows a darker background with most of the higher return signals in the upper right corner. There is also slightly less clutter which makes it a little bit easier to point out higher return values in the image. In the BEMD filtered image, four to five objects stand out as significant mine-like objects.

The fourth original image taken from the survey track in Figure 16, numbered 8, shows a seabed with what appears to be much less clutter, but it is because the return of these cluttered objects have less contrast with the background. The BEMD filtered image shows two significant objects, near the center and near the bottom left.

In all four of these BEMD filtered images, contacts were hidden or masked in noise, but the contacts were small in size and difficult to visually detect. BEMD allowed these significant contacts to be more visually detectable.

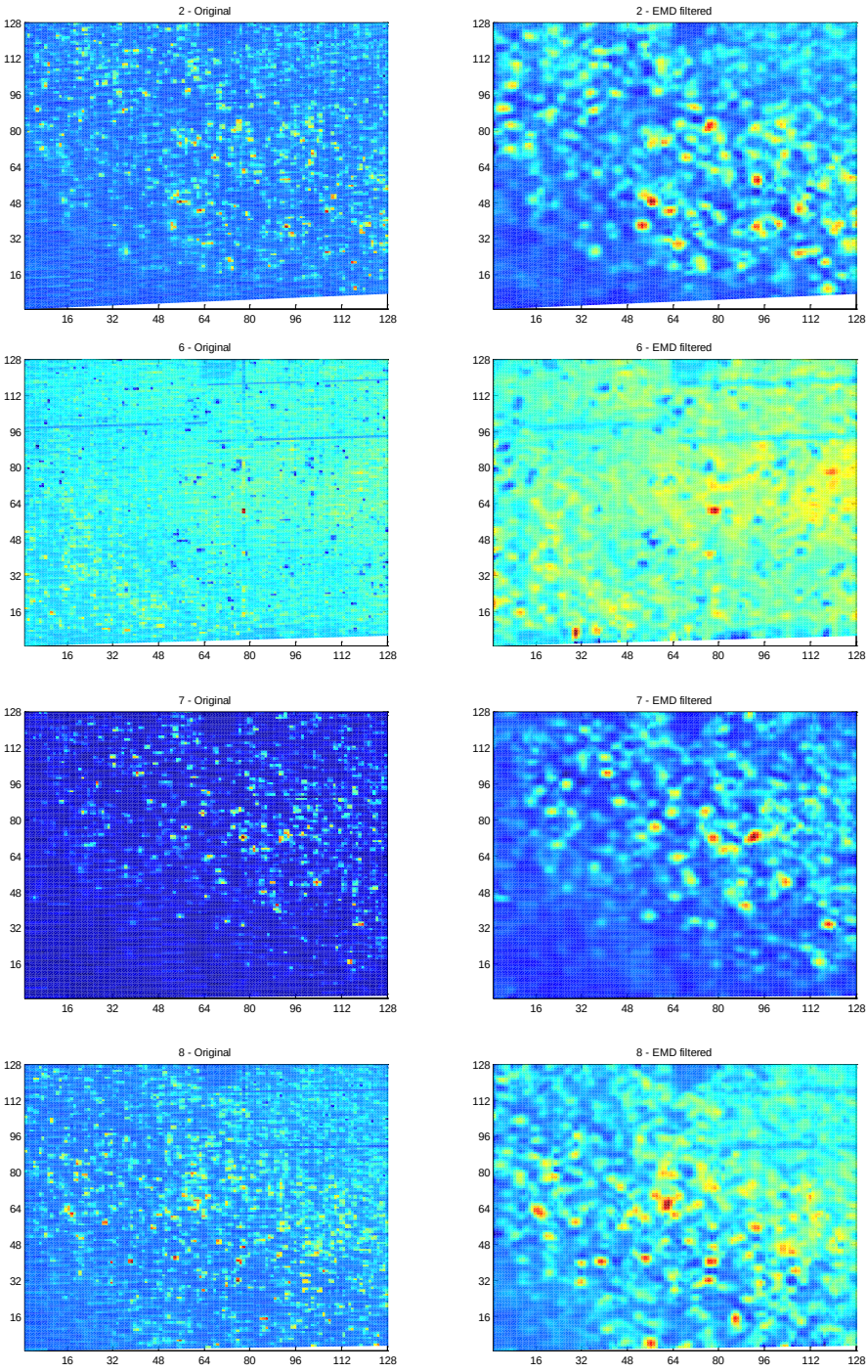


Figure 21. Left: Original lidar images. Right: EMD filtered image with IMF 1 removed.

With IMF 1 removed, higher value pixels become more evident and contrast against the background is improved compared to the original image in Figure 19. Histogram plots show the BEMD filtered image compared to the original image (Figure 22, 23). Frequency of values range from 0 to around 10,000, and was plotted on a logarithmic scale. Values of each image were scaled down to a range of 0 to 1 in 1000 bins.

Figure 22 shows that frequency of values in the low range 0–100 are much higher in the original image compared to the BEMD filtered image. Frequencies of values in the mid and high range are higher in the BEMD filtered image compared to the original image. Values at the very high end from 800 to 1000 occur more often in the high BEMD filtered image than the original image. This correlates to the filtered image having a better representation of high value pixels in Figure 17, which is related to a higher contrast of object against the background.

Figure 23 also shows that frequencies of values in the low range 0–50 are much higher in the original image compared to the BEMD filtered image. Frequencies of values in the mid and high range are higher in the BEMD filtered image compared to the original image. Values at the very high end from 500 to 1000 occur more often in the high BEMD filtered image than the original image. This correlates to the filtered image having a better representation of high value pixels in Figure 19, which is related to a higher contrast of object against the background. The difference between the filtered and unfiltered image is greater in this example. The number of contacts is much less in this image.

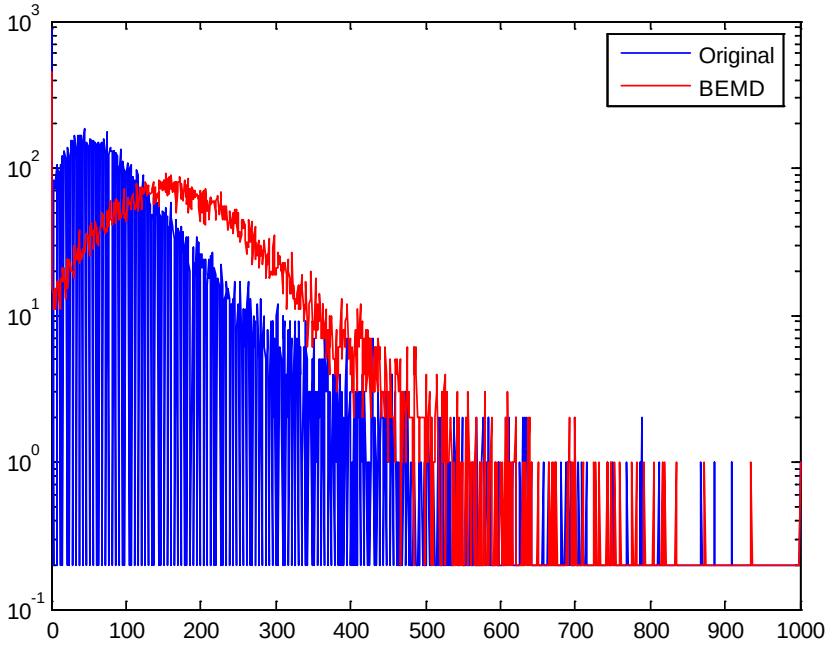


Figure 22. Original (blue) and IMF 1 removed (red) lidar imagery comparison for Figure 17.

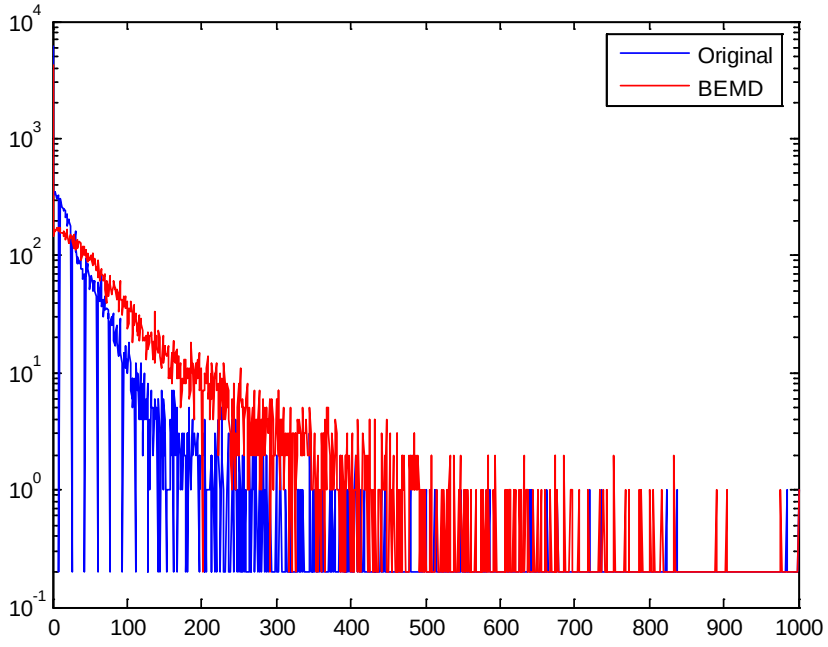


Figure 23. Original (blue) and IMF 1 removed (red) lidar imagery comparison for Figure 19.

VI. BEMD APPLICATIONS TO SIDESCAN SONAR

A. SIDESCAN SONAR DATA FROM KLEIN 5000

1. Survey Environment

Survey location was San Diego harbor for 2007 at the same location, and also Puget Sound, Bremerton, WA, in 2009. Sonar equipment used was the Klein 5000 towed sonar system. Both harbors are sufficiently protected from large scale open ocean processes. Figures 24 and 25 describe the bottom characteristics of the Klein 5000 survey.

a. San Diego Harbor

Clutter amount: 70% low, 30% high. Roughness: 70% low, 30% high.

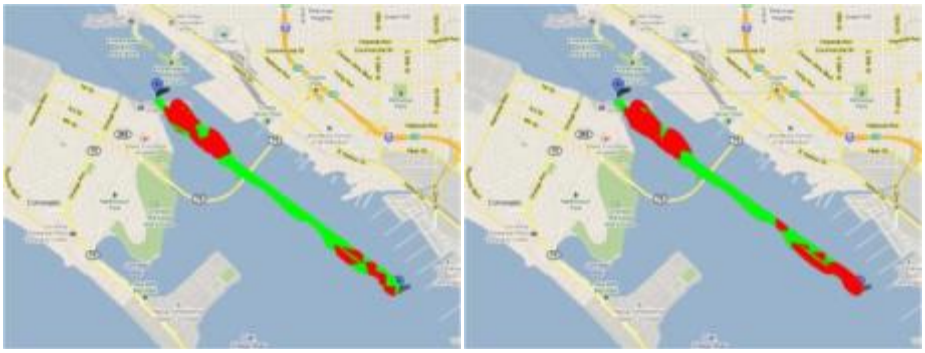


Figure 24. a) Clutter and b) roughness for San Diego harbor survey area. High values in red and low values in green.

b. Puget Sound

Clutter amount: 40% low, 60% high. Roughness: 40% low, 60% high.

Figure 25. a) Clutter and b) roughness for Puget Sound survey area. High values in red and low values in green.

2. Survey Method

Sensor was towed at a speed of 10 knots along the channel with some overlapping areas.

3. Survey Data

Raw sonar data was converted into a mosaic using NAVO's EPMA 3.0 program, then converted into a JPEG image. Only survey areas with mine-like contacts were captured and analyzed for the study. The JPEG format data were input in to MATLAB to perform EMD through a function called HFT2Dsub. The output of the file was two IMF layers which were decomposed from the original image, only the first IMF was removed to avoid loss of the object's signal. Stop criteria of *SD=0.2* was used.

4. Findings

Mine-like objects of significant size and circular shape remained detectable in the filtered dataset. Smaller mine-like shapes, which may be assumed as false contacts had a reduced signal due to filtering. This filtering was able to show that smaller objects which is too small to be considered. Similar to the ROAR data, IMF 1 was removed and compared to the original image. In EPMA, areas that had significant bottom objects were cropped into 128 x 128 images and run through the BEMD process.

5. Results

Objects of interest or large objects in a low-noise or noise free environment are not amplified by BEMD. It actually produces a blurring effect which has the ability to filter out smaller objects which could produce false alarms. Objects that are 2 to 3 pixels in size are nearly completely filtered out, and objects that are six pixels or more in size retain their shape and shadow features.

VII. SUMMARY

Mines continue to be a viable threat to our fleet operations and homeport security. Ideally MCM operations should be conducted quick and efficiently. Lidar sensors being deployed on aircraft are allowing for faster surveys and clearance of the mine threat. The very shallow water and surf zone are difficult areas to survey due to many dynamic processes which create noise and inhibit sensor performance in detecting mines. BEMD filtering proves to be effective in the noise problem present in lidar imagery to amplify the object of interest. BEMD effectiveness could not be fully researched due to the low resolution of airborne lidar, a disadvantage that comes with conducting an expeditious survey.

Knowledge of environmental conditions prior to survey is important as mine objects can be completely undetectable even after filtering due heavy sediment and turbidity in the water. This understanding can allow understanding of the operating ranges and conditions for the lidar sensor to perform detection of mines (V. T. Holmes 2003). The results of BEMD performed on lidar imagery are studied in this thesis only by qualitative measurements. A survey with observed surf conditions, sea state, surface currents, sediments, water depth, tidal data, and actual verification of bottom objects would provide good data to effectively measure filter capability of BEMD.

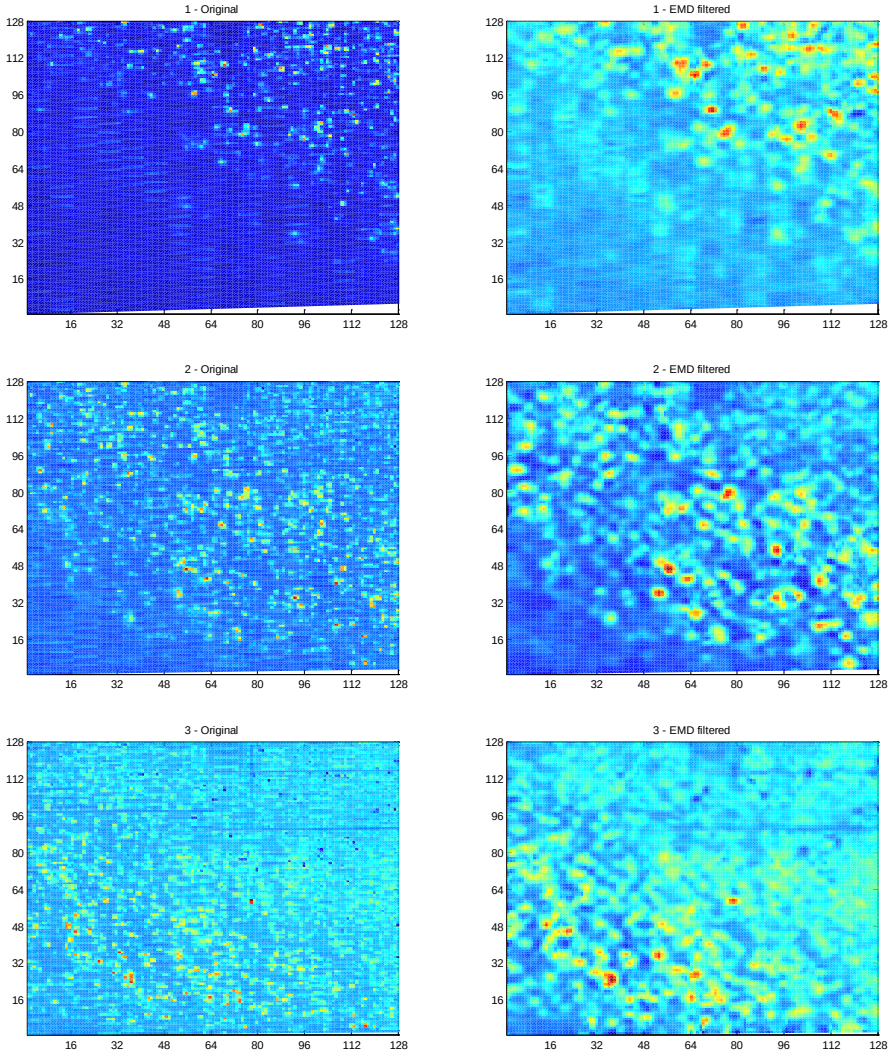
BEMD computational time is higher than other methods, as the image is decomposed starting from the small scale and successively filtering out the larger scale features. In order to filter out noise for a high resolution image, filtering down to the larger scale is required, but for low resolution imagery as in the ROAR lidar data, the object is only a few pixels in size and requires only one or two layers to be removed. Applying BEMD to an image using a compiled executable program rather than MATLAB will also improve computational time. Spline method and stop criteria can also be adjusted for BEMD which would change the computational time.

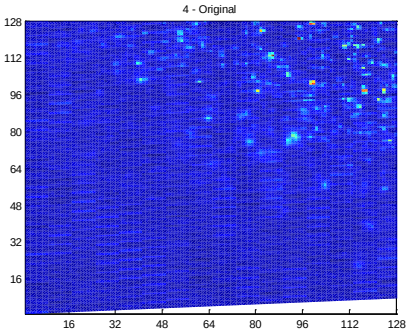
Sonar imagery from towed sensors is relatively minimal in noise, as the Klein 5000 survey of San Diego and Puget Sound did not provide much noise for BEMD to

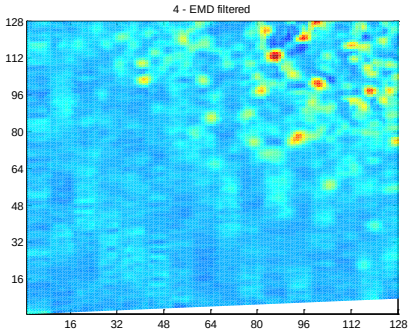
filter. But the use of sonar in shallow and dynamic environments would produce non-linear signals and noise which could be analyzed by BEMD. Side-scan sonar survey of the surf zone may not be an effective means of surveying, but worth conducting to investigate the effectiveness of filters in removing noise.

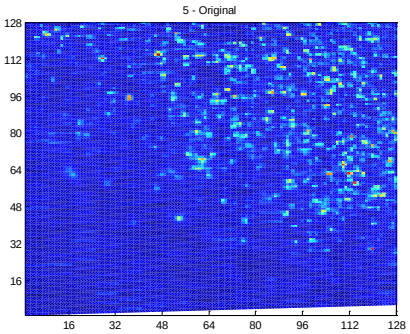
APPENDIX A. ROAR DATA EMD FILTERED FORT WALTON, FL, LIDAR IMAGES

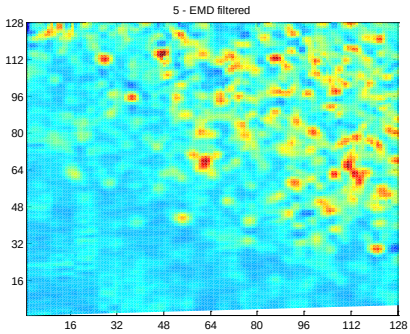
Out of 53 images captured from the lidar sensor, 16 were free of serious data impurities. Image dimensions are 128 by 128 pixels. Original image (left column) is a 1.5-meter-per-pixel image with a map orientation, with north and shallower water towards the top of the image. The EMD filtered image (right column) is the original image with only the first IMF filtered out.

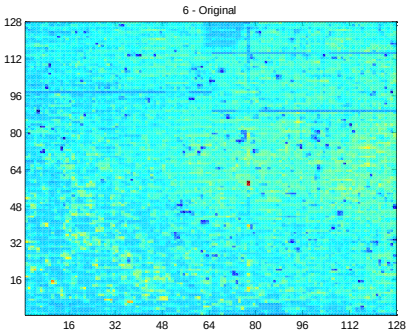


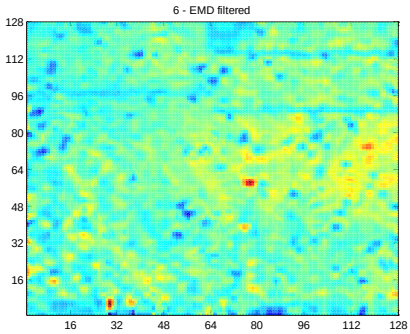

4 - Original
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

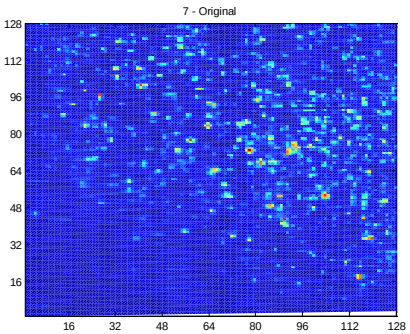

4 - EMD filtered
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

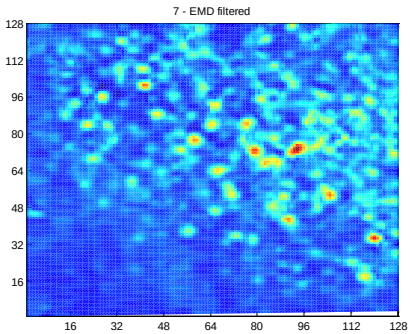

5 - Original
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

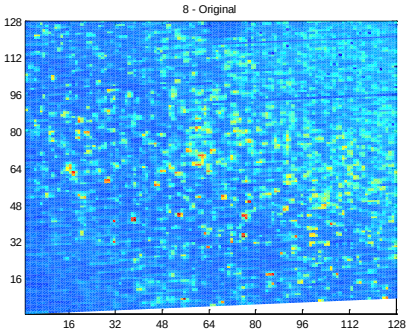

5 - EMD filtered
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

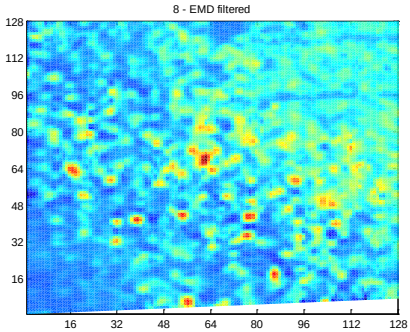

6 - Original
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

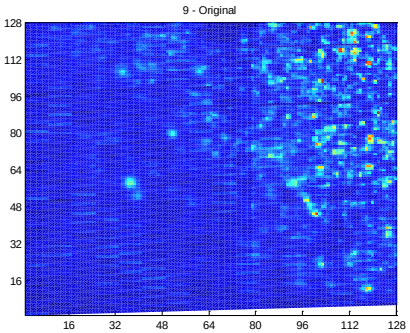

6 - EMD filtered
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

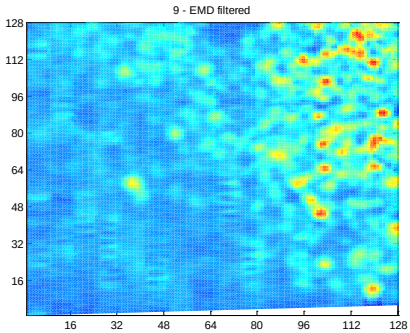

7 - Original
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

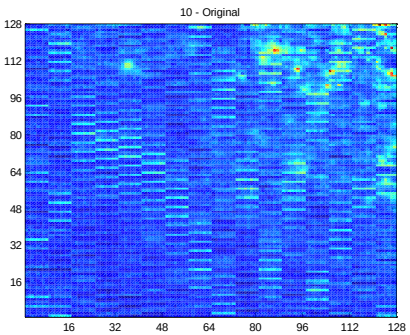

7 - EMD filtered
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

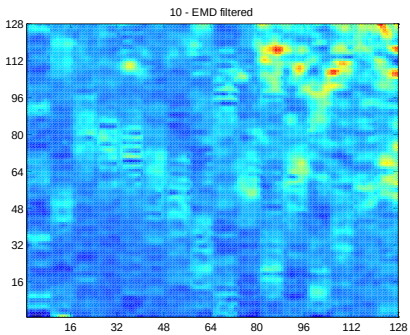

8 - Original
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

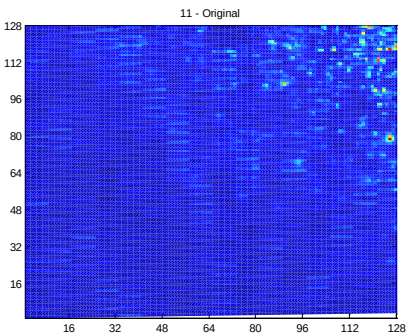

8 - EMD filtered
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

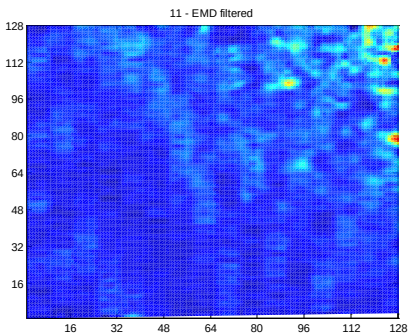

9 - Original
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

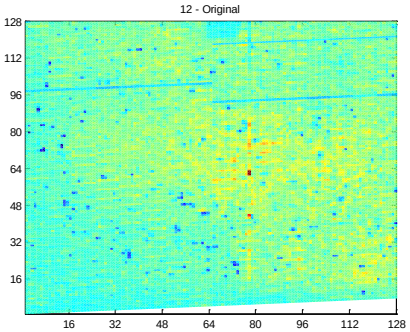

9 - EMD filtered
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

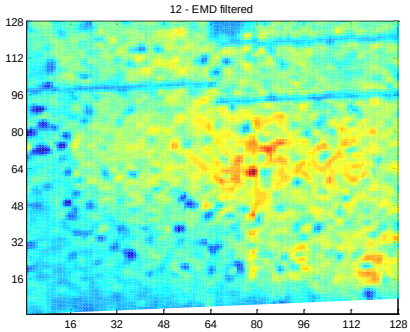

10 - Original
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

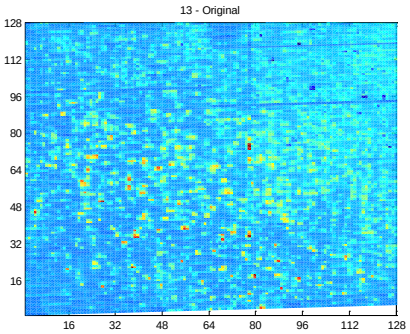

10 - EMD filtered
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

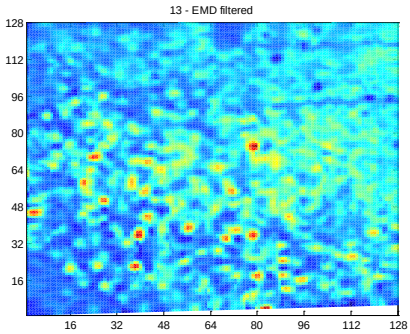

11 - Original
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

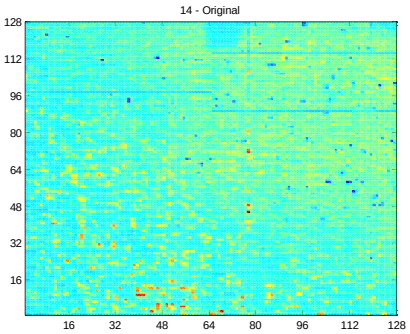

11 - EMD filtered
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

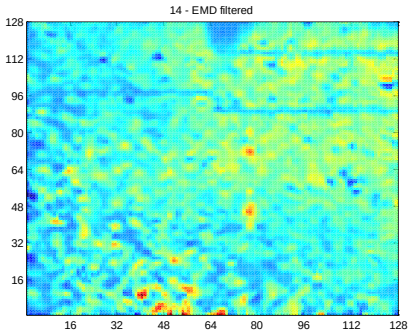

12 - Original
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

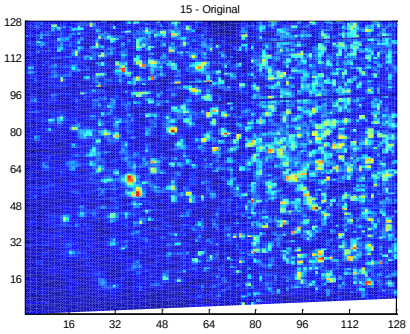

12 - EMD filtered
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

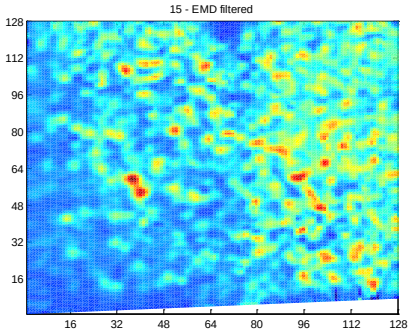

13 - Original
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

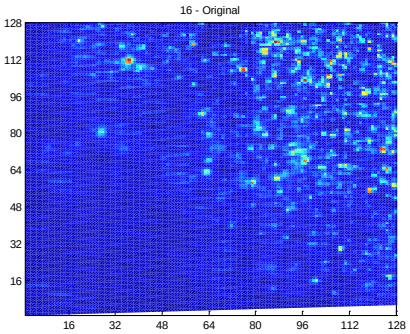

13 - EMD filtered
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128

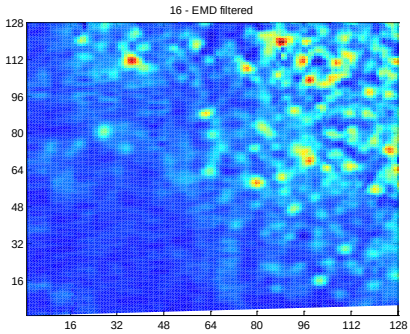

14 - Original
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128


14 - EMD filtered
128
112
96
80
64
48
32
16
16 32 48 64 80 96 112 128


15 - Original
128
112
96
80
64
48
32
16
16
32
48
64
80
96
112
128


15 - EMD filtered
128
112
96
80
64
48
32
16
16
32
48
64
80
96
112
128


16 - Original
128
112
96
80
64
48
32
16
16
32
48
64
80
96
112
128


16 - EMD filtered
128
112
96
80
64
48
32
16
16
32
48
64
80
96
112
128

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. MATLAB CODE FOR FILTERS

A. BEMD ALGORITHM

```
function rsltf = HFT2dsub(data);

% loading image data
% load lena2.dat;
% load lena2.mat;
% image data size in one-dimension
%N=111;
[N,M]=size(data);
TNM=4; TNM1=TNM+1;
rsltd1=zeros(N,M,TNM1); rsltd2=zeros(N,M,TNM1,TNM1);
% specify ensemble number
%nesb=100;
nesb=N-11;
% decomposition in the first dimension and arrange the output
for j=1:N
    disp(['j=',int2str(j)]);
    temp=data(j,:);
    rslt=eemdrow(temp,0.2,nesb,TNM);
    rslt
    for k=1:TNM1
        rsltd1(j,:,k)=rslt(k+1,:);
    end
end
% decomposition in the second direction
for k=1:TNM1
    for i=1:M
        disp([int2str(k),'   i=',int2str(i)]);
        temp2=rsltd1(:,i,k);
        rslt=eemdcol(temp2,0.2,nesb,TNM);
        for kk=1:TNM1
            rslt2d(:,i,k,kk)=rslt(:,kk+1);
        end
    end
end
% combine modes
for m=1:TNM1
    rsltf(:,:,m)=rslt2d(:,:,m,m);
    for k=m+1:TNM1
        rsltf(:,:,m)=rsltf(:,:,m)+rslt2d(:,:,k,m);
        rsltf(:,:,m)=rsltf(:,:,m)+rslt2d(:,:,m,k);
    end
end
```

```
function allmode=eemdcol(Y,Nstd,NE,TNM)
% Y: Inputted data;
% Nstd: ratio of the standard deviation of the added noise and that of
Y;
% NE: Ensemble member being used
% TNM: total number of modes (not including the trend)
%
% find data length
xsize=length(Y);
dd=[1:1:xsize]';
% Nornaliz data
Ystd=std(Y);
Y=Y/Ystd;
% Initialize saved data
allmode=zeros(xsize,TNM+2);
for iii=1:1:NE,
    % adding noise
    temp=randn(xsize,1)*Nstd;
    X1=Y+temp;
    X2=Y-temp;
    % sifting X1
    xorigin = X1;
    xend = xorigin;
    % save the initial data into the first column
    allmode(:,1)=allmode(:,1)+xorigin;
    for nmode =2:TNM+1
        xstart = xend;
        for iter=1:10
            [spmax, spmin, flag]=extremacol(xstart);
            upper= spline(spmax(:,1),spmax(:,2),dd);
            lower= spline(spmin(:,1),spmin(:,2),dd);
            mean_ul = (upper + lower)/2;
            xstart = xstart - mean_ul;
        end
        xend = xend - xstart;
        % save a mode
        allmode(:,nmode)=allmode(:,nmode)+xstart;
    end
    % save the trend
    allmode(:,TNM+2)=allmode(:,TNM+2)+xend;
    %%%===========================================================
    % sifting X2
    xorigin = X2;
    xend = xorigin;
    % save the initial data into the first column
    allmode(:,1) =allmode(:,1)+ xorigin;
    for nmode =2: TNM+1
        xstart = xend;
        for iter=1:10
            [spmax, spmin, flag]=extremacol(xstart);
            upper= spline(spmax(:,1),spmax(:,2),dd);
            lower= spline(spmin(:,1),spmin(:,2),dd);
            mean_ul = (upper + lower)/2;
            xstart = xstart - mean_ul;
        end
```

```
        xend = xend - xstart;
        % save a mode
        allmode(:,nmode) = allmode(:,nmode)+xstart;
    end
    % save the trend
    allmode(:,TNM+2)=allmode(:,TNM+2)+xend;
end
% ensemble average
allmode=allmode/NE/2;
% Rescale mode to origional unit.
allmode=allmode*Ystd;
```

```
% Y: Inputted data;
% Nstd: ratio of the standard deviation of the added noise and that of
Y;
% NE: Ensemble member being used
% TNM: total number of modes (not including the trend)
%
function allmode=eemdrow(Y,Nstd,NE,TNM)
% find data length
xsize=length(Y);
dd=[1:1:xsize];
% Nornaliz data
Ystd=std(Y);
Y=Y/Ystd;
% Initialize saved data
allmode=zeros(TNM+2,xsize);
for iii=1:1:NE,
    % adding noise
    temp=randn(1,xsize)*Nstd;
    X1=Y+temp;
    X2=Y-temp;
    % sifting X1
    xorigin = X1;
    xend = xorigin;
    % save the initial data into the first column
    allmode(1,:)=allmode(1,:)+xorigin;
    for nmode =2:TNM+1
        xstart = xend;
        for iter=1:10
            [spmax, spmin, flag]=extremarow(xstart);
            upper= spline(spmax(1,:),spmax(2,:),dd);
            lower= spline(spmin(1,:),spmin(2,:),dd);
            mean_ul = (upper + lower)/2;
            xstart = xstart - mean_ul;
        end
        xend = xend - xstart;
        % save a mode
        allmode(nmode,:)=allmode(nmode,:)+xstart;
    end
    % save the trend
    allmode(TNM+2,:)=allmode(TNM+2,:)+xend;
    %%%============================================================
    % sifting X2
    xorigin = X2;
    xend = xorigin;
    % save the initial data into the first column
    allmode(1,:) =allmode(1,:)+ xorigin;
    for nmode =2: TNM+1
        xstart = xend;
        for iter=1:10
            [spmax, spmin, flag]=extremarow(xstart);
            upper= spline(spmax(1,:),spmax(2,:),dd);
            lower= spline(spmin(1,:),spmin(2,:),dd);
            mean_ul = (upper + lower)/2;
            xstart = xstart - mean_ul;
        end
```

```
        xend = xend - xstart;
        % save a mode
        allmode(nmode,:) = allmode(nmode,:)+xstart;
    end
    % save the trend
    allmode(TNM+2,:)=allmode(TNM+2,:)+xend;
end
% ensemble average
allmode=allmode/NE/2;
% Rescale mode to origional unit.
allmode=allmode*Ystd;
```

```
function [spmax, spmin, flag]= extremacol(in_data);

flag=1;
dsize=size(in_data);
N=dsize(1); M=dsize(2);
dd=diff(in_data(:,1));
% find all the maximums
ii=find(dd(1:end-1)>=0 & dd(2:end)<=0)+1;
spmax=cat(2,ii,in_data(ii,1));
% Exten interpolate to two end
spmax=cat(1,[1,in_data(1,1)],spmax,[N,in_data(end,1)]);
if(size(spmax,1)>=4)
    slope1=(spmax(2,2)-spmax(3,2))/(spmax(2,1)-spmax(3,1));
    tmp1=slope1*(spmax(1,1)-spmax(2,1))+spmax(2,2);
    if tmp1>spmax(1,2)
        spmax(1,2)=tmp1;
    end
    slope2=(spmax(end-1,2)-spmax(end-2,2))/(spmax(end-1,1)-spmax(end-
2,1));
    tmp2=slope2*(spmax(end,1)-spmax(end-1,1))+spmax(end-1,2);
    if tmp2>spmax(end,2)
        spmax(end,2)=tmp2;
    end
else
    flag=-1;
end
%  find all the minimums
ii=find(dd(1:end-1)<=0 & dd(2:end)>=0)+1;
spmin=cat(2,ii,in_data(ii,1));
% Exten interpolate to two end
spmin=cat(1,[1,in_data(1,1)],spmin,[N,in_data(end,1)]);
if(size(spmin,1)>=4)
    slope1=(spmin(2,2)-spmin(3,2))/(spmin(2,1)-spmin(3,1));
    tmp1=slope1*(spmin(1,1)-spmin(2,1))+spmin(2,2);
    if tmp1<spmin(1,2)
        spmin(1,2)=tmp1;
    end
    slope2=(spmin(end-1,2)-spmin(end-2,2))/(spmin(end-1,1)-spmin(end-
2,1));
    tmp2=slope2*(spmin(end,1)-spmin(end-1,1))+spmin(end-1,2);
    if tmp2<spmin(end,2)
        spmin(end,2)=tmp2;
    end
else
    flag=-1;
end

if(M==2)
    spmax(:,1)=in_data(spmax(:,1),2);
    spmin(:,1)=in_data(spmin(:,1),2);
end
```

```
function [spmax, spmin, flag]= extremarow(in_data);

flag=1;
dsize=size(in_data);
N=dsize(2); M=dsize(1);
dd=diff(in_data(1,:));
% find all the maximums
ii=find(dd(1:end-1)>=0 & dd(2:end)<=0)+1;
spmax=cat(1,ii,in_data(1,ii));
% Exten interpolate to two end
spmax=cat(2,[1;in_data(1,1)],spmax,[N;in_data(1,end)]);
if(size(spmax,2)>=4)
    slope1=(spmax(2,2)-spmax(2,3))/(spmax(1,2)-spmax(1,3));
    tmp1=slope1*(spmax(1,1)-spmax(1,2))+spmax(2,2);
    if tmp1>spmax(2,1)
        spmax(2,1)=tmp1;
    end
    slope2=(spmax(2,end-1)-spmax(2,end-2))/(spmax(1,end-1)-spmax(1,end-
2));
    tmp2=slope2*(spmax(1,end)-spmax(1,end-1))+spmax(2,end-1);
    if tmp2>spmax(2,end)
        spmax(2,end)=tmp2;
    end
else
    flag=-1;
end
%  find all the minimums
ii=find(dd(1:end-1)<=0 & dd(2:end)>=0)+1;
spmin=cat(1,ii,in_data(1,ii));
% Exten interpolate to two end
spmin=cat(2,[1;in_data(1,1)],spmin,[N;in_data(1,end)]);
if(size(spmin,2)>=4)
    slope1=(spmin(2,2)-spmin(2,3))/(spmin(1,2)-spmin(1,3));
    tmp1=slope1*(spmin(1,1)-spmin(1,2))+spmin(2,2);
    if tmp1<spmin(2,1)
        spmin(2,1)=tmp1;
    end
    slope2=(spmin(2,end-1)-spmin(2,end-2))/(spmin(end-1)-spmin(1,end-
2));
    tmp2=slope2*(spmin(1,end)-spmin(1,end-1))+spmin(2,end-1);
    if tmp2<spmin(2,end)
        spmin(2,end)=tmp2;
    end
else
    flag=-1;
end

if(M==2)
    spmax(1,:)=in_data(2,spmax(1,:));
    spmin(1,:)=in_data(2,spmin(1,:));
end
```

B. MEDIAN FILTER

```
function b = medfilt2(varargin)
%MEDFILT2 Perform 2-D median filtering.
%   B = MEDFILT2(A,[M N]) performs median filtering of the matrix
%   A in two dimensions. Each output pixel contains the median
%   value in the M-by-N neighborhood around the corresponding
%   pixel in the input image. MEDFILT2 pads the image with zeros
%   on the edges, so the median values for the points within
%   [M N]/2 of the edges may appear distorted.
%
%   B = MEDFILT2(A) performs median filtering of the matrix A
%   using the default 3-by-3 neighborhood.
%
%   B = MEDFILT2(...,PADOPT) controls how the matrix boundaries
%   are padded.  PADOPT may be 'zeros' (the default),
%   'symmetric', or 'indexed'. If PADOPT is 'zeros', A is padded
%   with zeros at the boundaries. If PADOPT is 'symmetric', A is
%   symmetrically extended at the boundaries. If PADOPT is
%   'indexed', A is padded with ones if it is double; otherwise
%   it is padded with zeros.
%
%   Class Support
%   -------------
%   The input image A can be logical or numeric (unless the
%   'indexed' syntax is used, in which case A cannot be of class
%   uint16).  The output image B is of the same class as A.
%
%   Remarks
%   -------
%   If the input image A is of integer class, all of the output
%   values are returned as integers. If the number of
%   pixels in the neighborhood (i.e., M*N) is even, some of the
%   median values may not be integers. In these cases, the
%   fractional parts are discarded. Logical input is treated
%   similarly.
%
%   Example
%   -------
%       I = imread('eight.tif');
%       J = imnoise(I,'salt & pepper',0.02);
%       K = medfilt2(J);
%       figure, imshow(J), figure, imshow(K)
%
%   See also FILTER2, ORDFILT2, WIENER2.

%   Copyright 1993-2004 The MathWorks, Inc.
%   $Revision: 5.18.4.7 $  $Date: 2004/08/10 01:40:54 $

[a, mn, padopt] = parse_inputs(varargin{:});

domain = ones(mn);
if (rem(prod(mn), 2) == 1)
    order = (prod(mn)+1)/2;
    b = ordfilt2(a, order, domain, padopt);
```

```
else
    order1 = prod(mn)/2;
    order2 = order1+1;
    b = ordfilt2(a, order1, domain, padopt);
    b2 = ordfilt2(a, order2, domain, padopt);
if islogical(b)
        b = b | b2;
else
        b = imlincomb(0.5, b, 0.5, b2);
end
end



%%%
%%% Function parse_inputs
%%%
function [a, mn, padopt] = parse_inputs(varargin)
iptchecknargin(1,4,nargin,mfilename);

% There are several grandfathered syntaxes we have to catch
% and parse successfully, so we're going to use a strategy
% that's a little different that usual.
%
% First, scan the input argument list for strings.  The
% string 'indexed', 'zeros', or 'symmetric' can appear basically
% anywhere after the first argument.
%
% Second, delete the strings from the argument list.
%
% The remaining argument list can be one of the following:
% MEDFILT2(A)
% MEDFILT2(A,[M N])
% MEDFILT2(A,[M N],[Mb Nb])
%
% Any syntax in which 'indexed' is followed by other arguments
% is grandfathered.  Any syntax in which [Mb Nb] appears is
% grandfathered.
%
% -sle, March 1998

a = varargin{1};

charLocation = [];
for k = 2:nargin
if (ischar(varargin{k}))
        charLocation = [charLocation k];
end
end

if (length(charLocation) > 1)
% More than one string in input list
    eid = 'Images:medfilt2:tooManyStringInputs';
    error(eid,'%s','Too many input string arguments.');
elseif isempty(charLocation)
```

```
% No string specified
    padopt = 'zeros';
else
    options = {'indexed', 'zeros', 'symmetric'};

    padopt = iptcheckstrs(varargin{charLocation}, options,
mfilename, ...
'PADOPT', charLocation);

    varargin(charLocation) = [];
end

if (strcmp(padopt, 'indexed'))
if (isa(a,'double'))
        padopt = 'ones';
else
        padopt = 'zeros';
end
end

if length(varargin) == 1,
  mn = [3 3];% default
elseif length(varargin) >= 2,
  mn = varargin{2}(:).';
if size(mn,2)~=2,
    msg = 'MEDFILT2(A,[M N]): Second argument must consist of two
integers.';
    eid = 'Images:medfilt2:secondArgMustConsistOfTwoInts';
    error(eid, msg);
elseif length(varargin) > 2,
    msg = ['MEDFILT2(A,[M N],[Mb Nb],...) is an obsolete syntax. [Mb
Nb]'...
' argument is ignored.'];
    wid = 'Images:medfilt2:obsoleteSyntax';
    warning(wid, msg);
end
end

% The grandfathered [Mb Nb] argument, if present, is ignored.
```

C. ORDERED FILTER

```
function B = ordfilt2(varargin)
%ORDFILT2 Perform 2-D order-statistic filtering.
%   B=ORDFILT2(A,ORDER,DOMAIN) replaces each element in A by the
%   ORDER-th element in the sorted set of neighbors specified by
%   the nonzero elements in DOMAIN.
%
%   B = ORDFILT2(A,ORDER,DOMAIN,S), where S is the same size as
%   DOMAIN, uses the values of S corresponding to the nonzero
%   values of DOMAIN as additive offsets.
%
%   B = ORDFILT2(...,PADOPT) controls how the matrix boundaries
%   are padded.  PADOPT may be 'zeros' (the default) or
%   'symmetric'.  If PADOPT is 'zeros', A is padded with zeros at
%   the boundaries.  If PADOPT is 'symmetric', A is symmetrically
%   extended at the boundaries.
%
%   Class Support
%   -------------
%   The class of A may be numeric or logical.  The class of B is
%   the same as the class of A, unless the additive offset form of
%   ORDFILT2 is used, in which case the class of B is double.
%
%   Example
%   -------
%   Use a maximum filter on snowflakes.png with a [5 5] neighborhood.
This is
%   equivalent to imdilate(image,strel('square',5)).
%
%       A = imread('snowflakes.png');
%       B = ordfilt2(A,25,true(5));
%       figure, imshow(A), figure, imshow(B)
%
%   Remarks
%   -------
%   DOMAIN is equivalent to the structuring element used for
%   binary image operations. It is a matrix containing only 1's
%   and 0's; the 1's define the neighborhood for the filtering
%   operation.
%
%   For example, B=ORDFILT2(A,5,ONES(3,3)) implements a 3-by-3
%   median filter; B=ORDFILT2(A,1,ONES(3,3)) implements a 3-by-3
%   minimum filter; and B=ORDFILT2(A,9,ONES(3,3)) implements a
%   3-by-3 maximum filter.  B=ORDFILT2(A,4,[0 1 0; 1 0 1; 0 1 0])
%   replaces each element in A by the maximum of its north, east,
%   south, and west neighbors.
%
%   See also MEDFILT2.

%   Copyright 1993-2004 The MathWorks, Inc.
%   $Revision: 5.17.4.6 $  $Date: 2004/08/10 01:41:00 $

[A,order,domain,s,padopt,msg] = ParseInputs(varargin{:});
```

```
domainSize = size(domain);
center = floor((domainSize + 1) / 2);
[r,c] = find(domain);
r = r - center(1);
c = c - center(2);
padSize = max(max(abs(r)), max(abs(c)));
originalSize = size(A);
if (strcmp(padopt, 'zeros'))
    A = padarray(A, padSize * [1 1], 0, 'both');
elseif (strcmp(padopt, 'ones'))
% padopt of 'ones' is for support of medfilt2; it is
% undocumented
    A = padarray(A, padSize * [1 1], 1, 'both');
else
    A = padarray(A, padSize * [1 1], 'symmetric', 'both');
end
Ma = size(A,1);
offsets = c*Ma + r;

% make sure that offsets are valid
if ~isreal(offsets) || any(floor(offsets) ~= offsets) ||
any(~isfinite(offsets))
%should never get here
    eid = sprintf('Images:%s:internalError', mfilename);
    msg = 'Internal error: bad OFFSETS.';
    error(eid,'%s',msg);
end

if isempty(s)
%ORDFILT2(A,ORDER,DOMAIN)
  B = ordf(A, order, offsets, [padSize padSize] + 1, ...
             originalSize, domainSize);
else
%ORDFILT2(A,ORDER,DOMAIN,S,PADOPT)
  B = ordf(A, order, offsets, [padSize padSize] + 1, ...
           originalSize, domainSize, s);
end


%%%
%%% ParseInputs
%%%
function [A,order,domain,s,padopt,msg] = ParseInputs(varargin)

A = [];
order = [];
domain = [];
s = [];
padopt = 'zeros';
msg = '';

iptchecknargin(3,5,nargin,mfilename);

A = varargin{1};
order = varargin{2};
```

```
domain = varargin{3};
options = {‘zeros’, ‘ones’, ‘symmetric’};
% padopt of ‘ones’ is for supporting medfilt2; it is undocumented.

if (nargin == 4)
if (ischar(varargin{4}))
    padopt = iptcheckstrs(varargin{4},options,mfilename,’PADOPT’,4);
else
    s = varargin{4};
end

elseif (nargin == 5)
  s = varargin{4};
  padopt = iptcheckstrs(varargin{5},options,mfilename,’PADOPT’,5);
end

% make sure that arguments are valid
iptcheckinput(order,{‘double’},{‘real’,’scalar’,’integer’},mfilename, .
..
‘ORDER’,2);

if ~isempty(s)
if (~isa(A, ‘double’))
    A = double(A);
end
  iptcheckinput(A, {‘double’}, {‘2d’,’real’}, mfilename, ‘A’, 1);
  s = s(find(domain));
  iptcheckinput(s, {‘double’}, {‘real’}, mfilename, ‘S’, 4);
else
  iptcheckinput(A, {‘numeric’,’logical’}, {‘2d’,’real’}, mfilename, ‘A’,
1);
end
```

D. WIENER FILTER

```
function [f,noise] = wiener2(varargin)
%WIENER2 Perform 2-D adaptive noise-removal filtering.
%   WIENER2 lowpass filters an intensity image that has been degraded by
%   constant power additive noise. WIENER2 uses a pixel-wise adaptive Wiener
%   method based on statistics estimated from a local neighborhood of each
%   pixel.
%
%   J = WIENER2(I,[M N],NOISE) filters the image I using pixel-wise adaptive
%   Wiener filtering, using neighborhoods of size M-by-N to estimate the local
%   image mean and standard deviation. If you omit the [M N] argument, M and N
%   default to 3. The additive noise (Gaussian white noise) power is assumed
%   to be NOISE.
%
%   [J,NOISE] = WIENER2(I,[M N]) also estimates the additive noise power
%   before doing the filtering. WIENER2 returns this estimate as NOISE.
%
%   Class Support
%   -------------
%   The input image I can be uint8, uint16, int16, double, or single.  The
%   output image J has the same class as I.
%
%   Example
%   -------
%       RGB = imread('saturn.png');
%       I = rgb2gray(RGB);
%       J = imnoise(I,'gaussian',0,0.005);
%       K = wiener2(J,[5 5]);
%       figure, imshow(J), figure, imshow(K)
%
%   See also FILTER2, MEDFILT2.

%   The following syntax is grandfathered:
%
%   J = WIENER2(I,[M N],[MBLOCK NBLOCK],NOISE) or [J,NOISE] = WIENER2(I,[M
%   N],[MBLOCK NBLOCK]) processes the intensity image I as above but in blocks
%   of size MBLOCK-by-NBLOCK.  Use J = WIENER2(I,[M N],SIZE(I),NOISE) to
%   process the matrix all at once.

%   Copyright 1993-2004 The MathWorks, Inc.
%   $Revision: 5.18.4.6 $  $Date: 2004/08/10 01:46:55 $

% Reference: "Two-Dimensional Signal and Image Processing" by
% Jae S. Lim, p. 548, equations 9.44 - 9.46.

[g, nhood, noise] = ParseInputs(varargin{:});

classin = class(g);
classChanged = false;
if ~isa(g, 'double')
  classChanged = true;
  g = im2double(g);
end

% Estimate the local mean of f.
localMean = filter2(ones(nhood), g) / prod(nhood);

% Estimate of the local variance of f.
localVar = filter2(ones(nhood), g.^2) / prod(nhood) - localMean.^2;
```

```
% Estimate the noise power if necessary.
if (isempty(noise))
  noise = mean2(localVar);
end

% Compute result
% f = localMean + (max(0, localVar - noise) ./ ...
%           max(localVar, noise)) .* (g - localMean);
%
% Computation is split up to minimize use of memory
% for temp arrays.
f = g - localMean;
g = localVar - noise;
g = max(g, 0);
localVar = max(localVar, noise);
f = f ./ localVar;
f = f .* g;
f = f + localMean;

if classChanged
  f = changeclass(classin, f);
end


%%%
%%% Subfunction ParseInputs
%%%
function [g, nhood, noise] = ParseInputs(varargin)

g = [];
nhood = [3 3];
noise = [];

wid = sprintf('Images:%s:obsoleteSyntax',mfilename);

switch nargin
case 0
    msg = 'Too few input arguments.';
    eid = sprintf('Images:%s:tooFewInputs',mfilename);
    error(eid,'%s',msg);

case 1
% wiener2(I)

    g = varargin{1};

case 2
    g = varargin{1};

switch numel(varargin{2})
case 1
% wiener2(I,noise)

        noise = varargin{2};

case 2
% wiener2(I,[m n])

        nhood = varargin{2};

otherwise
        msg = 'Invalid input syntax';
```

```
        eid = sprintf('Images:%s:invalidSyntax',mfilename);
        error(eid,'%s',msg);
end

case 3
    g = varargin{1};

if (numel(varargin{3}) == 2)
% wiener2(I,[m n],[mblock nblock])  OBSOLETE
        warning(wid,'%s %s',...
'WIENER2(I,[m n],[mblock nblock]) is an obsolete syntax.',...
'Omit the block size, the image matrix is processed all at once.');

        nhood = varargin{2};
else
% wiener2(I,[m n],noise)
        nhood = varargin{2};
        noise = varargin{3};
end

case 4
% wiener2(I,[m n],[mblock nblock],noise)  OBSOLETE
    warning(wid,'%s %s',...
'WIENER2(I,[m n],[mblock nblock],noise) is an obsolete syntax.',...
'Omit the block size, the image matrix is processed all at once.');
    g = varargin{1};
    nhood = varargin{2};
    noise = varargin{4};

otherwise
    msg = 'Too many input arguments.';
    eid = sprintf('Images:%s:tooManyInputs',mfilename);
    error(eid,'%s',msg);

end

% checking if input image is a truecolor image-not supported by WIENER2
if (ndims(g) == 3)
    msg = 'WIENER2 does not support 3D truecolor images as an input.';
    eid = sprintf('Images:%s:wiener2DoesNotSupport3D',mfilename);
    error(eid,'%s',msg);
end
```

E. IDEAL FILTER

```
%Question No:5
%IDEAL LOW-PASS FILTER

function g = idealfilter(X,P)
f=X;
[M,N]=size(f);
F=fft2(double(f));
u=0:(M-1);
v=0:(N-1);
idx=find(u>M/2);
u(idx)=u(idx)-M;
idy=find(v>N/2);
v(idy)=v(idy)-N;
[V,U]=meshgrid(v,u);
D=sqrt(U.^2+V.^2);
H=double(D<=P);
G=H.*F;
g=real(ifft2(double(G)));
%imshow(f),figure,imshow(g,[ ]);
end
```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Almquist, B., 2009: Assault breaching technologies update. MINWARA Spring 2009 regional Conference on Mine Warfare – "Home" and "Away" Game Challenges, May 18–21, 2009.

An, M., J. T. Cobb, B. Shenefelt, and R. Tolimieri, 2006: Advances in Group Filter Applications to Sea Mine Detection. *Oceans 2006*, 1–5.

Avery, J. 1998: The Naval Mine Threat to U.S. Surface Forces. *Surface Warfare*, May/June, 4–9.

Bhuiyan, S. M. A., N. O. Attoh-Okine, K. E. Barner, A. Y. Ayenu and R. R. Adhami, 2009: Bidimensional empirical mode decomposition using various interpolation techniques, *Adv. Adapt. Data Anal.*,**1,** 309–338.

Chu, P. C., Cintron, C. J., Haeger, S. D., and Keenan, R. E., 2002: Acoustic mine detection using the Navy's CASS/GRAB model. *Journal of Counter-Ordnance Technology*, 1–10.

Chu, P. C., Cintron, C. J., Haeger, S. D., Schneider, D., Keenan, R. E., and Fox, D. N., 2002: Yellow Sea acoustic uncertainty caused by hydrographic data error. *Impact of Littoral Environmental Variability on Acoustic Predictions and Sonar Performance*, Kluwer Science Publishing Co., 563–570.

Chu, P. C., Vares, N., and Keenan, R., 2004: Uncertainty in acoustic mine detection due to environmental variability, *Sixth International Symposium on Technology and Mine Problems*, NPS, Monterey, California, May 10–13, 2004.

Chu, P. C., Vares, N., 2004: Uncertainty in shallow sea acoustic detection due to environmental variability. *U.S. Navy Journal of Underwater Acoustics*, Naval Research Laboratory, **54,** 347–367 (unclassified).

Chu, P. C., Cornelious, M., and Wagstaff, M., 2005: Effects of suspended sediment on acoustic detection using reverberation. *Marine Technology Society Journal*, Marine Technology Society, **39 (2),** 105–109.

Chu, P. C., Cornelious, M., and Wagstaff, M., 2006: Suspended sediment effect on buried mine detection using CASS/GRAB model. *U.S. Navy Journal of Underwater Acoustics*, Naval Research Laboratory, 1–7 (unclassified).

Chu, P. C., Amezaga, G. R., Gottshall, E. L., and Cwalina, D. S., 2007: Ocean nowcast/forecast systems for improvement of naval undersea capabilities. *Marine Technology Society Journal*, Marine Technology Society, **41 (2),** 23–30.

Dobeck, G. J., 2001: Algorithm Fusion for Automated Sea Mine Detection and Classification, *OCEANS, 2001.*, MTS/IEEE Conference and Exhibition, **1,** 130 – 134.

Holmes, J. J., 2006: *Exploitation of a ship's Magnetic Field Signatures*, Morgan and Claypool Publishers, 1–78

Holmes, V. T., and Wright J. A., 2003: *Lidar Signatures of Very Shallow Water (VSW) and Surf Zone (SZ) Mines.* Proc. SPIE, **5089,** 284–295.

Huang, N. E., Z. Shen, Long, S. R., Wu, M. C., Shih, E. H., Zheng, Q., Tung, C. C., and Liu, H. H., 1998: *The empirical mode decomposition method and the Hilbert spectrum for non-stationary time series analysis.* Proc. Roy. Soc. London, **A454,** 903–995.

Kuosmanen, P., Astola J., 1997: *Fundamentals of Non-linear Digital Filtering*, CRC Press, 1997, 1–277.

Liu, Z. F., and Luo, Z. M., 2010: Bidimensional empirical mode decomposition for noise reduction in sonar images, *International Forum on Strategic Technology (IFOST) 2010*, IFOST.2010, 225–229.

Maussang, F., Rombaut, M., Chanussot, J., Hétet, A., and Amate, M., Fusion of Local Statistical Parameters for Buried Underwater Mine Detection in Sonar Imaging, *EURASIP Journal on Advances in Signal Processing*, Hindawi Publishing Corporation, **2008,** 1–19.

Moran, S., Austin, W. L., Murray, J. T., Roddier, N. A., Bridges, R., and Vercillo, R., 2003: Rapid Overt Airborne Reconnaissance (ROAR) for Mines and Obstacles in very shallow water, surf zone, and beach; *Proc. SPIE*, Lite Cycles Incorporated (USA), **5089,** 214–224 .

Morison, S. L. 1995. *Guide to Naval Mine Warfare.* Pasha Publications, 1–432.

Ocean Studies Board, National Research Council, 2000: *Oceanography and Mine Warfare.* National Academy of Sciences National Academy Press, 1–112.

Xu, Y., Liu, B., Liu, J., and Riemenschneider, S., 2006: Two-dimensional empirical mode decomposition by finite elements, *Proc. Roy. Soc. London*, **A 462,** 3081–3096.

Zwolski, M., LT. 1998. *Airborne Mine Countermeasures (AMCM) History*. [Available online at http:// members.aol.com/helmineron/history.htm.]

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Rear Admiral David Titley, USN
   Oceanographer and Navigator of the Navy
   Washington, DC

4. Rear Admiral Jonathan White, USN
   Commander, Naval Meteorology and Oceanography Command
   Stennis Space Center, Mississippi

5. Rear Admiral/Professor Richard Williams
   Chair of Mine Warfare
   Wayne E. Meyer Institute of Systems Engineering
   Naval Postgraduate School
   Monterey, California

6. CAPT Brain Brown
   Commander, Naval Oceanographic Office
   Stennis Space Center, Mississippi

7. CAPT Ashley Evans
   Deputy Oceanographer & Navigator of the Navy
   3450 Massachusetts Ave., NW, Bldg. 1
   Washington, DC

8. CAPT Robert E. Kiser
   Commanding Officer
   Naval Research Laboratory

9. CAPT Greg Ulses
   Commander, Naval Oceanographic Office
   Stennis Space Center, Mississippi

10. Mr. Tom Cuff
    Technical Director, Naval Oceanographic Office
    Stennis Space Center, Mississippi

11. Mr. Ronald E. Betsch
MIW Program Manager
Naval Oceanographic Office
Stennis Space Center, Mississippi

12. Dr. James Rigney
Chief Scientist
Naval Oceanographic Office
Stennis Space Center, Mississippi

13. Dr. Peter Fleischer
Naval Oceanographic Office
Stennis Space Center, Mississippi

14. CDR John Dumas
Program Officer
Naval Postgraduate School
Monterey, California

15. Professor Jeffrey Paduan
Chair, Department of Oceanography
Naval Postgraduate School
Monterey, California

16. Professor Peter Chu
Naval Postgraduate School
Monterey, California