Progressively Communicating Rich Telemetry from Autonomous Underwater Vehicles via Relays

by

Christopher Alden Murphy

B.S. Olin College (2006), M.S. MIT/WHOI Joint Program (2008)

Submitted to the Dept. of Electrical Engineering and Computer Science and the
Joint Program in Applied Ocean Science and Engineering

in partial fulfillment of the requirements for the degree of
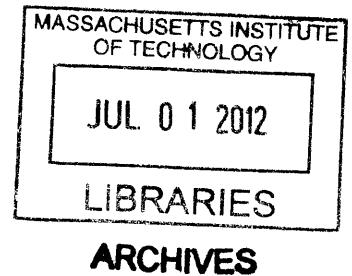
Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

and the

WOODS HOLE OCEANOGRAPHIC INSTITUTION

June 2012

Author.. χ

Dept. of Electrical Engineering and Computer Science and the
Joint Program in Applied Ocean Science and Engineering
March 2, 2012

Certified by ....

Dr. Hanumant Singh
Thesis Supervisor, Associate Scientist, WHOI

Accepted by ....

Dr. Henrik Schmidt
Chair, Joint Committee for Applied Ocean Science and Engineering, WHOI

Accepted by.............

Professor Leslie A. Kolodziejski
Chair, EECS Committee on Graduate Students, MIT

Progressively Communicating Rich Telemetry from Autonomous Underwater Vehicles via Relays

by

Christopher Alden Murphy

Abstract

As analysis of imagery and environmental data plays a greater role in mission construction and execution, there is an increasing need for autonomous marine vehicles to transmit this data to the surface. Without access to the data acquired by a vehicle, surface operators cannot fully understand the state of the mission. Communicating imagery and high-resolution sensor readings to surface observers remains a significant challenge – as a result, current telemetry from free-roaming autonomous marine vehicles remains limited to 'heartbeat' status messages, with minimal scientific data available until after recovery. Increasing the challenge, long-distance communication may require relaying data across multiple acoustic hops between vehicles, yet fixed infrastructure is not always appropriate or possible.

In this thesis I present an analysis of the unique considerations facing telemetry systems for free-roaming Autonomous Underwater Vehicles (AUVs) used in exploration. These considerations include high-cost vehicle nodes with persistent storage and significant computation capabilities, combined with human surface operators monitoring each node. I then propose mechanisms for interactive, progressive communication of data across multiple acoustic hops. These mechanisms include wavelet-based embedded coding methods, and a novel image compression scheme based on texture classification and synthesis. The specific characteristics of underwater communication channels, including high latency, intermittent communication, the lack of instantaneous end-to-end connectivity, and a broadcast medium, inform these proposals. Human feedback is incorporated by allowing operators to identify segments of data that warrant higher quality refinement, ensuring efficient use of limited throughput. I then analyze the performance of these mechanisms relative to current practices.

Finally, I present CAPTURE, a telemetry architecture that builds on this analysis. CAPTURE draws on advances in compression and delay tolerant networking to enable progressive transmission of scientific data, including imagery, across mul-

tiple acoustic hops. In concert with a physical layer, CAPTURE provides an end-to-end networking solution for communicating science data from autonomous marine vehicles. Automatically selected imagery, sonar, and time-series sensor data are progressively transmitted across multiple hops to surface operators. Human operators can request arbitrarily high-quality refinement of any resource, up to an error-free reconstruction. The components of this system are then demonstrated through three field trials in diverse environments on SeaBED, OceanServer and Bluefin AUVs, each in different software architectures.

Thesis Supervisor: Dr. Hanumant Singh
Title: Associate Scientist, WHOI

Acknowledgments

While thanking all those who have made this journey so positive is a daunting task, there's a clear place to start. Hanu, thank you for oh-so-many things: for the many cruises and once in a lifetime experiences, for your mentorship and your encouragement. Clay, thank you for your friendship, and for helping ensure that $N_{recoveries} = N_{deployments}$ on all of our cruises. You'll never beat me at arctic recovery golf, but you dig a mean antarctic ice hole.

I've been blessed with an incredibly supportive family, all of whom have frequently had more faith in me than myself. This thesis would never have been completed without the support of my wife, Susan. Thank you for knowing I could complete this, even when I did not. Kathryn, thank you for the edits, and for the kind words. Mom, thank you for the constant encouragement and cheerleading as I worked my way through. Thanks you to Nancy and Jenny for your words, and poem, of support.

Thank you to Julie, Jasper, Clio, Sandipa, Kurran and Shefali for sharing your husbands and fathers on late nights and long trips, and for welcoming me into your homes on many occasions.

Thanks to Roy Armstrong, Scott, Liz, John, Jeremy, Jeff, Erika and the rest of the crew at the NOAA Northwest and Pacific Islands Fisheries Science Centers, I will greatly miss the many hours I've spent at sea with each of you, and sincerely hope our paths cross again in the future. Liz and Scott, thank you particularly for trusting me to take risks with your robot in support of this thesis.

I thank Ryan Eustice, Hayato Kondo, Toby Schneider and Jeff Walls for their collaboration during field trials, and for putting up with several ridiculously late nights to get it all working. Thanks to Ryan also for his guidance on how to navigate the Joint Program.

I owe debts to Eric, Sandipa and Keenan in the acoustic modem group for more last-minute favors than I can count, and to John Bailey for making things work when they seemed beyond salvage.

I am grateful to those others who have made the Deep Submergence Lab such a friendly home over the last $5\frac{1}{2}$ years, including Jordan, Derya, Jeff, Geoff, and Peter. I particularly thank Mike Jakuba, for being a "uniter not a divider", as well as one heck of an all around nice guy. Many others who have walked this way before have made the path less daunting through their words and actions, including Chris Roman, James Kinsey, Ballard Blair and Sarah Webster. Thank you to Cathy Offinger for sharing Sampson, to Dana for sharing Ollie, and to both for sharing their kind words and friendly faces. Thank you also to Judy Fenwick for helping make sense of the nonsensical on occasion.

To my committee members, this thesis is stronger for your feedback, and I thank you for helping me see the bigger picture. Bill, thank you for sharing your group meetings (and pizza) with me as I started my journey, and making me feel welcomed in CSAIL when I didn't really have an academic home. John, thank you for your advice on graduate school (more of which I should have taken) and for your frequent, genuine, and warm offers of help. Seth, thank you for guiding me towards success, from early days in CSAIL to my RQE and now my thesis. Milica your help on the core of this work has been invaluable, and I thank you for showing me through both word and deed the value of collaboration between 'paper people' and 'engineering people'.

Dedicated to my amazing wife,

Susan Fredholm Murphy

I could not have a better partner in life.

In memory of my father,

Michael James Murphy

with whom I wish I could have shared this journey.

Contents

List of Figures

Glossary

Vehicles and Places

AUV Autonomous Underwater Vehicle

ROV Remotely Operated Vehicle

MBARI Monterey Bay Aquarium Research Institute

WHOI Woods Hole Oceanographic Institution

Communication and Compression

AWGN Additive White Gaussian Noise

BEC Binary Erasure Channel

BER Bit Error Rate

CAPTURE Communication Architecture for Progressive Transmission via Underwater Relays and Eavesdropping

CCL Compact Control Language

DCCL Dynamic Compact Control Language

DCT Discrete Cosine Transform

DTN Delay Tolerant Networking

DTNRG Delay Tolerant Networking Research Group

DWT Discrete Wavelet Transform

EZW Embedded Zerotree of Wavelets

FEC Forward Error Correction

FER Frame Error Rate

FH-FSK Frequency Hopping Frequency Shift Keying

ISI Inter-Symbol Interference

JPEG Joint Photographic Experts Group

MAC Media Access Control

QPSK Quadrature Phase Shift Keying

SNR Signal to Noise Ratio

SPIHT Set Partitioning in Hierarchical Trees

TCP Transmission Control Protocol

TDMA Time Division Multiple Access

UDP User Datagram Protocol

CHAPTER 1

Introduction

This thesis presents a novel method for communicating scientific telemetry from underwater vehicles. The contributions of this thesis include: a multi-hop relay protocol incorporating advances from the field of Delay Tolerant Networking (DTN) and designed for Autonomous Underwater Vehicles (AUVs), a method for incorporating human feedback into the selection of science telemetry, identification of and extensions to compression techniques well suited to underwater data, a novel compression scheme based on texture classification and synthesis, an architecture for AUV telemetry that integrates these advances, and the demonstration of the architecture's viability through a prototype system and multiple field tests on diverse vehicles.

This telemetry architecture, nicknamed CAPTURE, has been designed to enable progressive communication of rich scientific data from underwater vehicles to human operators on the ocean surface, across a sequence of free-swimming relay vehicles. Progressive transmission provides operators with a gradually improving approximation to environmental data, sonar imagery, or photographs, over the course of a normal mission. Operator feedback can be used to obtain arbitrarily high-quality refinement of specific sections of interesting data, up to an error-free reconstruction. The use of multiple relay vehicles allow efficient long-distance communication, even with contemporary fixed-power acoustic modems.

1.1 Autonomous underwater exploration

For those who study the ocean, AUVs provide unique capabilities to explore what can be an extremely forbidding environment. Free of a physical surface tether, AUVs are able to perform surveys without human intervention, and many kilometers from a surface vessel. The Sentry[132] AUV, developed at Woods Hole Oceanographic Institution (WHOI), is capable of traveling nearly one hundred kilometers on a single charge, and the Tethys vehicle developed at Monterey Bay Aquarium Research Institute (MBARI) is expected to have a range of nearly two thousand kilometers through extensive optimization of the sensor suite and hydrodynamics. AUVs enable scientists from across the oceanographic disciplines to answer questions about the health of our nation's fisheries[121] or learn the secrets of ancient ship and airplane wrecks[35]. AUVs have operated in environments as diverse as lively Puerto Rican coral reefs[4], the world's longest aqueduct[112], hydrothermal vents along the mid-oceanic ridges[133], and the Arctic seafloor[107].

This independence from a surface ship is a significant asset in polar environments, where surface movement of any sort is challenging and slow work. AUVs have proven to be particularly effective tools for under-ice research since they can range freely for great distances under the ice. Recently intensified interest in the polar regions has driven a number of AUV missions in both the Arctic and Antarctic[75, 59, 60]. Fig. 1-1 shows the SeaBED AUV preparing to survey the underside of an Antarctic ice-flow in 2010. While the earliest of these missions involved skirting the edges of ice flows, autonomous vehicles now venture farther and farther under ice from their launch point as climate scientists seek answers to vexing questions about the causes and progress of climate change[3].

The Nioghalvfjerdsfjorden Glacier, shown in Fig. 1-2, poses just such questions. The melting of Greenland ice sheets, driven by climate change, currently accounts for nearly a one millimeter rise in sea level each year[68]. A significant driver of this melting is believed to be warm subtropical waters at the ocean / ice / land triple-point[113], yet efforts to characterize these processes by any means are ham-
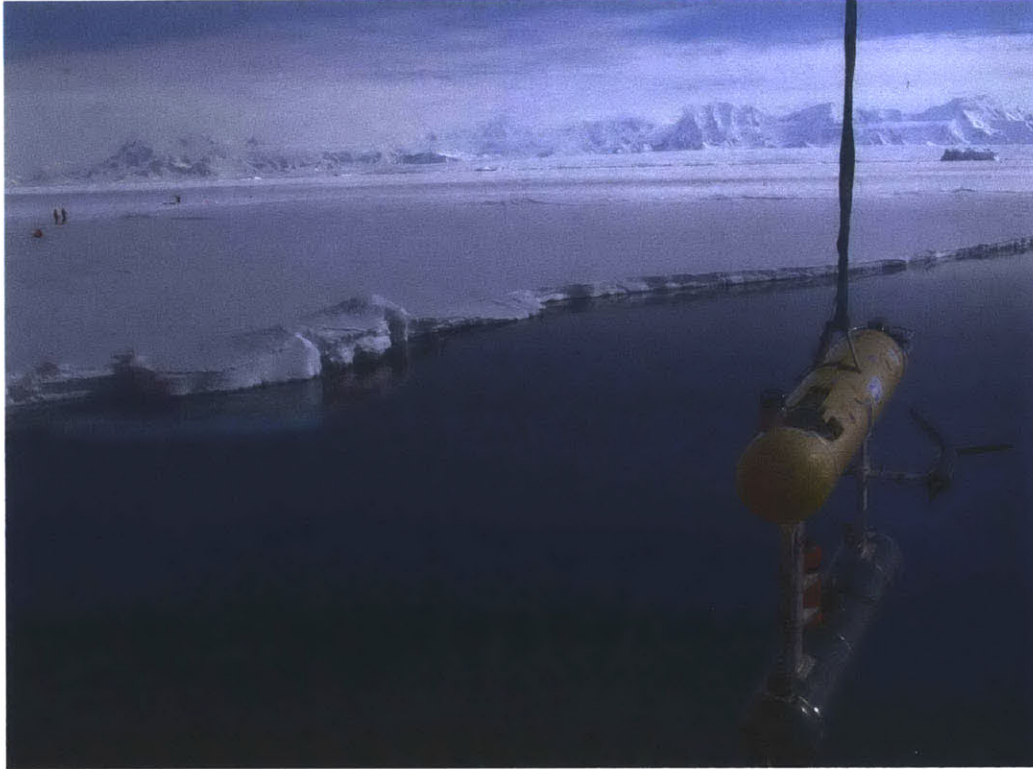
Figure 1-1: The SeaBED AUV preparing for an under-ice mission in Antarctica during the 2010 IceBELL expedition.

pered by an ice thickness of 100m or more, as shown in Fig. 1-2a. The nearest feasible access point for a vehicle is a small rift, shown in Fig. 1-2b, tens of kilometers away. An AUV could be inserted there and travel horizontally to the area of interest. Should that AUV become entrapped, or suffer a mechanical failure under the one hundred meter thick ice tongue, the environmental data it collected would likely be irrecoverable. This is in stark contrast with other field exploration robots, such as the Mars rovers[15, 2], which have returned incredibly valuable scientific data despite every vehicle remaining behind on the Martian surface.

1.2 Operator involvement

AUVs, by their very nature, do not require active human intervention to complete a mission. As the mission progresses, AUVs typically transmit small 'heartbeat messages' containing the current status of the mission and health of the vehicle. These
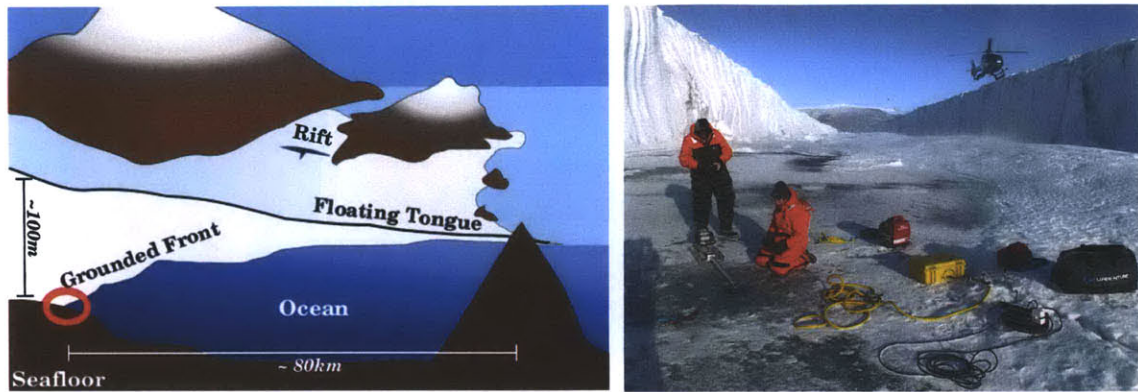
(a) Cross-section, Nioghalvfjerdsfjorden Glacier   (b) Rift, Nioghalvfjerdsfjorden Glacier

Figure 1-2: *(Left)* Cross section of the glacier showing the ocean-ice-land triple point, circled in red, and the rift where AUVs could be inserted. *(Right)* The rift in the glacier up close. The rift is a crack in the roughly 100m thick floating ice tongue (walls visible on either side), which is covered in a 10 inch layer of sea-ice. *Photo courtesy Eric Philips, IceTrek.*

messages are closely watched in realtime by human operators to ensure the safety of the vehicle, but they have historically been of little scientific value. Communication constraints ensure that the vast majority of science data is not available until the vehicle has surfaced at the completion of the mission. If AUVs were able to communicate acquired science data to surface operators, in a manner such as this thesis enables, operators would be able to review some of that data before mission completion and adjust the goals of the vehicle while it is still deployed and near the regions of interest. While the vehicle is operating normally, these ship-board operators currently represent an underutilized resource, waiting hours for the vehicle to return.

The opportunity presented by involving surface operators more closely into AUV missions was recently demonstrated during an environmental survey of the 2010 Deepwater Horizon oil spill by the Sentry AUV. During that mission, individual readings from an onboard mass spectrometer were transmitted to the surface ship[53]. Previous work by this author[74] was used for the rendering and display of the data as it was received. Even this limited, highly subsampled view of the data led to site selection and survey design, the selection of locations for further
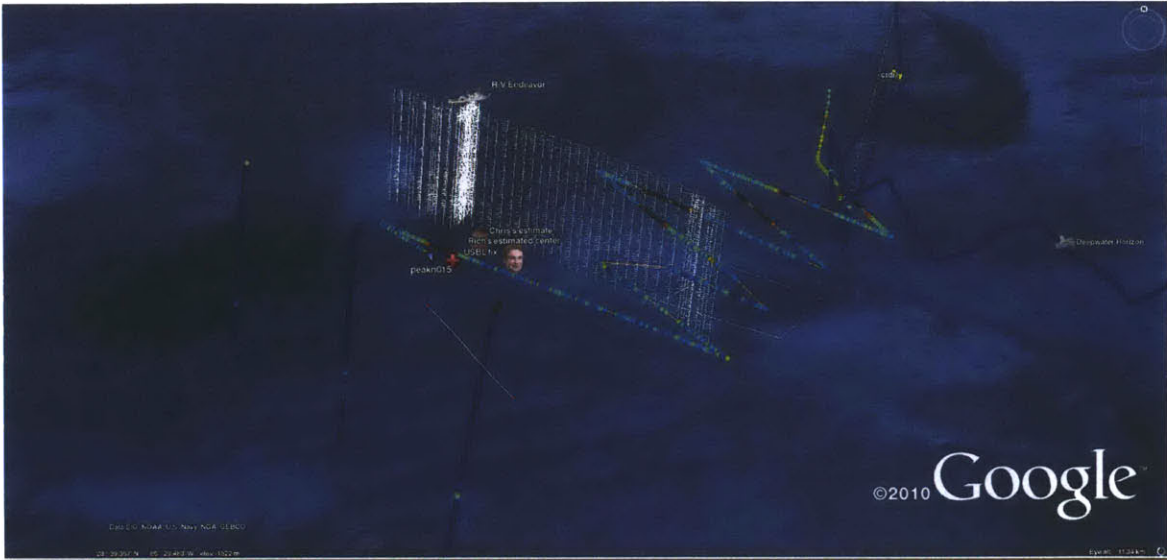
Figure 1-3: Screenshot of data communicated to the surface during an autonomous survey of the environmental impacts of the Deepwater Horizon disaster, integrated with other sources and overlayed on satellite photographs in Google Earth. White lines indicate data from the shipboard acoustic doppler current profiler. Vertical dark blue lines indicate data collected over the preceding days with CTD casts, and the zigzag located at center indicates mass spectrometer data telemetered acoustically from the AUV in real time.

study with other instruments, real-time survey modification, and provided the first visual confirmation of a coherent subsea oil plume, all while the vehicle was underwater. This capability remains rare, and this thesis extends it to imagery and high quality scalar data, while enabling multihop communication for more challenging environments.

1.3 Motivation for this work

Telemetry from AUVs historically has been limited to a small and predefined set of vehicle state information, such as the position, interspersed with the occasional scalar measurement from one or two simple sensors. This level of communication has proven adequate (if unsatisfying) when the only decision facing an operator is whether or not to abort the mission of a single vehicle. Missions, however, may now involve multiple vehicles working towards a set of goals, in dangerous and uncon-

strained environments such as under-ice, operating great distances from surface ships. There is an increasing need for human operators to have access to the data gathered by an AUV prior to any planned recovery. Specific benefits include:

New opportunities High-risk exploration missions, like the proposed Nioghalvf-jerdsfjorden glacier mission, are currently impractical given the value of an AUV and the likelihood of learning nothing from the mission. Even for more traditional missions, AUV recoveries from the open ocean are challenging and risky for both vehicles and operators. Returning scientific data prior to the end of a mission would lower the risk of failure, and thereby enable missions that are currently impractical or impossible.

Financial incentive Vehicles may take hours to ascend from missions in the deep sea. The deployment and recovery process for an AUV may take an hour each. Since a single day of sea time on a fully staffed oceanographic ship can cost $25 000, and an icebreaker upwards of $100 000 per day, maximizing the scientific return on each mission is critical. Observing a subset of the vehicle's data prior to recovery could suggest small adjustments to the current mission with potentially large payoffs, or allow planning to begin for future missions.

Improved autonomy As advanced mission executives, such as MOOS-IvP[9], T-REX[87], and DAMN[92, 93], enable complex subsea analysis of gathered data, communicating that data to the surface becomes (perhaps counter-intuitively) of greater importance. One two-year study of interactions between human operators and Zoë, a field robot deployed in the Atacama desert, found that as the level of vehicle autonomy increased over the years, users needed significantly more transparency into the robot's decision-making processes - operator questions changed from "what happened" to "why is it doing this"[114].

Components of this work offer significant benefit during more traditional AUV missions as well. The photomosaic of the World War II torpedo bomber shown in Fig. 1-4, generated by the author, consists of images captured during an AUV dive
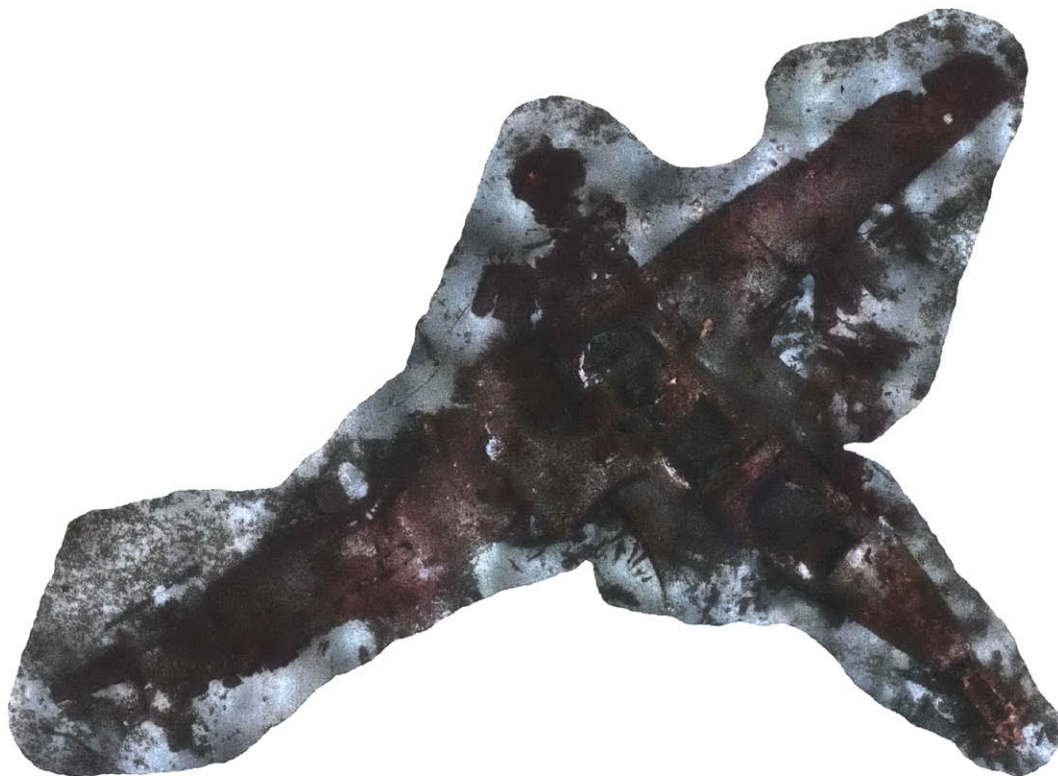
Figure 1-4: Photomosaic of the submerged wreck of an Avenger torpedo bomber, lost in the Channel Islands National Marine Sanctuary and Park. Raw images courtesy NOAA Northwest Fisheries Science Center.

in the Channel Islands National Marine Sanctuary. This dive was the third attempt to capture imagery of that site - the first dive was ruined by a faulty strobe, and the second by a disconnected cable. These types of errors, while avoidable, are difficult to completely prevent due to the complexity of AUVs. Had previews of the imagery been available during the mission, it could have been cancelled early rather than wasting expensive ship time. Relaying this data to the surface requires advances in the current state of both compression for AUV telemetry, and communication.

1.4 Wireless communication underwater

While typical land or air-based robots might communicate data to human operators using high-frequency electromagnetic signaling, such as radio modems or 802.11 "WiFi", electromagnetic radiation is quickly dispersed by water. Table 1.1

lists the current viable methods for underwater communication. Unfortunately, the freedom of AUVs to operate without a physical tether comes at a cost. Whereas tethered vehicles deliver environmental data and imagery to surface operators in real time, like the JASON II[31] or Nereus[12] Remotely Operated Vehicles (ROVs), AUV sensor data is typically inaccessible until after the vehicle has been recovered.

	Throughput (kbps)	Long Range	Free Motion
Acoustic Modem	0.01–0.5	6km ✓	✓
Acoustic Tether	1–15	6km ✓	×
Optical Modem	5 000	100m ×	✓
Physical Tether	1 000 000	12km ✓	×

Table 1.1: Viable communication options for underwater vehicles.

(a) Acoustic Modem   (b) Acoustic Tether
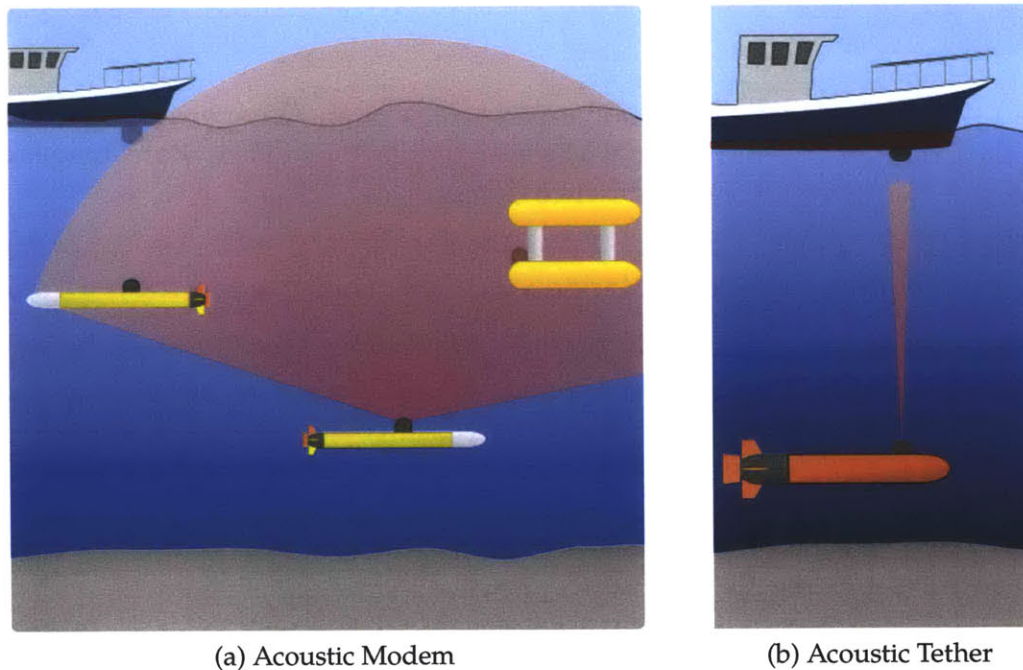
Figure 1-5: *(Left)* A vehicle equipped with an acoustic modem. Note the omni-directional beam pattern, allowing communication over great horizontal distances. *(Right)* A vehicle equipped with an acoustic tether. Note the narrow beam pattern, yielding high bandwidth but requiring vertical communication.

While there have been advances in high-bandwidth short-range (< 100m) optical modems, acoustic communication remains the only viable option for wireless

underwater communication over multiple kilometers. In addition to wireless communication methods, the recently developed Nereus[12] vehicle at WHOI spools out tens of kilometers of fine fiber-optic cable, capable of supporting communication without the weight of a traditional tether. While an exciting development, this solution still limits the vehicle to operations in the proximity of a ship, and is impractical for multiple vehicles. In addition, fiber optic tethers are currently only usable a single time, and are then discarded into the ocean as trash. Should the fiber tether be severed, the vehicle must fall back on purely acoustic communications methods.

Acoustic tethering, a specific method of communicating with an acoustic modem, relies on transceivers with narrow beams and high bandwidth, where the surface ship is located directly above the vehicle as shown in Fig. 1-5. This vertical relationship and focused transducer beam significantly limits the effects of multipath interference, drastically improving the quality of the communication link. It also imposes specific constraints on the geometry of the undersea vehicle *and* the surface ship. This is impractical for many autonomous operations, nearly impossible for those in polar environments, and does not scale easily to multiple vehicles. This thesis instead assumes the use of a relay chain of AUVs for long range horizontal communication.

1.5 Telemetry Encoding

Long-range horizontal acoustic communications underwater is largely limited to rates up to hundreds of kilobits per second[108], but low throughput is far from the only obstacle in underwater communications. The speed of sound in seawater is approximately 1.5 kilometers per second, which leads to packet latencies of several seconds. Additionally, while traditional 10/100 Ethernet breaks data into packets of up to 1500 bytes, acoustic modems may require fragmenting data into fixed-length packets as small as 32 bytes to facilitate encoding. Transmitting large chunks of data, such as imagery, requires heavily fragmenting it and reunifying

these fragments on the surface. Since any single fragment may be lost, the system must be resilient to lost fragments or provide a way to request retransmission. Given the high latencies involved, requesting retransmissions after every packet results in highly inefficient communication between the recipient and transmitter, where the majority of both nodes' time is spent waiting for packets to arrive.

Common terrestrial protocols are poorly suited for underwater use without adaptation on a number of fronts. The required header for a single User Datagram Protocol (UDP) packet (the more minimalist of the two transport protocols used by the majority of internet traffic) would alone consume three-quarters of the standard 256-bit frame used by default on the WHOI Micro-Modem. Data encoding is typically performed using the Compact Control Language (CCL), which defines a set of short messages typically needed by AUV. While there currently exist no transport or application layer protocols in widespread use for underwater vehicles, there has been significant research on higher networking layers[19]. Numerous Media Access Control (MAC) protocols, including MACA[57], MACAW[11] FAMA derivatives[37, 71] Aloha derivatives[101], and others [116] have been developed to mediate between multiple communicating nodes. Some protocols provide for single transmissions, whereas others allow long periods of time to be reserved by a vehicle for data transmission, amortizing the cost of a traditional CTS/RTS exchange over a longer transmission[17, 84]. Many recent protocols incorporate knowledge of vehicle location into the MAC process, to allow estimation of latencies[79] and tuning of transmission power[136]. Recent research has shown that modulating the power of the transmitter based on the required transmission distance is an efficient way to minimize the energy spent successfully transmitting each bit across simulated networks. In [77], a decentralized neighbor discovery protocol is presented that builds on this cross-layer design by reaching out to nearby neighbors before expending power to communicate more broadly. In practice, most AUVs rely on some form of time-based multiplexing to allow for acoustic navigation and sensing. Few field experiments have involved multiple AUVs acting as relays.

While there are no known examples of underwater systems for relaying high-

bandwidth imagery data across multiple AUV hops, there is extensive work in related areas. A survey of the relevant point to point telemetry research is in Table 1.2. The papers are grouped into four categories. Papers in this first group are the most widely used in the field, and consist of simple one-packet messageing systemes. The second group of papers are those that describe a source coding (compression) method, optimized in some way for underwater imagery, without describing any particular communication technique. Each of the methods in this second group could be used for compression in conjunction with CAPTURE, but is not a telemetry solution in and of itself. The third group of papers are those that combine source and channel coding. These approaches to transmission apply FEC in such a way that errors result in the image degrading gracefully, rather than preventing decoding of the packet. These methods are the ones used with special purpose modems, such as acoustic tether systems. They are incompatible with the general-purpose acoustic modems used by most AUVs. The final group, error-tolerant approaches to source coding, break data across multiple frames to to increase robustness to lost packets. The data is restructured such that more important segments of data have higher amounts of protection than less important segments.

A survey of relevant multihop research follows in Table 1.3. These multi-hop papers are grouped into three categories. The first category is those systems designed for data collection from static nodes. While these approaches in some of the same concerns related to MAC as more complicated multi-hop networks, the vehicle is communicating with each node independently in a point-to-point manner with no routing across multiple hops. The second class of papers are those that are designed for communicating data from AUVs, and the third class are those systems that have been implemented. I have included in this final class two significant field experiments in Delay-Tolerant Networking, one using buses as the mobile nodes, and the other using zebras.

The routing method for each of the multi-hop papers is determined to be either forwarding-based, or replication-based. Forwarding-based methods rely on strict transfer of data from one node to another node, along a single route towards

the receiver. These approaches employ various strategies to ensure the transfer from node to node, but are fragile to the loss (permanent or temporary) of a single node. Replication-based methods rely on broadcasting data to multiple receivers, employing multiple possible paths to to receiver.

	PHYSICAL LAYER	PROGRESSIVE SOURCE CODING	PAYLOAD TYPE	ROUTING
This Work	Conventional Modem	Yes - SPIHT or Other Progressive	Imagery	Yes

Scalar Telemetry Systems — *No support for data fragmentation*

	PHYSICAL LAYER	PROGRESSIVE SOURCE CODING	PAYLOAD TYPE	ROUTING
Schneider and Schmidt[98]	Conventional Modem	No - (D)CCL	Single-Packet	No
Jakuba[52]	Conventional Modem	No - (D)CCL	Single-Packet	No
Webster et al.[127]	Conventional Modem	No - (D)CCL	Single-Packet	No
Marques et al.[65], Martins et al.[67]	Conventional Modem	No - Custom	Single-Packet	No
Rajala et al.[86]	Conventional Modem	No - Custom	Single-Packet	No
Smith et al.[106]	Conventional Modem	No - Custom	Single-Packet	No

Suggested Compression Techniques — *Unimplemented, but designed for underwater telemetry*

	PHYSICAL LAYER	PROGRESSIVE SOURCE CODING	PAYLOAD TYPE	ROUTING
Li et al.[61]	Simulation	Yes - WDR-like	Imagery	No
Walker et al.[126]	Simulation	Yes - WDR-like	Imagery	No
Hoag and Ingle[46], Hoag et al.[47]	Simulation	No - Wavelet VQ	Imagery	No

Joint Source / Channel Coding — *Single-purpose point-to-point links, may impose specific node geometries*

	PHYSICAL LAYER	PROGRESSIVE SOURCE CODING	PAYLOAD TYPE	ROUTING
Beaujean and Carlson[8]	Short-Range Acoustic Tether	No - BCH	Sonar	No
Kristensen and Vestgard[58]	2kbps Acoustic Tether	No - Raw	Imagery	No
Suzuki et al.[115]	16kbps Acoustic Tether	No - DCT (256px×240px)	Imagery	No
Vall et al.[123]	OFDM In-Air Acoustic	No - MPEG4 / ERT	Video	No
Iglesias et al.[51]	Simulation	No - DT Analog Compressed Sensing	Imagery	No
Zhao and Cheng[134]	Simulation	Yes - SS-SPIHT	Imagery	No

Error Tolerant Source Coding — *Source coding augmented with unequal FEC to guard against packet loss*

	PHYSICAL LAYER	PROGRESSIVE SOURCE CODING	PAYLOAD TYPE	ROUTING
Collins et al.[23], Collins and Atkins[22]	Simulation	Yes - SPIHT with EREC	Imagery	No
Mohr et al.[69], Mohr et al.[70]	Simulation	Yes - SPIHT with ULP	Imagery	No

Table 1.2: A Survey of Point to Point AUV Telemetry Systems. While this thesis does support multihop communication, it is included at the top of the table in gray for comparison. Gray cells in the table represent shared characteristics with this thesis.

	PHYSICAL LAYER	**SOURCE CODING**	**PAYLOAD TYPE**	**ROUTING**
Data Collection — *Mobile vehicles roaming between static nodes, collecting data*				
Hollinger et al.[49],Hollinger et al.[48]	Simulation	N/A	Single-Packet	None
Dunbabin et al.[27]	Short-range optical	Custom	Single-Packet	None
Misc Protocols — *Selected protocols for communicating from AUVs*				
Zorzi et al.[137]	Simulation	N/A	N/A	Forwarding
Jones et al.[54]	Simulation	N/A	N/A	Replication
Nimbalkar and Pompili[76]	Simulation	N/A	N/A	Replication
Talavage et al.[117]	Simulation	N/A	N/A	Forwarding
Toni et al.[122]	Simulation	Progressive + UEP	Imagery	Forwarding
Implemented — *Systems which have been used in the field*				
This Work	Conventional Modem	Progressive	Imagery	Replication
Xie and Gibson[130], Rice et al.[91], Rice and Green[88], Rice and Ong[89]	Conventional Modem	Varied	Varied	Forwarding
Goel et al.[43], Haag et al.[45], Benton et al.[10], Duarte et al.[26]	Conventional Modem	Unknown	Single-Packet	Replication
Balasubramanian et al.[6]	RF on 40 Buses	None	Random Data	Replication
Juang et al.[55]	RF on Wild Zebras	Unknown	Timeseries of Positions	Replication

Table 1.3: A Survey of Multihop AUV Telemetry Systems. This thesis is included in gray for comparison. Gray cells in the table represent shared characteristics with this thesis.

I highlight four different aspects of the papers. First, I identify the physical layer described in the paper. Most experiments have been performed as software simulations, but several have been implemented for use with either high-bandwidth acoustic tether systems, or on conventional, broadcast acoustic modems. Of the cited papers that are designed for conventional acoustic modems, only one set has exhibited the capability to communicate imagery or other data more complex than basic vehicle health. Those papers describe the set of experiments performed by Benthos and the Navy Postgraduate School as part of the U.S. Navy's SeaWEB[90] program. In contrast with my work, which relies entirely on networks of free-swimming AUVs, SeaWEB relies on a dense cellular network of many fixed seafloor nodes. Vehicles in the area of the network communicated with the nearest fixed node, which then relays data back to land via a fixed routing table. Data is re-layed from fixed node to fixed node, with each attempting to immediately forward acquired data in the manner of a traditional terrestrial network. Should a node become disabled after accepting a transfer, there are no end-to-end guarantees or ways of working around the lost data.

When an end-to-end path is not immediately available, and nodes are moving relative to each other, replication of data rather than handing it off has several bene-fits, as described in greater detail in Chapter 2. This strategy, known as replication routing or store and forward routing, is used in this thesis, as well as in a set of papers describing work on the Solar AUV at AUSI. In that work, a vehicle mov-ing between two portions of a partitioned network stored transmissions until they could be delivered to the second portion of the network. These transmissions were standalone messages containing vehicle states, which could be stored without any need for ordering or fragmentation.

1.6 Organization of this thesis

This thesis begins with analysis of the need for, and characteristics of, delay-tolerant underwater multi-hop relay networks (Chapter 2). It then continues through a dis-

cussion of data compression techniques for AUVs, including novel approaches to scalar telemetry and image compression (Chapter 3). These contributions comprise key components of CAPTURE, my proposed architecture for AUV telemetry compatible with multiple contemporary AUVs. Chapter 4 lays out the overall CAPTURE architecture, and describes the integration of CAPTURE into existing vehicle platforms. Field results from three separate trials are presented in Chapter 5. Finally, Chapter 6 wraps up by discussing limitations, future work and conclusions.

Multi-Hop Relay Communication

As described in Chapter 1, underwater communication over long horizontal distance currently requires the use of acoustic modems. In this chapter, I provide an analysis of the benefits of using multiple AUV 'hops' to relay vehicle telemetry over long horizontal distances. Specifically, these benefits include increased communication efficiency and decreased power usage. I then analyze the challenges presented by communicating across such a sequence of relays, including high latencies, the lack of an instantaneous end-to-end path, and mobility of nodes. Finally, I propose an approach to relay communication tuned to the challenges and strengths of these AUV relay chains, including the presence of storage onboard the vehicle, and the necessarily small number of nodes.

2.1 Small, Multi-hop Relay Links

The ocean imposes severe limitations on acoustic communication, including low available bandwidth and long propagation delays[1, 5, 108], which lead to frequent data corruption and high latencies. These communication challenges are made worse by operating over large distances, by heavy ship traffic in the area, by strong winds and by the presence of multi-path interference. Despite these challenges, robust physical communication layers exist off-the-shelf in the form of acoustic modems from manufacturers including Linkquest, Sonardyne, Teledyne Benthos, and WHOI. To correct for bit errors during transmission, while minimizing power usage, acoustic modems typically offer a discrete set of pre-programmed Forward Error Correction (FEC) levels[85]. Users typically provide a short and fixed-length

payload to the modem, which applies FEC and includes a checksum for verification on the receiving end. This data, now with added redundancy, is modulated and transmitted via a transducer into the water. In the case of general purpose modems, FEC is applied uniformly to the transmitted message, without regard for the importance of individual bits.

For a fixed power and transmission bandwidth, the level of FEC applied prior to transmission determines the balance between throughput and reliability. When the data is received, it is demodulated and equalized before the modem attempts to decode the data. If the data has been heavily corrupted, the errors will not be entirely correctable and the checksum will not match. In this case, most acoustic modems simply discard the received data. As a result, commercially available acoustic modems present a Binary Erasure Channel (BEC) to users – packets are either successfully received, or lost. If the level of FEC is insufficient for the current channel then communication may be extremely intermittent, with long periods of no connectivity. The percentage of these transmissions which are unsuccessful is the Frame Error Rate (FER). The WHOI Micro-Modem[36], as one example, can encode its data using spreading or block codes with varying levels of redundancy. As a result, transmitted packets may range from 32 bytes to 2048 bytes.

Figure 2-1 illustrates the tradeoff between rate and reliability obtained at three levels of FEC with data acquired during a typical AUV mission near Guam. The mission was performed by the AUV Lucille. Lucille, a SeaBED-class[104] AUV operated by the NOAA Northwest Fisheries Science Center, was equipped with a WHOI Micro-Modem[36] and a 12.5 kHz ITC-3013 hemispherical transducer for acoustic communications. Messages sent using the 80bps encoding and Frequency Hopping Frequency Shift Keying (FH-FSK) modulation, in red, were received consistently throughout the dive. Messages sent with the lowest level of FEC and Quadrature Phase Shift Keying (QPSK) modulation, in blue, were received inconsistently but delivered a significantly higher instantaneous throughput. The intermediate encoding with QPSK modulation, in green, performed between the two extremes. These statistics are typical of those realized in practice - achieved throughputs from
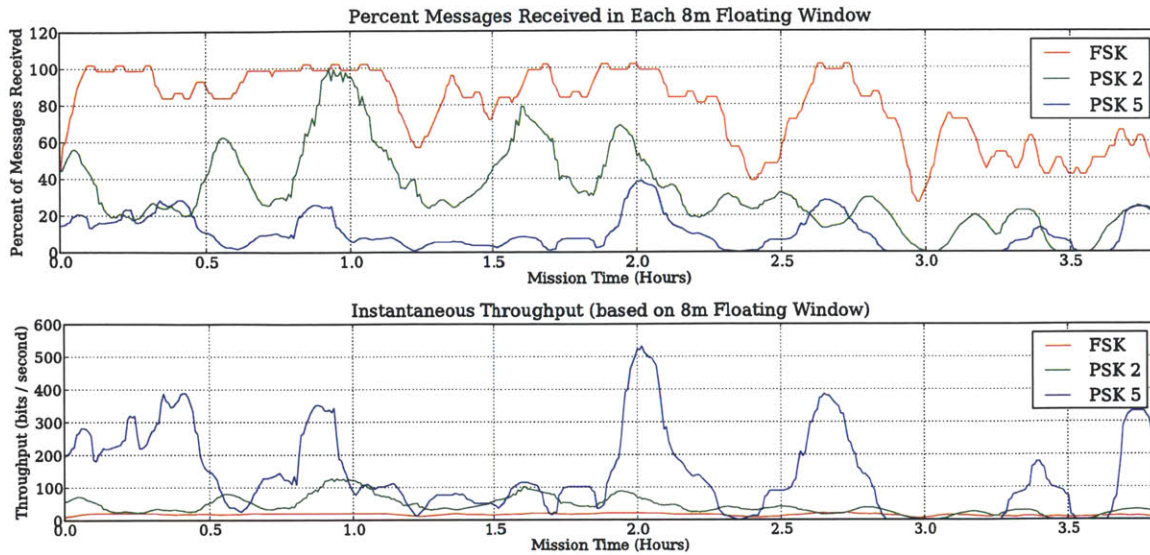
Figure 2-1: At top, the percentage of messages received in each eight-minute time period. At bottom, the instantaneous throughput in bits per second based on those percentages. Both plots have been filtered with a 5-point moving average to reduce jitter.

free-swimming AUV are commonly as low as tens or hundreds of bits per second, with long periods of disruption.

2.1.1 Frame Error Rate (FER)

Whether or not a frame is successfully received depends on the number of bits corrupted during transmission being fewer than the number that the FEC is able to correct. The number of bit errors is governed directly by the Signal to Noise Ratio (SNR) of the signal at the receiver. The number of frames that successfully get through, then, is also a function of the SNR. Fig. 2-2 shows the actual FER versus SNR during a 2010 mission of the Lucille AUV. In September of 2010 Lucille assisted in mapping the submerged portion of the San Andreas Fault off Northern California, at approximately $39°50'$N, $124°$W. During this survey, the AUV's onboard networking stack transmitted once every five seconds using QPSK and alternating levels of FEC. A particularly interesting case study of frame error rates in the vertical channel is provided by Singh et al.[105], which analyzes data obtained during a full-ocean depth experiment in the Mariana Trench.

33
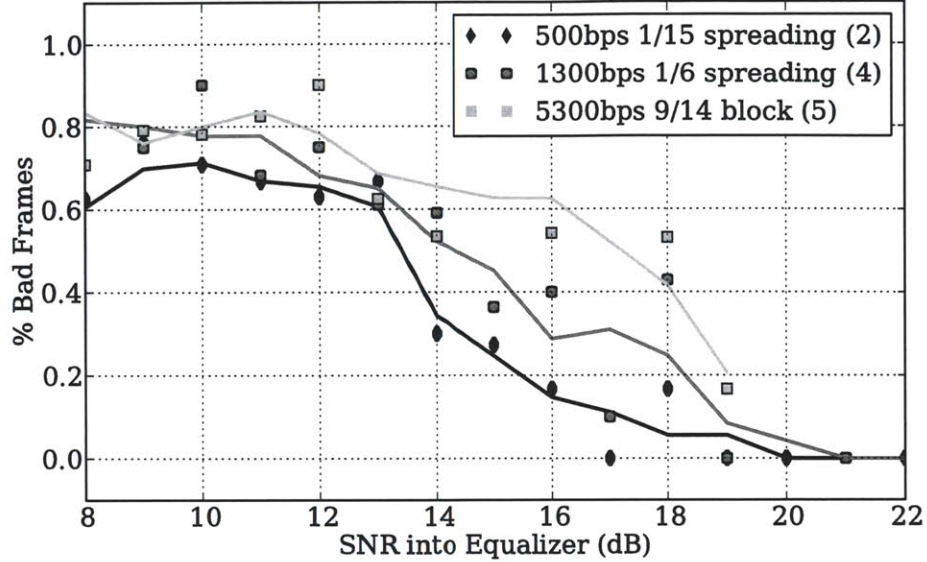
Figure 2-2: Mean Frame Error Rate versus measured SNR of detected packets during the Pacific Storm 2010 field experiment. The solid line is a Wiener-filtered 'best-fit' line to the points for each level of FEC. For very low SNR values, it is likely that many packets are simply not being detected. The packets were transmitted using 4 kHz bandwidth around a center frequency of 10 kHz.

2.1.2 Receiver SNR modelling

For a narrowband signal, the SNR is the ratio of the received signal strength to the strength of the ambient noise, as shown in (2.1), where the received signal strength is the transmission power multiplied by some attenuation due to transmission losses. Here P is the initial transmission power, A is the attenuation through the water column, N is noise level, and d and f are distance and frequency.

$$SNR(d, f) \propto \frac{P}{A(d, f)N(f)} \tag{2.1}$$

Attenuation

Following closely the derivation by Stojanović in [109], the attenuation of a narrowband acoustic signal underwater comes from absorption by water and spreading losses, as in (2.2).

34

$$A(d, f) \propto d^k \cdot a(f)^d \tag{2.2}$$

The spreading losses are independent of frequency, and represented by d^k, where d is the propagation distance and k describes the propagation geometry as spherical (2.0), or 'practical' (1.5). The absorption coefficient, $a(f)$, is dependent on the frequency. For an unobstructed path, the coefficient can be modeled using Thorp's formula[119, 13] as expressed in (2.3), where frequency (f) is in kiloHertz and $a(f)$ is in decibels per kilometer:

$$10 \log a(f) = 0.11 \frac{f^2}{1 + f^2} + 44 \frac{f^2}{4100 + f^2} + 2.75 \cdot 10^{-4} f^2 + 0.003 \tag{2.3}$$

$$a(f) \approx 0.06 f^{1.35} \tag{2.4}$$

Across the frequencies 100Hz-100kHz, which includes those used in long-range
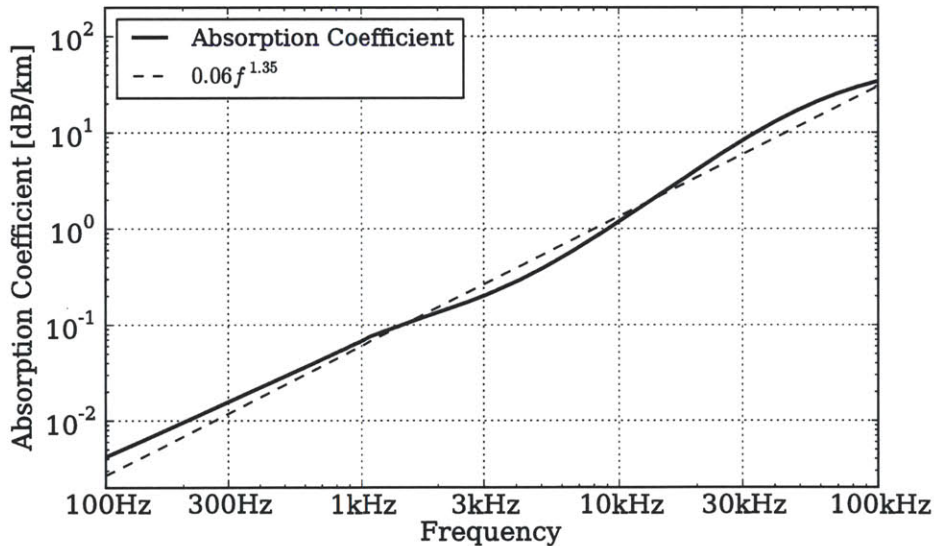
Figure 2-3: Absorption Coefficient [dB/km] versus Frequency [kHz]. The solid line presents the absorption coefficient for a range of frequencies as modeled by Thorp while the dotted line presents a numerically simpler approximation, $0.06 f^{1.35}$. Both approximations produce similar values for the absorption coefficient across 100Hz to 100kHz.

underwater acoustic communication, I have found that the absorption coefficient can be modeled by $a(f) \approx 0.06 \cdot f^{1.35}$, where f is the frequency in kiloHertz, as

shown in Fig. 2-3.

Noise

Noise in the ocean comes from four primary sources: turbulence, shipping, waves, and thermal agitation[20]. Each of these can be reasonably modeled with the approximations in Eqs. 2.5–2.8, where s represents the level of shipping traffic (0.0-1.0) and w is the wind speed in meters per second.

$$10 \log N_{turb}(f) = 17 - 30 \log f \tag{2.5}$$

$$10 \log N_{ship}(f) = 40 + 20(s - 0.5) + 26 \log f - 60 \log(f + 0.03) \tag{2.6}$$

$$10 \log N_{wind}(f) = 50 + 7.5\sqrt{w} + 20 \log f - 40 \log(f + 0.4) \tag{2.7}$$

$$10 \log N_{therm}(f) = -15 + 20 \log f \tag{2.8}$$

$$N(f) = N_{turb} + N_{ship} + N_{wind} + N_{thermal} \tag{2.9}$$

$$N(f) \approx 50 - 15 \log f \tag{2.10}$$

Across the frequency range used by acoustic communication systems, the primary variable source of noise is the surface motion of waves, driven by wind. Fig. 2-4 illustrates the total value of the noise for three different levels of wind and shipping (Eq. 2.9), along with the noise approximation of $50 - 15 \log(f)$ (Eq. 2.10) used by Stojanovic[109] and in this thesis.

AN Product

If we assume a transmitter with fixed power and recall Eq. 2.1, the variable and frequency-dependent component of the receiver SNR is simply $\frac{1}{A(d,f)N(f)}$. If we plot this quantity versus frequency for several values of distance (d), as in Fig. 2-5, clear maxima are visible. For any given distance, there is therefore a frequency which maximizes the SNR, based upon the attenuation and noise. Using the approximations from Eq. 2.10 and 2.4, we can solve for the closed-form solution shown in 2.16. This solution provides a close approximation to the AN product for frequen-
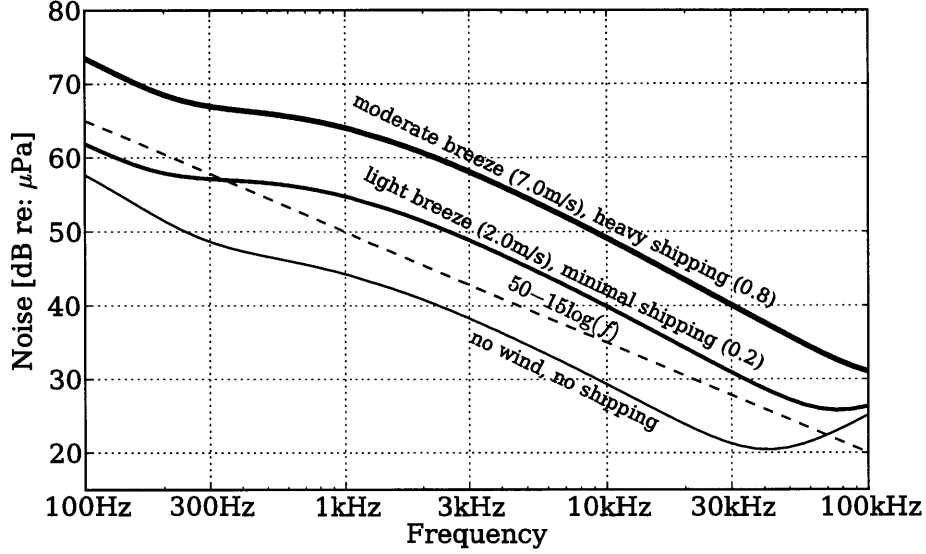
Figure 2-4: Noise Level vs. Frequency. Solid lines are used to display the noise level of three different levels of surface waves as presented in [20]: 1) no wind, no shipping; 2) light breeze, minimal shipping, and 3) moderate breeze, heavy shipping. The dotted line represents the value of $50 - 15\log(f)$, used as an approximation.

cies between 100Hz and 100kHz, where d is the distance in kilometers and f the frequency in kilohertz.

$$0 = \frac{\partial}{\partial f} \left[-10 \log A(d, f) - 10 \log N(f) \right] \tag{2.11}$$

$$0 = \frac{\partial}{\partial f} \left[-(k \cdot 10 \log d + d \cdot 10 \log a(f)) - (50 - 15 \log f) \right] \quad \text{from (2.5)} \tag{2.12}$$

$$0 = \frac{\partial}{\partial f} \left[-(k \cdot 10 \log d + d \cdot 0.06 f^{1.35}) - (50 - 15 \log f) \right] \quad \text{from (2.4)} \tag{2.13}$$

$$0 = \frac{\partial}{\partial f} \left[-d \cdot 0.06 f^{1.35} + 15 \log f \right] \tag{2.14}$$

$$d f^{1.35} = \frac{185.19}{\ln 10} \tag{2.15}$$

$$f = \left(\frac{80.425}{d} \right)^{0.741} \tag{2.16}$$

The WHOI Micro-Modem operates around a center frequency of 10, 15, or 25 kHz. Most other commercially available modems operate at similar center fre-
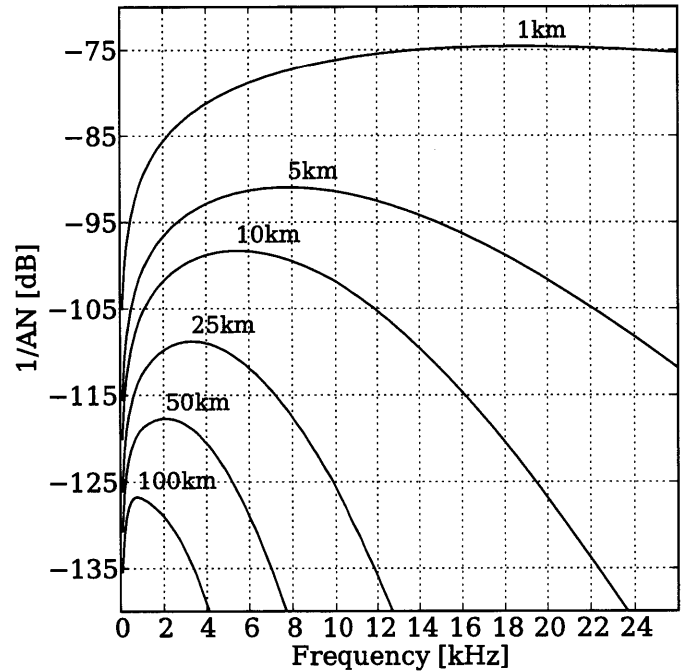
Figure 2-5: This figure depicts $\frac{1}{A(d,f)N(f)}$ for several values of d, the distance. This represents the frequency-dependent component of the Signal to Noise ratio (SNR) at the receiver. Maxima are clearly visible for each range, indicating the optimal frequency (in terms of noise and attenuation) for transmission at that range [109].

quencies. As shown in Fig. 2-6, these commercially-available modems are designed to perform optimally at distances between 1 and 5 km. If we invert the plot shown in Fig. 2-5 to generate Fig. 2-7, we can observe the performance of each frequency across a range of distances. While a 25kHz modem operates well over short (<1km) distances, the performance rapidly falls off as distance increases. While 3kHz modems have been used for long-distance underwater communication in the past, they are attenuated much more at short ranges. 10kHz modems therefore represent a good compromise for high performance over both short and long distances. For this reason, 10kHz is the frequency used by most long-range AUVs for communication.
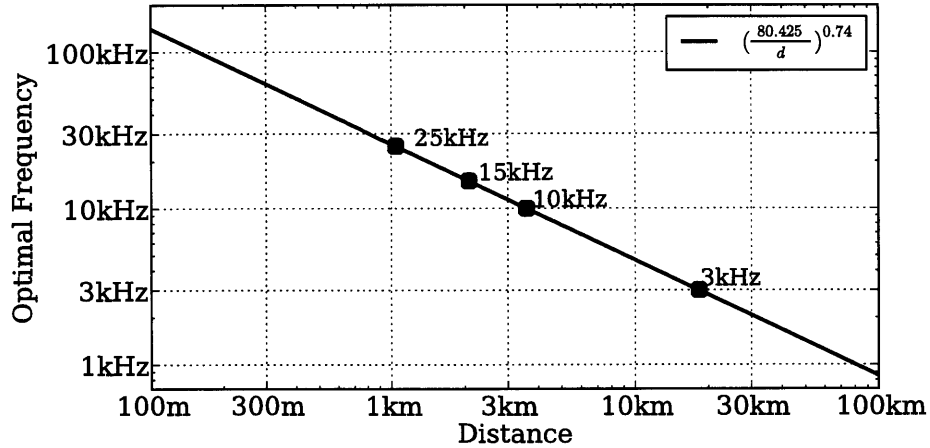
Figure 2-6: Optimal frequency vs. distance as calculated by the approximation in (2.16), $f = \left(\frac{80.425}{d}\right)^{0.741}$. Typical frequencies for AUV communication are indicated, along with the 3kHz band that offers benefits for long-range single-hop communication.

2.1.3 Motivation for a small multi-hop AUV network

Every non-decodable transmitted packet wastes, minimally, the power required to transmit it, the time to transmit it, and the power required to attempt decoding it. In addition, some form of feedback from receiver to transmitter may also be required to convey the mis-communication. As shown previously, the FER is closely related to the SNR at the receiver, which is governed by the distance between nodes, the operating frequency, and the transmission power. In particular, to achieve some FER, P_{fe}, there exists some minimum SNR so that the achieved FER is less than P_{fe}. The particular relationship between SNR and FER depends on the level and type of FEC applied, and the modulation scheme. Fig. 2-8 indicates this relationship for the five QPSK-modulated FEC levels supported by the WHOI Micro-Modem, computed assuming no Inter-Symbol Interference (ISI) and Additive White Gaussian Noise (AWGN).

To ensure transmission across a given distance, the power of the transmitter could simply be increased. This strategy does not by itself, however, provide a workable solution to communicating data from AUVs over a long distance for two reasons. First, AUVs have a limited supply of power available, relying on large
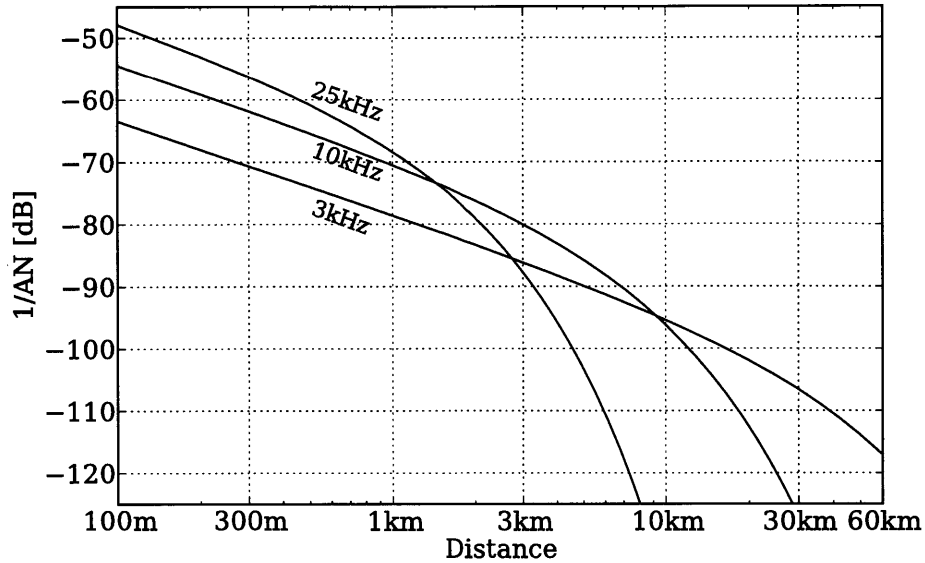
Figure 2-7: Frequency-dependent component of SNR vs. Distance for three modem frequencies. SNR is on the vertical axis, with higher SNR indicating a greater likelihood of reception. The horizontal axis displays transmission distance. Out to about a kilometer and a half, a 25kHz modem is the most efficient. Long-range performance is, however, quite poor. 10kHz performs the best all the way out to approximately 10km. 3kHz links have significantly poorer performance, nearly 10dB, over both short and medium ranges. A 10 kHz center frequency reflects a reasonable compromise between short and long-range performance.

battery packs to sustain them until recovery. Second, many acoustic modems have a fixed transmission power, which cannot be controlled underwater. In [109], the power required to achieve a fixed SNR is shown to have an exponential relationship with distance (Eq. 2.17). Similarly, for a given SNR, a channel has a certain theoretical maximum capacity. The closed-form functions below are derived for these relationships, where c, p, γ, and ψ, are constants derived from modeling in the same paper[109] and dependent upon the desired SNR.

$$C(d) = cd^{-\gamma} \tag{2.17}$$

$$P(d) = pd^{\psi} \tag{2.18}$$

In [110], the case is made that using multiple relay 'hops' to communicate data allows for more efficient power usage. I trace that argument here, and point out that
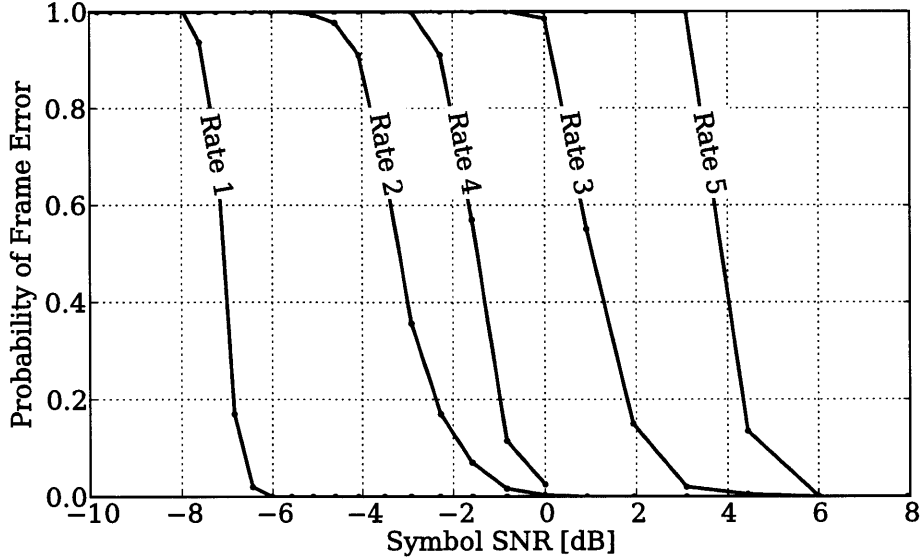
Figure 2-8: Frame Error Rate (FER) for the WHOI MicroModem versus receiver SNR, derived from simulation. Additive White Gaussian Noise (AWGN) and the absence of Inter-Symbol Interference (ISI) are assumed. Simulation results provided by Sandipa Singh, Acoustic Communications Laboratory, WHOI.

in the case of AUVs the number of relays will necessarily be small. In particular, small multi-hop relay chains of AUVs offer an appropriate solution for communication over many tens of kilometers.

In order to to transmit data as efficiently as possible, we seek to transmit messages using the minimum amount of power, while still ensuring the message is successfully received. In other words, we want to minimize the ratio of power (P) to capacity (C). Considering the case for not only a single hop but for multiple hops, the total power used by the relay network would be $n \cdot P(d/n)$, where n is the number of hops and d is the link distance. The capacity across each hop, and therefore across the entire sequence of hops, would be $C(d/n)$. A plot of energy (E) per bit $E_n(d) = \frac{n \cdot P(d/n)}{C(d/n)}$ can then be derived for a given SNR. Fig. 2-9 shows the energy per bit for a target SNR of $20dB$.

While transmission efficiency is greater with a larger number of hops, this analysis assumes that there is no cost associated with adding a single hop. By combining the transmission cost, E_b, with a fixed per-node cost, an expression for the optimum number of relay hops for a given communication distance can be obtained. In [110],
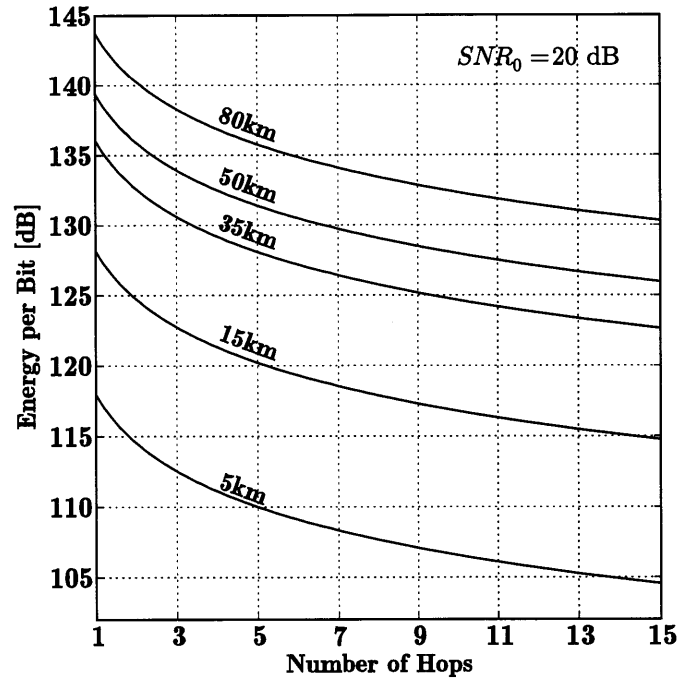
Figure 2-9: Energy per bit versus the number of hops, from [109]. The energy per bit required can be reduced by transmitting across shorter distances, or by increasing the number of hops present in the network.

an empirically determined per-hop cost of $120dB$ is used. Using this analysis, the optimal number of hops is found to be fewer than nine for ranges out to 50 kilometers. It is reasonable to expect that the per-hop cost is linear and relatively low for pre-existing, fixed seafloor nodes.

AUVs serving as hops, however, come with significant costs in practice. Each additional hop requires the purchase and deployment of another vehicle, a more complicated task for vehicles than for simple seafloor nodes. Increasing the per-hop cost by $6dB$ significantly decreases the determined number of optimal hops over longer ranges, as in Fig. 2-10a. If the vehicles must travel from the network endpoint to their relay location, as when deploying through ice, the additional energy for this deployment process may be significant and should be included in the cost function. For the case of a linear sequence of relays, that will be the inter-hop distance $\frac{distance}{nodes-1}$ multiplied by the number of hops each vehicle must travel, times some transit cost. Fig. 2-10b shows that incorporating such a cost also has the effect

42
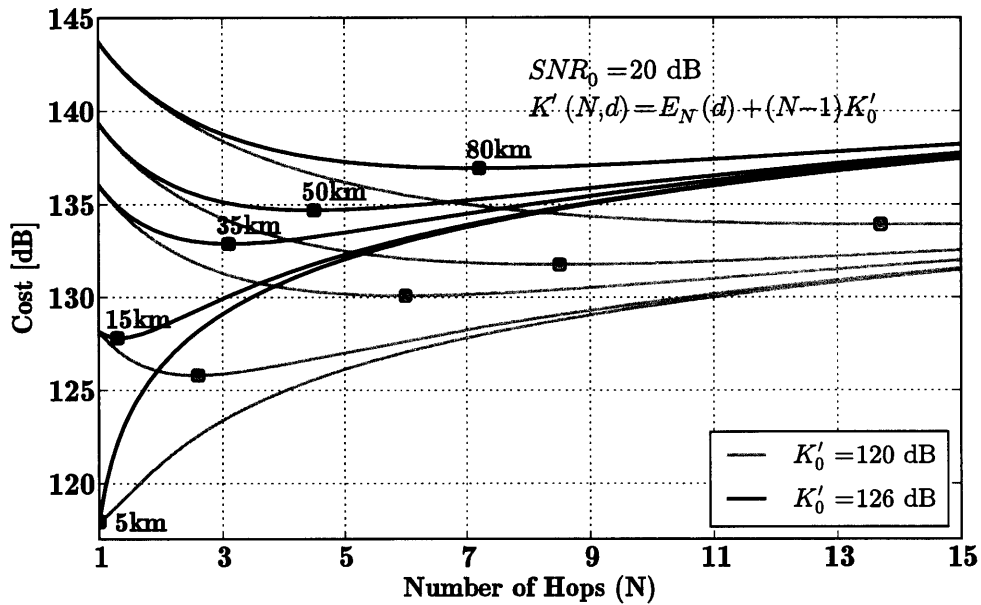
(a) Higher per-node cost
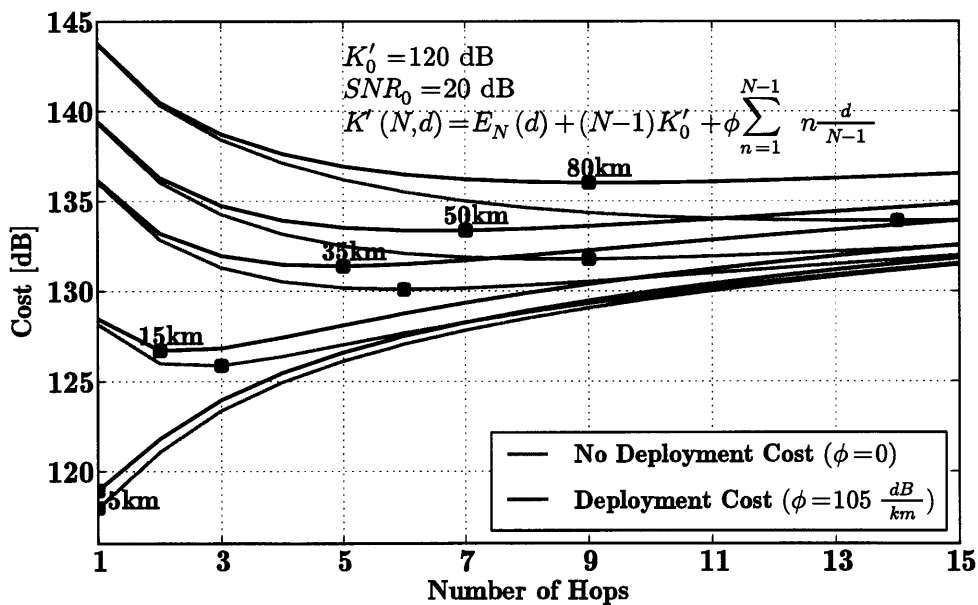
(b) Incorporated deployment cost

Figure 2-10: The optimal number of hops for communicating over five different ranges. The gray lines in both figures represent the optimal number as calculated in [110]. The top figure shows the effect of increasing the (empirical) per-node fixed cost by 6dB. The bottom figure indicates the result of introducing a deployment cost based on the distance travelled by each node.

of skewing the cost function towards a lower number of optimal hops. Adding a second, third, or fourth AUV to the water also increases the complexity of an expedition significantly, increasing the risk of losing any single vehicle. Even without establishing the actual value of the per-node cost, it becomes clear that operating in hazardous environments with significant external per-node costs will result in a lower theoretic optimum for the relay network size. While Stojanovic shows that nine hops is optimal for a link distance of nearly 50 kilometers[110], these results with alternative cost functions indicate that communication over distances of up to 80 kilometers may be optimally performed in less than nine hops. This formulation assumes that no power is spent to receive data, only to transmit. Zorzi et al[137] show that in the specific case of the WHOI Micro-Modem, the energy used when receiving packets becomes a dominant factor in total energy consumption after only a few hops. To communicate over 50 kilometers, their results suggest that only four nodes is optimal.

2.2 Relaying with CAPTURE
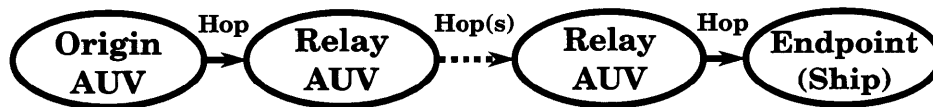
Figure 2-11: A three-hop network, labeled with the names used in this thesis.

The goal of a multi-hop relay system is to transmit data from the origin to the endpoint, across one or more 'hops', as in Fig. 2-11. As even compressed imagery data will easily dwarf the maximum transmission size of most acoustic modems, the data must be fragmented into pieces, or "segments", prior to being relayed across the multi-hop chain. It is tempting to consider this relaying as a simple extension of the single-hop case; each segment needs to be communicated to the next node in the relay chain, which is then responsible for delivering the segment further down the chain. While this approach has been used successfully by SeaWEB[90]

44

and in other networking systems, it suffers from drawbacks which we now discuss in turn.

First, the ocean is shared by all communicating nodes. Although individual vehicles may be unable to receive specific messages, underwater acoustic communication is broadcast in nature – transmitted messages will be heard by all vehicles within transmission range. This can be exploited to improve the throughput of the relay – if the origin's message is heard by the endpoint, there is no need for a relay AUV to repeat the message.

Second, if a relay AUV leaves the network, it will not be possible to forward any segments which have been successfully delivered to that vehicle but not forwarded beyond it until that vehicle re-enters the network. This is particularly unfortunate given the ever-changing underwater environment, which can frequently cause such disruptions.

Third, simple relay communication systems do not take advantage of the specific capabilities of AUVs. Unlike a seafloor node on a long-term deployment, AUV are equipped with large capacity batteries, the capability for significant computation, and large amounts of data storage. This chapter therefore proposes a relay approach better suited to the unique capabilities and limitations of AUVs. The specific characteristics of this approach are now discussed in turn.

2.2.1 Store and Forward

Terrestrial networking commonly relies on the capability to rapidly forward segments from one network node to another node, dropping those segments which cannot be immediately forwarded due to a bad link. If the segments are important, they must be retransmitted by the origin again. This is impractical in an underwater relay link, given the high probability of at least one hop failing. Early research into Disruption-Tolerant Networking[34] suggests that employing a "store and forward" approach, where relays store data until passing it off to another relay, can significantly improve the performance of high latency and intermittently-connected networks. An implementation of this strategy, the Bundle Protocol, is now being

45

pursued under the auspices of the Internet Society's Delay Tolerant Networking Research Group (DTNRG)[102]. In that work, responsibility for eventual data transmission can be conveyed from one node to another node, which is then obligated to deliver the data at all costs. The original node can then delete the stored data to free up storage.

While storage is a relevant concern on space vehicles, which may be deployed indefinitely, it is less of a concern for underwater vehicles. Relative to the bandwidth available with modern acoustic modems, AUVs can be considered to have nearly infinite storage. Ten AUVs communicating constantly at a generous throughput of 10kbps for one month would have exchanged only about thirty gigabytes, easily capable of fitting on a small and cheap flash drive. CAPTURE nodes exploit this capability mismatch by having every node in the network permanently store each piece of data that it overhears, regardless of the transmitter.

2.2.2 Broadcast Medium

In most networks, including the space networks targeted by the DTNRG, transmissions are relayed from a single network node to another single node. Underwater, all transmissions are broadcast in nature. Rather than focusing on relaying a specific segment of data from the first hop to the second hop, CAPTURE nodes track which segments are known to be possessed by any vehicle closer to the endpoint. Segments which are not known to be possessed by downstream vehicles are then transmitted. CAPTURE encodes enough metadata to uniquely identify every segment of data that it transmits. This allows any receiver to fully decode any received segment of data, regardless of whether it has previously received any information about the resource it belongs to. If a receiver is downstream from the intended recipient, it may be unnecessary for the intended recipient to ever transmit that segment.

46

2.2.3 Selective Acknowledgement

While the use of multiple relay vehicles increases the overall efficiency of an acoustic link, it also introduces additional challenges. Even the simple three-hop network shown in Fig. 2-11, if spread across twenty kilometers, would have an end-to-end latency from the origin to the endpoint of tens of seconds. Since vehicles move relative to each other, there may be long periods without an end-to-end path through the network. Were the origin to wait for confirmation that the endpoint had received a data segment before moving onto a new piece of data, end-to-end transmission would slow to a crawl. Even waiting for acknowledgement of reception from the next node in the chain halves throughput, since each transmission must take twice as long for the round-trip acknowledgement.

Selective acknowledgement is a well established technique, and has even been added to mainstream protocols like Transmission Control Protocol (TCP). Instead of transmitting an acknowledgement after every received segment of data, a single acknowledgement is sent at some future time that allows the transmitter to identify which segments were successfully received and which were not. Drawing inspiration from peer-to-peer file sharing services and the work of Wiemann et al.[128], CAPTURE nodes acknowledge not only the segments that they possess but also a list of all the segments other nodes report to have possessed, by node. This epidemic routing of the segment masks would not be practical for large numbers of nodes, but is possible for these small relay networks. Keeping track of which segments have been received locally, and which are known to be possessed by a downstream node, also aids in prioritizing segments for transmission. If a downstream node is disabled before successfully passing on segments it possesses, other nodes can easily identify which pieces remain to be relayed and fill them in. This stands in contrast to networks that hand off delivery responsibility.

2.2.4 Routing and disruption tolerance

While this thesis primarily provides an approach to returning data from a single vehicle by way of multiple relay 'hops', it is desirable that the identity of the origin vehicle be able to change over the course of a dive. In addition, relay vehicles may become disabled and unable to perform their duties. It is important, therefore, that there exists a method for specifying which vehicle is the origin, and what the sequence of vehicle 'hops' is that will convey data to the surface endpoint.

In large networks, routing tables typically proscribe the ideal path through a network from one node to any other node. In the case of a large mobile network, determining these tables presents a significant challenge. Ad-hoc routing methods designed for routing in networks of unknown connectivity such as AODV[81] would seem an ideal fit, yet high latencies make on-demand route discovery challenging. In an underwater network of AUVs, surface operators frequently have out-of-band information, such as vehicle locations and future mission plans, that may inform selection of an appropriate route. Rather than nodes attempting to learn routing information independently, I propose that surface operators are best equipped to identify which vehicle should transmit as the 'origin', and which vehicles are most appropriate to aid in relay communication. For networks of less than eight vehicles, including this data in each packet consumes a very small number of bits. CAPTURE therefore includes such information in every acknowledgement.

2.3 Comparison of Performance

To illustrate the benefits of an architecture like CAPTURE which incorporates these techniques, a set of network simulations were run. Three protocols were implemented in Python, and simulated under a variety of conditions. The first protocol is a basic node-wise acknowledgement protocol, which requires each segment to be successfully received by the next node before accepting an additional segment. This protocol would clearly be expected to perform poorly, though it has the ben-
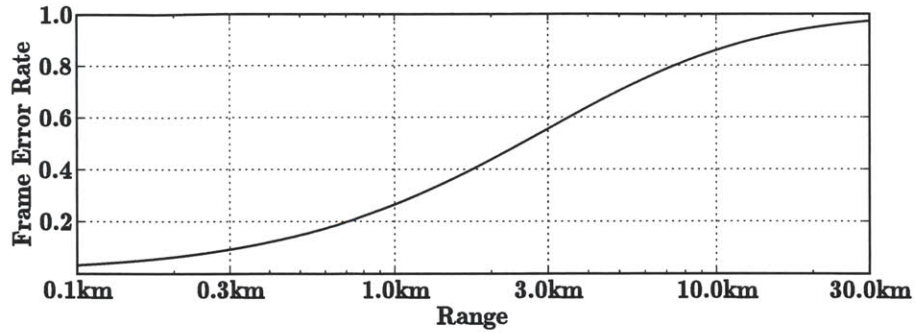
Figure 2-12: Frame Error Rate (FER) versus distance used in the simulations below.

efits of: being similar to that which is currently done in practice, being simple to implement, and using no portion of the transmissions for metadata.

The second protocol implements selective acknowledgement without the additional improvements incorporated into CAPTURE. After transmitting six data segments, the relay nodes would transmit a segment mask of all the segments they had received. The origin and endpoint transmitted only chunks and segment lists, respectively.
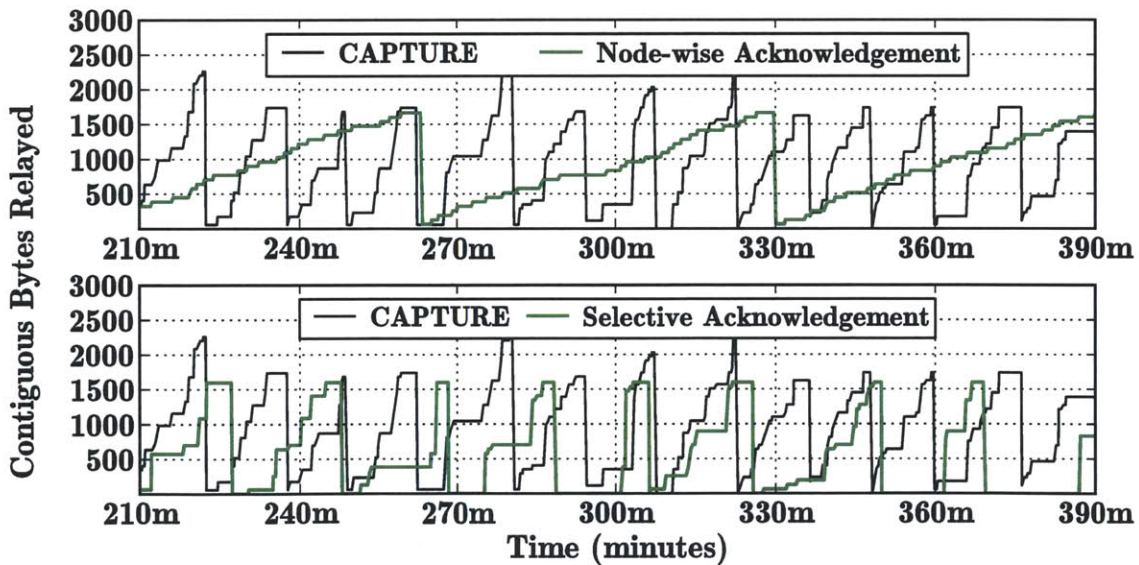
Figure 2-13: CAPTURE's performance versus simpler protocols. Note that the Y axis indicates *contiguous* bytes received, starting with the first.

For the purposes of the simulation, a fixed frame size of 64 bytes per transmis-

sion was used, along with the FER probabilities shown in 2-12. The simulated relay link consisted of four vehicles spaced two kilometers apart over six kilometers was simulated, with a fixed TDM cycle. The TDM cycle provided for the origin and two hops to each transmit three times for each single cycle of the endpoint. Since the endpoint has no data to transmit, it need not communicate as frequently. A single cycle of the CAPTURE and selective acknowledgement protocols was assumed to consume 5 seconds, whereas a single cycle of the node-wise acknowledgement protocol was assumed to take 8 seconds to account for a required immediate return acknowledgement. Ten hours of transmission were simulated, with the middle three hours shown in Fig. 2-13. After successfully transmitting a minimum of 1600 bytes, each node began transmitting a new artificial data source. For the simulation run illustrated, the number of preview images successfully received within ten hours is shown in Table 2.1.

Protocol	Previews Received
Node-wise Acknowledgement	8
Selective Acknowledgement	28
CAPTURE	40

Table 2.1: Number of preview-sized 'images' received over the course of a 10 hour simulation, consisting of a three-hop (four vehicle) network.

The simplest protocol, node-wise acknowledgement, performed quite poorly as expected. Both CAPTURE and the selective acknowledgement protocol show significant non-linearities in the progress of each image preview – these nonlinearities occur when a missing segment is received that connects a large number of received segments to the first segments. The performance of the network is closely tied to the FER for each hop in the network, which is closely tied to the length of the hop. Fig. 2-15 shows the results of running the same simulation several times for a simpler two-hop network. The x-axis is the distance between the origin and relay in the relay chain, and the y-axis represents the distance between the relay and the endpoint.
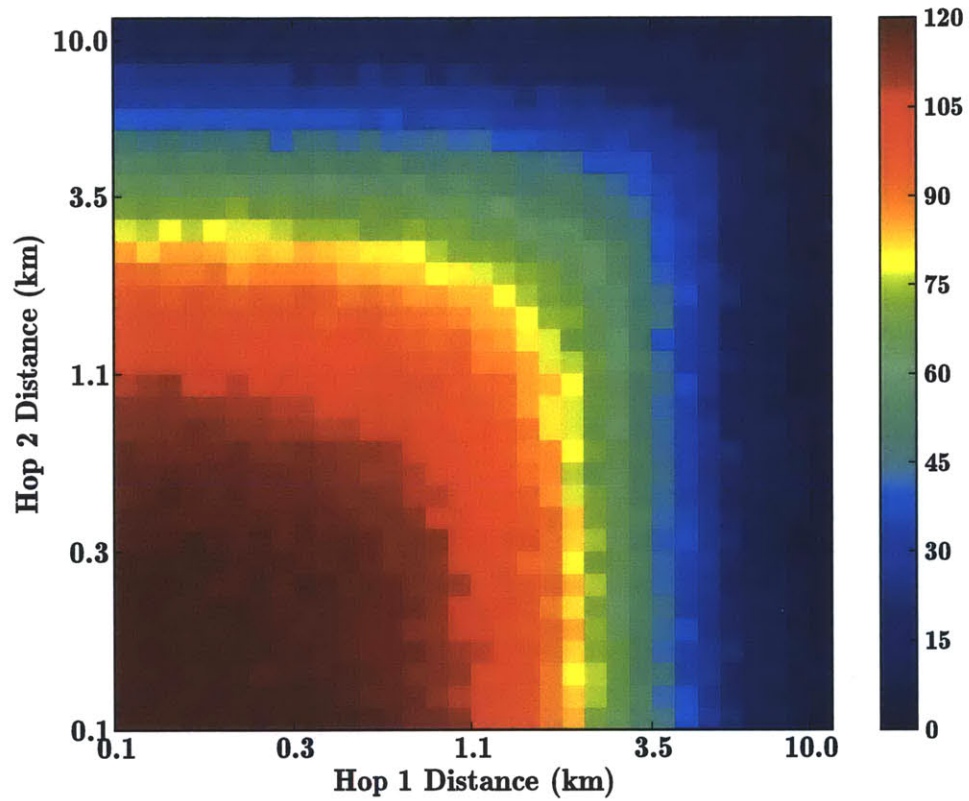
Figure 2-14: Results of simulated transmission across two hops using CAPTURE, by distance. The color represents the number of 1600 byte preview images received during a simulation of a twelve-hour mission.

(a) 2 hops, repeating until acknowledged

(b) 2 hops using selective acknowledgement

(c) log ratio

(d) log ratio
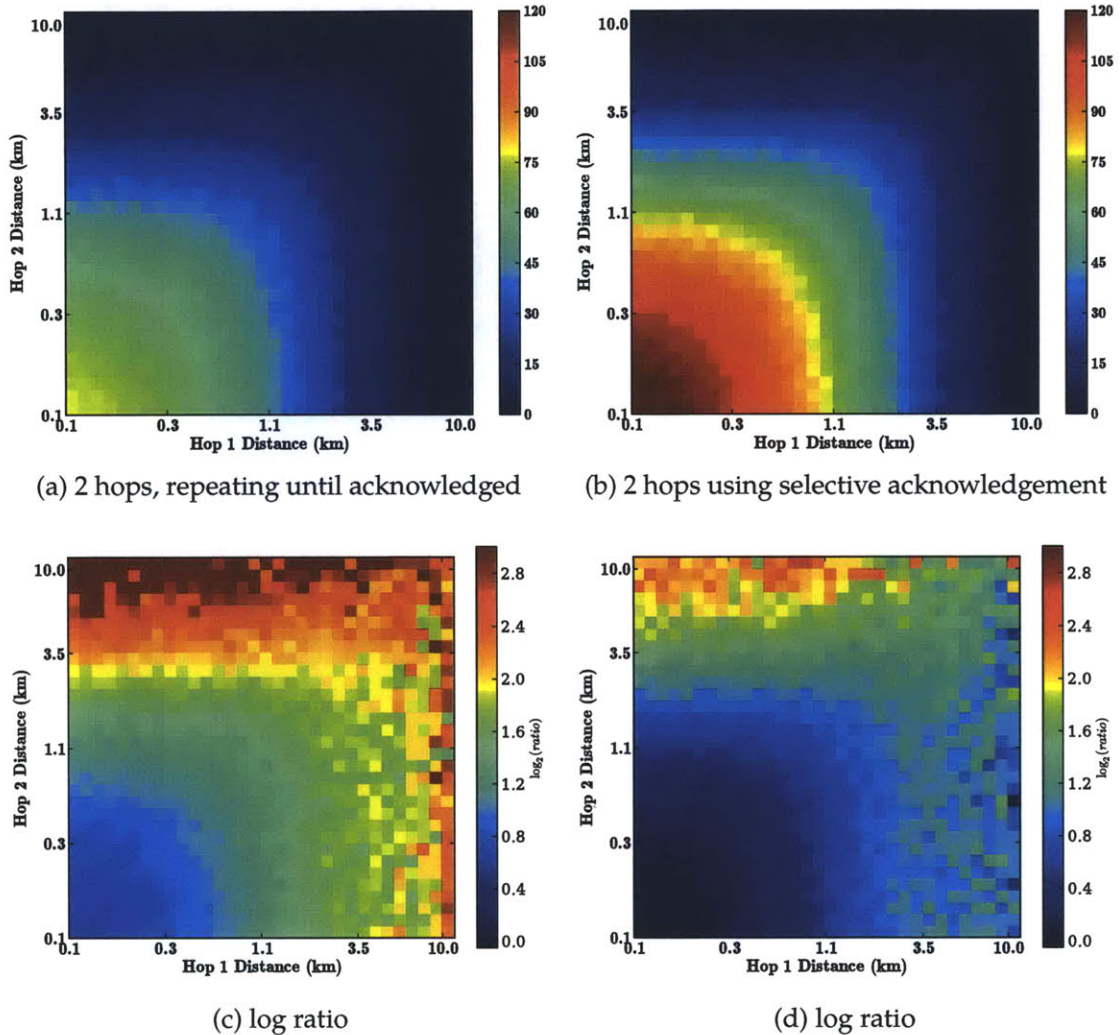
Figure 2-15: Results of simulated transmission across two hops, comparing CAPTURE and simpler protocols. After simulated transmission for twelve hours, the number of successfully received 1600 byte image thumbnails was compared between the two approaches. At bottom, the ratio of CAPTURE's performance to that of the simpler protocols are shown on a logarithmic scale. For very low probabilities of frame error ($< 5\%$, the low overhead of the selective acknowledgement protocol allows it to outperform CAPTURE by nearly 10%. Otherwise, CAPTURE significantly outperforms both alternatives.

2.4 Humans in the loop

While it is mathematically satisfying to consider metrics such as channel capacity and throughput, they do not fully capture the utility of the data transmitted to the surface. *What* is transmitted is just as important as how effectively it is transmitted. Autonomous robots are used in both exploration and emergency response. While it's no doubt possible to codify into an algorithm the appropriate search method for a submerged oil plume, there simply is not time to prepare such complicated behaviors in the wake of disasters like the Deepwater Horizon spill. Involving human operators in the selection and prioritization of telemetry increases the overall value of the telemetry just as much as increasing the throughput. In the next chapter, I present compression methods that both encode data efficiently to make use of the limited throughput, and increase the overall efficiency of AUV telemetry by incorporating user feedback.

CHAPTER 3

Data Coding

This chapter outlines two key characteristics of the telemetry compression algorithms used by CAPTURE—efficient bandwidth usage and progressive encoding—and discusses their importance to operating in underwater environments. Methods for compressing typical AUV data subject to those constraints are then presented. Results of applying these methods to both imagery and scalar environmental data are compared against current approaches using data collected during AUV missions. I propose a new technique for compression whereby wavelet coefficients are pre-scaled with a weighting function, prior to quantization. This enables transmission of greater detail in the most important areas of a signal while minimizing the number of bits used elsewhere in that signal. Wavelet compressors are highly efficient at encoding intra-image redundancy, having amongst the highest known compression ratios on single images. For images which we have prior information about, or sequences of repetitive imagery, it seems beneficial to seek an algorithm that makes use of significant inter-image redundancy as well. I present such an algorithm, which relies on texture segmentation, classification, and synthesis for image compression. We start with a brief review of the current state of the art.

3.1 Background

AUV missions primarily call for collecting two forms of data: readings from scalar environmental sensors, and sonar or optical imagery. Over the course of a dive, an AUV could easily collect one million samples of scalar environmental data, such as water temperatures or methane concentrations[16]. In addition to that data,

SeaBED AUVs capture color photographs every 3 seconds at 1360×1024 resolution, and 36 bits per pixel, from each of up to four cameras – tens of thousands of image per dive. Transmitting a single one of these images would take nearly two days at a sustained (and optimistic) rate of 300 bits per second. Getting every bit of the collected data to the surface during a mission is currently impossible.

3.1.1 Scalar Environmental Data
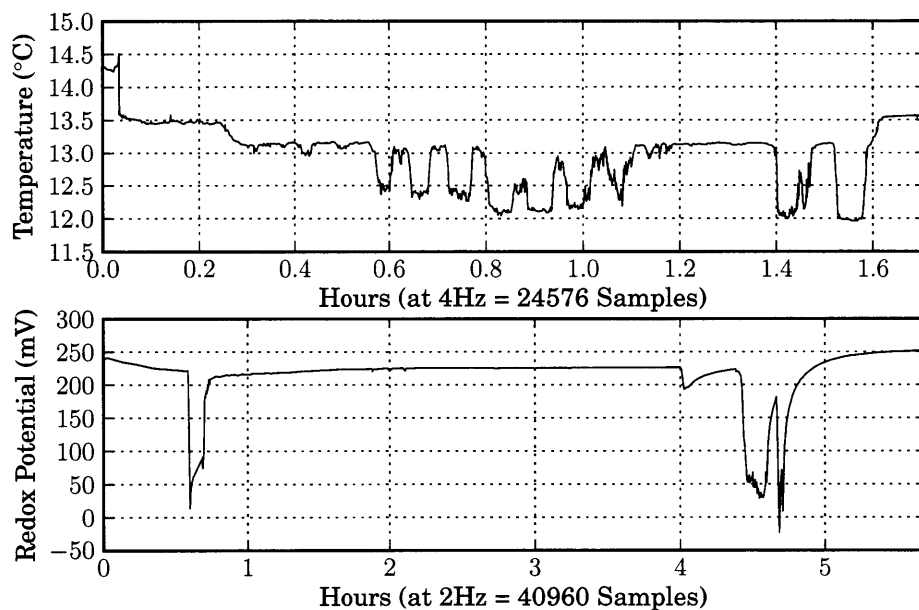
Figure 3-1: Sample scalar environmental data. Temperature data was collected over an archaeological site near Santa Barbara, California using a SeaBED AUV The reduction potential data was collected as part of the Arctic Gakkel Vent Expedition [60], and provided by Dr. Koichi Nakamura.

Modern AUVs commonly transmit a predefined set of state data to the surface, such as the vehicle position, depth, battery life, heading and similar status information. Some augment these transmissions with a small number of environmental data samples, though most environmental data is trapped on the vehicle until after recovery. Fig. 3-1 shows typical Temperature and Reduction Potential (Eh) data acquired during two SeaBED AUV dives. While temperature varies throughout the mission, Eh remains relatively constant except for brief periods of activity.

55

The diversity of AUV missions has led to a variety of custom approaches to encoding and decoding these vehicle status messages. Many of these solutions are based on the CCL[111] standard for acoustic communication, which provides a number of standard algorithms for encoding 256-bit messages containing depth, latitude, bathymetry, altitude, salinity, and other data. The messages are designed to communicate only *current* information about the vehicle. If the communication link is temporarily not functioning, no data about the vehicle state during that time would later be communicated to the surface. CCL also relies upon quantization alone to provide compression, for instance, using only 256 discrete values to heading with an encoded precision of 1.4 degrees. While this reduces the number of bytes used, it makes no use of the inherent correlation between successive heading, temperature, or salinity measurements in oversampled data. Recognizing that many of the measurements are oversampled, Eastwood et al. proposed predictive coding methods that improved the performance of CCL[28]. Schneider and Schmidt incorporate predictive coding into their recent work with Dynamic Compact Control Language (DCCL)[99], sending up a mean value followed by smaller, quantized, difference values. For time-series data with significant redundancy, such as oversampled time-series data, transform compression allows much higher efficiency.

Transform compression methods typically follow a standard pattern. First, a source coder such as the Discrete Cosine Transform (DCT) or Discrete Wavelet Transform (DWT) exploits the inherent correlation within most data, and concentrates the energy of the signal into a sparse set of coefficients. Effective source encoders concentrate most of the energy of the original signal into a smaller number of coefficients. These coefficients will no longer be correlated across different input sequences, as they can otherwise be compressed further [96]. Next, this smaller set of significant coefficients is encoded in a way that allows reconstruction of an approximation to those coefficients[94]. The process is simply reversed to decode an approximation to the original data. Interestingly, many transform compression methods can be used for both one and two dimensional data, simply by using the

appropriate form of the source coder.

3.1.2 Imagery

There has been significant development of methods for the transmission of still[115] and video[80] imagery over relatively high bandwidth (~1-10kbps) acoustic tethers operating vertically. Early efforts employed the widely used JPEG image compression standard. JPEG performs transform compression using the DCT, and a fixed quantization table for a pre-chosen quality. Craig Sayers, and others at the University of Pennsylvania, developed techniques for selecting specific frames and 'regions of interest' from a video sequence that best describe an ROV manipulator and environment state, and transmitted these regions to surface operators over a 10 kbps acoustic tether as JPEG images[97]. There are fewer examples of free-ranging AUVs telemetering imagery. In one SeaWEB[90] experiment, fixed seafloor nodes were used to relay a small number of images. Unfortunately, JPEG performs quite poorly at the high compression ratios needed for acoustic telemetry. Eastwood et al. evaluated the performance of an early wavelet-based compressor, EPIC, and found that it had benefits at low bitrates relative to JPEG[28]. In addition, there has been some previous study indicating wavelet compression techniques are particularly applicable to underwater images, video, and acoustic imagery[44, 46, 47].

Image Statistics		SPIHT	
	Synthesis		Raw
.01	1	100	10,000	1,000,000+

Bytes per Image
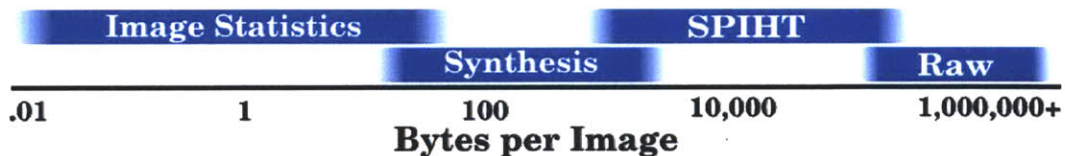
Figure 3-2: The spectrum of compression options for imagery, as developed in this thesis. Note that the options span many orders of magnitude.

To compress imagery to a size appropriate for acoustic transmission, a few hundred to few thousand bytes, requires very high compression ratios. While a typical compression ratio for a JPEG intended for human-viewing might be 10:1 or 30:1, converting a one megapixel color image to a few kilobytes implies a compression ratio of 1000:1 to 3000:1 – two orders of magnitude higher. This necessitates the

analysis and use of less common compression techniques. In this chapter, I present a range of options for communicating imagery, ranging from summarizing an entire dataset as a time series to encoding individual images using wavelet compression, as shown in Fig. 3-2. I also present a novel method for image compression based on texture synthesis and texture classification. This method, nicknamed Image Synthesis, fits between transmission of individual images and summary dataset statistics.

3.1.3 Discrete Wavelet Transform (DWT)

Transform compression using the DWT as a source coder, typically referred to as wavelet compression, has been found effective on a variety of real-world signals and imagery[14]. The DWT, a linear transform, is now widely used as a source encoder for imagery and biomedical data. The DWT is calculated by applying a low-pass filter to the input signal, generating one set of coefficients, and then applying a high-pass filter to the input signal to generate a second set of coefficients. Both sets of coefficients are downsampled by two, resulting in the same number of coefficients as the original input signal had samples. Calculating the DWT of a signal thus results in two distinct sets of coefficients; a decimated version of the signal known as the 'approximation coefficients', and a set of 'detail coefficients' which contain the higher-frequency information lost during decimation. Fig. 3-3 shows the full wavelet decomposition of a short one-dimensional signal of 32 samples.

The DWT is typically (as in Fig. 3-3) applied recursively to the approximation coefficients, generating several levels of detail coefficients; each level of detail coefficients then represents the detail lost by decimation at that iteration of the transform. Each detail coefficient in the resulting set is localized in time as well as being associated with a 'scale', or level of detail. The detail coefficients will generally be low in magnitude, except near areas of change for a given scale. This sparsity facilitates efficiently compressing the data. For a well-written introduction to wavelets, DeVore and Lucier provide an excellent reference[24]. As the DWT is separable, multi-dimensional data can be transformed a single dimension at a time, following

58

Level Coefficients	Level Contribution	Cumulative Reconstruction
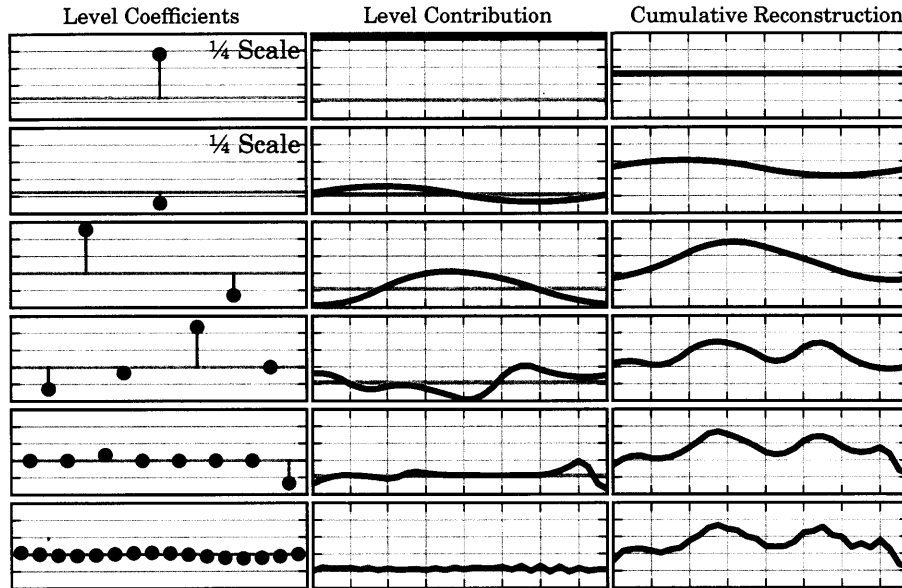
Figure 3-3: Wavelet coefficient magnitude is shown by the stem plots at left. The middle column indicates the sum of the inverse transformed wavelets at that level of detail. By cumulatively summing the levels (right column), increasingly detailed approximations to the original signal are produced until the original signal is recovered at the bottom right.

the same procedure.

3.1.4 Embedded Wavelet Coding

Progressive coding methods allow the reconstruction of intermediate data representations at one or more 'checkpoints' within an encoded data stream. *Fully Embedded* coding methods have the additional property that they do not require targetting any specific image 'quality' or final size. Specifically, if data is compressed twice with a fully embedded encoder, to sizes M and N, with $M > N$, then the first N bits are identical in both files. This makes fully embedded coding methods well suited to the underwater environment where computation ability is limited, communication is packetized, and transmission rates can vary from packet to packet, as it allows compression to be performed independent of the target transmission rate. Messages sent to nearby AUVs for multiple vehicle collaboration could be sent at a

higher rate, and those destined for a surface ship or transmission over longer distances can be sent at a more conservative rate without any need for recompression of the data. Low-fidelity color image thumbnails, transmitted at rates as low as a few hundred bits per image, can later be used as a basis for more refined versions of the same image. If the entire bitstream is sent, the compression process is entirely reversible and results in the original data with no loss of precision. Combined with the success of wavelet-based analysis techniques in the underwater domain, this suggests underwater AUV networking can greatly benefit from the use of fully embedded wavelet compression.

The Embedded Zerotree of Wavelets (EZW)[103] algorithm is one early example, which led to the more efficient Set Partitioning in Hierarchical Trees (SPIHT)[95] coding method, and derivatives[120, 125]. Each of these compression algorithms follows a similar process of three main steps. First, the DWT is applied to the data, resulting in a set of coefficients in the wavelet domain. Second, these (typically floating-point) coefficients are requantized as signed fixed point numbers. Finally, this fixed-point representation is encoded in an algorithm-specific way, which results in a sequence of bits. Any truncated portion of this bitstream can be decoded into a signed fixed-point approximation to the wavelet coefficients, after which the Inverse DWT restores an approximation to the original data. Each algorithm can be used effectively on scalar data, imagery, or even 3D volumetric data. For clarity, I discuss the one dimensional approach first, and then extend to imagery.

3.1.5 Image Synthesis using Texture Patches

Natural photographs exhibit intra-image redundancy, including smooth-varying colors and luminance. Embedded wavelet coding relies on this redundancy to transmit a facsimile of the image in fewer bytes than the uncompressed image would consume. With respect to its use for underwater image telemetry, there are two aspects of embedded wavelet compression methods that merit closer scrutiny. First, fine texture details are quickly lost at high compression ratios, due to the smoothing effects of the wavelet compression. This is undesirable if the informa-

tion surface operators hope to extract relies on the fine-scale texture of the image. Brown rocks and brown coral appear quite similar after smoothing out fine-scale details, for example, yet differentiating between classes may be important to surface operators. Second, when taking multiple photographs of a single area, there will be significant redundancy not only within each image, but across the set of images. Wavelet compressors are highly efficient at encoding intra-image redundancy, yet make no use of this inter-image redundancy.

Video compression techniques do make use of recurrence across frames, yet they do so in a time-localized manner. Rather than considering every frame previously recorded, they consider only those frames within immediate time proximity, and assume that motion is a dominant cause of inter-frame changes. For example, if a video began recording an outdoor scene, then moved indoors to a different scene, then returned to the same outdoor scene again, the compression of each outdoor scene would be completely independent. Compression of the second outdoor scene would not take advantage of the fact that this set of imagery is highly redundant of the first outdoor scene. Each video image is only compressed relative to the similar images in an adjacent time period.

When AUVs are compressing sequences of repetitive static imagery, the overlap of sequential frames may be low or non-existent, limiting the utility of motion-compensation as a compression technique. However, prior information may be available about the contents of the images in terms of texture, even though each individual image may be of different time periods and locations and thus vary significantly. Even though a single image may not look very similar to the previous image overall, the image may feature textures and objects very similar to those seen in prior images. Section 3.3 presents a compression option, nicknamed Image Synthesis, which utilizes the inter-frame redundancy of underwater data to provide extremely high compression while preserving some texture information.

↑	16b		20b	↓	24b		28b		32b
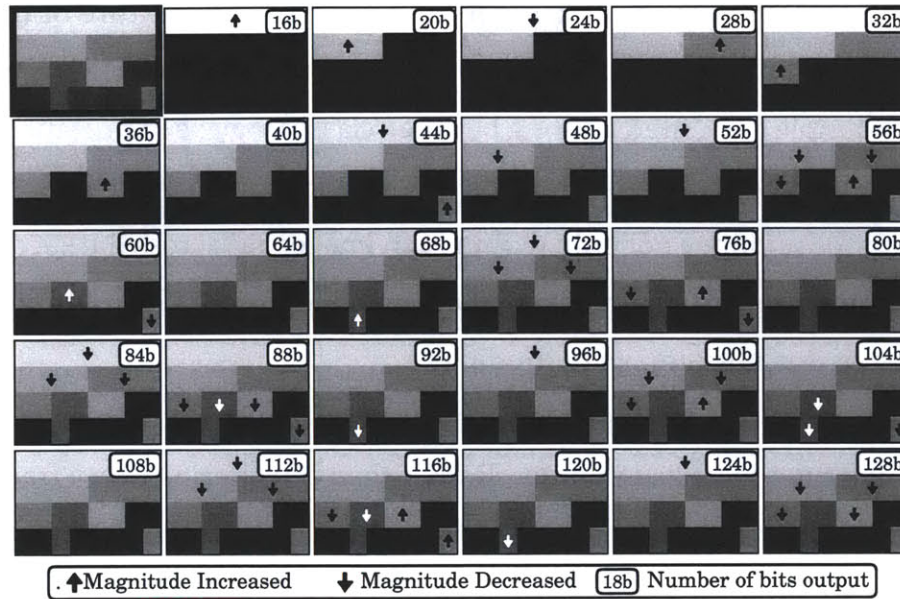
Figure 3-4: A wavelet decomposition at upper left, followed by the reconstruction from increasingly length SPIHT bitstreams. As the number of bits grows, the reconstruction is closer to the original coefficients. Coefficient signs have not been have been depicted.

3.2 Fully Embedded Wavelet Coding

SPIHT, its progenitor EZW[103], and similar algorithms, treat the wavelet decomposition as a tree of coefficients, rooted at the lowest level approximation coefficients. Many real signals that have large magnitude coefficients at high levels also have higher magnitude coefficients at lower levels. Fully embedded wavelet coders exploit this cross-level correlation. SPIHT does this via a clever sorting algorithm. As the authors write in their tutorial on the topic[78, p95], set partition coding

> ...is a procedure that recursively splits groups of [coefficients] guided by a sequence of threshold tests, producing groups of elements whose magnitudes are between two known thresholds.

A SPIHT-encoded bitstream consists of a sequence of refinement bits and sorting bits, interlaced in a data-dependent order. Sorting bits provide an efficient way to identify high magnitude, and therefore important, wavelet coefficients. Refinement bits provide a continually improving estimate for the magnitude of a wavelet

coefficient. In particular, sorting bits indicate:

- whether a coefficient is greater in magnitude than the current threshold, or 'significant',

- whether any descendant in the wavelet tree of the currently considered coefficient is 'significant', and

- whether any grand-descendant is significant.

Refinement bits indicate either the sign of a coefficient, or a single bit of a coefficient's magnitude.

Refinement bits provide a continually improving estimate for the magnitude of a wavelet coefficient. Sorting bits provide an efficient way to identify high magnitude, and therefore important, wavelet coefficients. Fig. 3-4 shows the progressive reconstruction of a small set of coefficients using an increasing number of (indicated) bits.

3.2.1 Scalar Environmental Data

Fig. 3-5 and 3-6 display respective approximations for the original scalar temperature and Eh data of Fig. 3-1 using an example fully-embedded wavelet coder, SPIHT, compared to the more traditional approach of interpolating quantized samples. These two coders are compared for each signal at three different encoding sizes: 28 bytes, 56 bytes, and 112 bytes. Paying particular attention to the extrema of each signal, the SPIHT encoded signals clearly better represent both original signals than the spline-interpolation at all three byte sizes. An additional side-effect of the full-embedded wavelet coding is that the reconstructed signal has been de-noised; discarding low-magnitude coefficients is an effective form of noise reduction [124].

In order to quantify the benefits of encoding the temperation and Eh scalar data with SPIHT, Fig. 3-7 displays the root mean squared error (RMS error) versus signal size for SPIHT and two interpolation coding methods. Here it can be seen that SPIHT displays significant improvement in data fidelity across a wide range of

14 Spline-Interpolated 16-bit Fixed Point Samples

SPIHT Encoded with 28 Bytes

28 Spline-Interpolated 16-bit Fixed Point Samples

SPIHT Encoded with 56 Bytes

56 Spline-Interpolated 16-bit Fixed Point Samples

SPIHT Encoded with 112 Bytes

Figure 3-5: SPIHT-encoded scalar temperature data at different levels of compression, compared to interpolating quantized samples. The original data is shown in gray in each graph while the black lines represent the various approximations. The approximations are grouped into three sets of SPIHT vs. fixed-point comparisons where each pair is encoded with the same number of bytes.

14 Spline-Interpolated 16-bit Fixed Point Samples

SPIHT Encoded with 28 Bytes

28 Spline-Interpolated 16-bit Fixed Point Samples

SPIHT Encoded with 56 Bytes

56 Spline-Interpolated 16-bit Fixed Point Samples

SPIHT Encoded with 112 Bytes
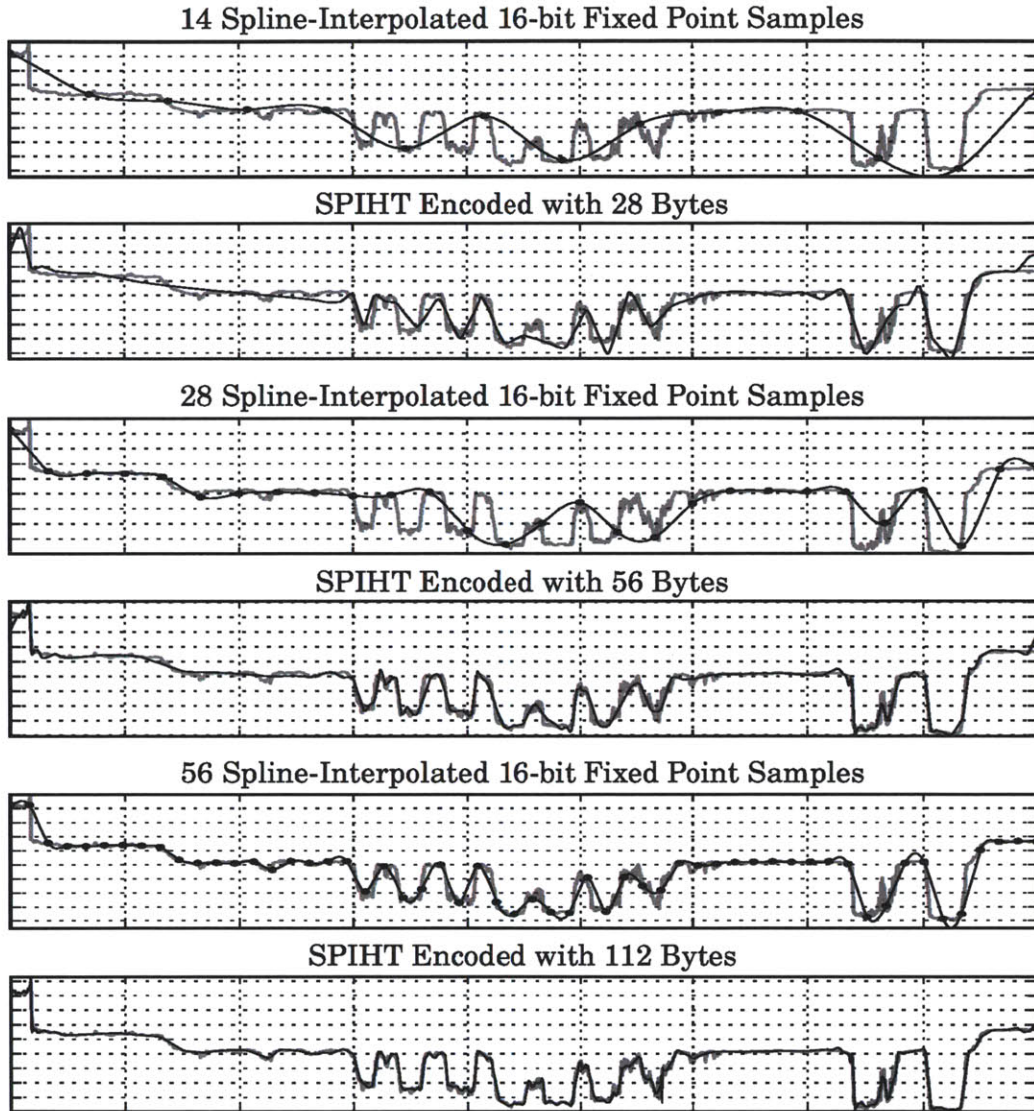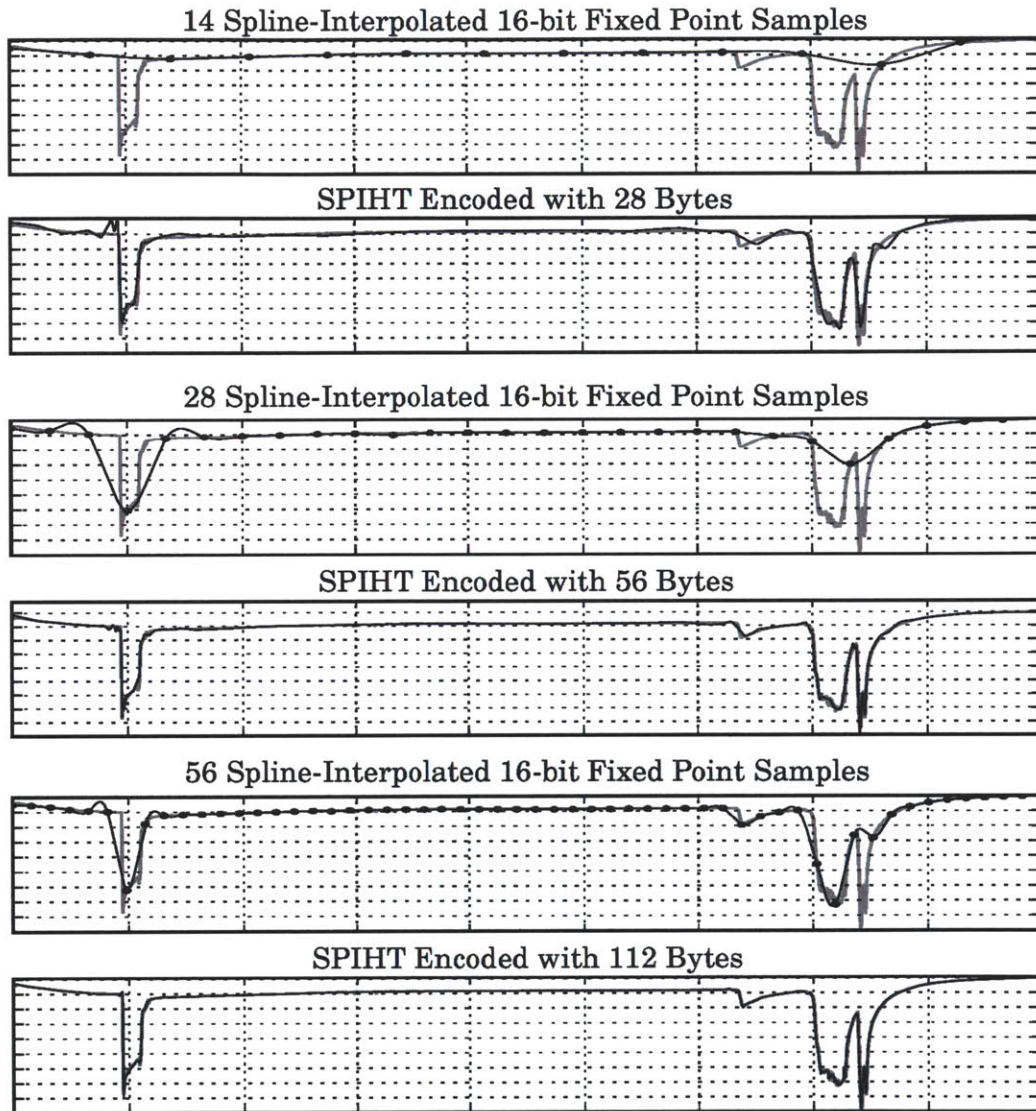
Figure 3-6: SPIHT-encoded scalar Eh data at different levels of compression, compared to interpolating quantized samples. The same original data is shown in gray in each graph while the black lines represent the various approximations. The approximations are grouped into three sets of SPIHT vs. fixed-point comparisons where each pair is encoded with the same number of bytes.
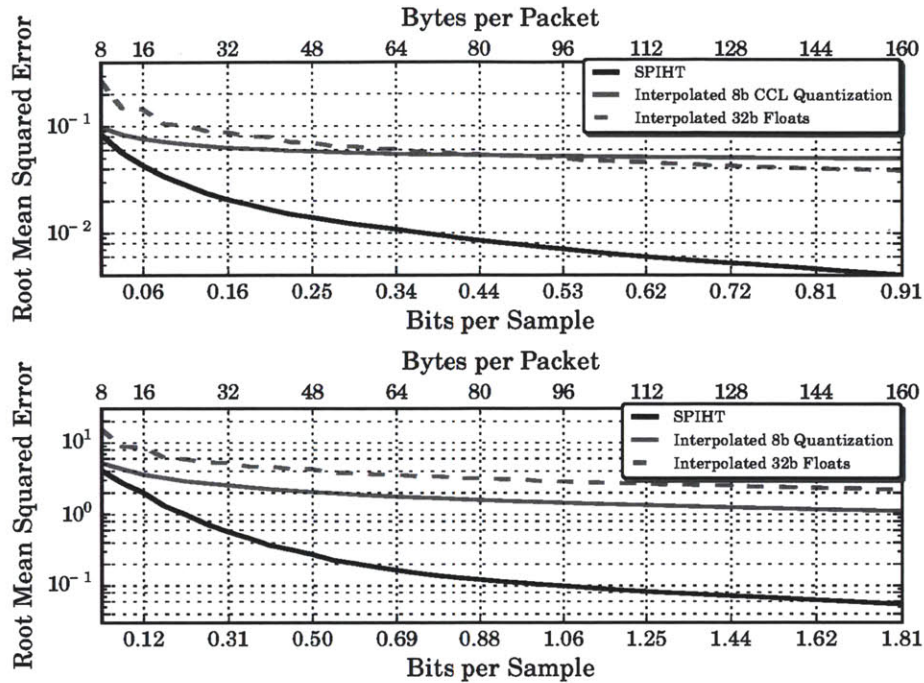
Figure 3-7: Comparison of SPIHT encoding with subsampling methods for Temperature data (top) and reduction potential data (bottom) across a wide range of encoding qualities.

transmission rates, when compared to simple subsampling. The received signal is both qualitatively (Fig. 3-5 and 3-6), and quantitatively (Fig. 3-7) more similar to the original data than the interpolated data points.

3.2.2 Segmenting Scalar Data

While a single image is easy to consider as a distinct 'resource', transmitting environmental sensor data requires identifying a section of data to transmit. This is best done by breaking a time-series into large chunks of data – for correlated time-series data, compressing a few samples at a time is much less efficient than compressing long sequences simultaneously. Fig. 3-8 shows this result while piecewise compressing a long series of temperature data.

66
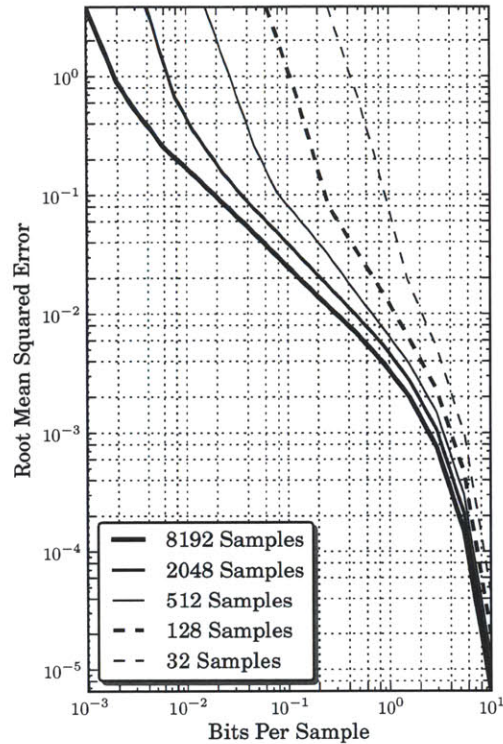
Figure 3-8: This figure shows the reconstruction error (Y axis, in RMSE) versus the compression level (X axis, in bits) for the two hour sequence of temperature data. Each plotted line shows the result of compressing the full dataset, but doing so by different length subsets of the data at a time. Since the original temperature data was collected at four Hertz, compressing 8192 samples at a time would be equivalent to transmitting updated data every 34 minutes, versus every 30 seconds when data is compressed 128 samples at a time. Encoding more samples in each transmission lowers the reconstruction error for any given compression level.

3.2.3 Spatially Varied Quantization

Prior to being coded by the embedded wavelet coders described in this chapter, wavelet coefficients are requantized into a standard sign-magnitude representation. While the level and method of quantization depend on the dynamic range of the time-series data, the quantization is typically constant for all wavelet coefficients.

Occasionally, it may be of value to provide higher fidelity to certain sections of data. Accenting recent data would allow decisions to be made about nearby
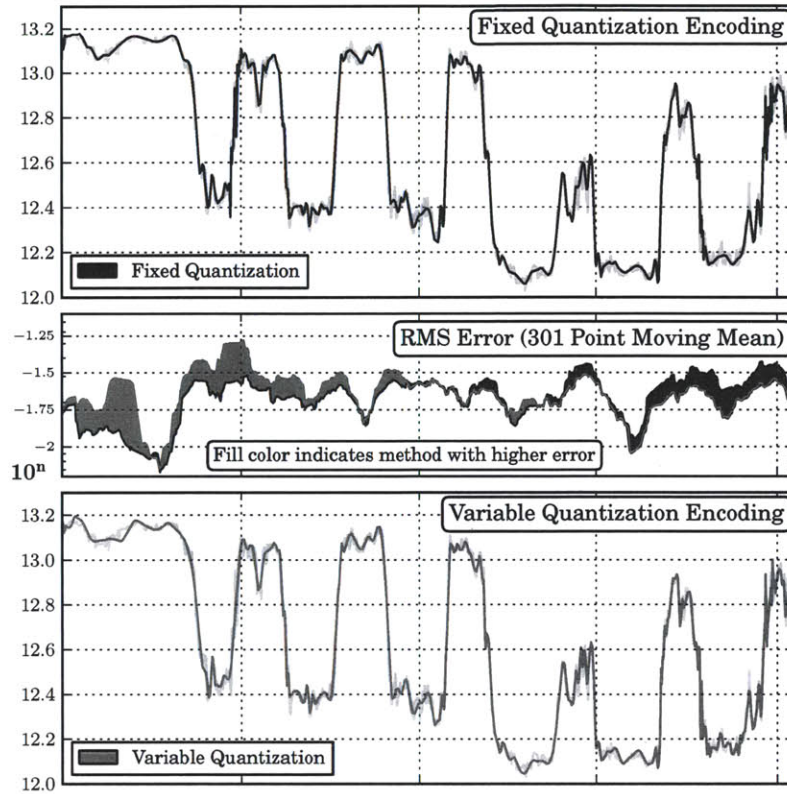
Figure 3-9: Standard SPIHT compared to time-varying quantization. Note that time-varying quantization performs better on the more recent data.

features of interest before they are left far behind. Images may have one or more regions of interest that warrant a higher quality encoding. I propose that this can be achieved by artificially pre-scaling the wavelet coefficients using a weighting function, prior to quantization. As wavelet coders prioritizes higher magnitude coefficients, this leads to greater detail being conveyed in those areas of the reconstructed signal at the cost of lower detail elsewhere. The receiver must also know the cost function so that the inverse weighting can be applied after decoding wavelet coefficients. This strategy has been employed to generate Fig. 3-9; wavelet magnitudes were artificially prescaled prior to encoding them with SPIHT.

$$
c_n = \begin{cases} 2^{18} & n = 1 \\ 2^{\left\{16 + \frac{2x}{n-1} \,:\, x \in \mathbb{N}_0, \, x < n\right\}} & n > 1 \end{cases} \tag{3.1}
$$

The weighting function used in Fig. 3-9 was generated using the logarithmically increasing sequence of quantization coefficients shown in Equation 3.1, where n is the number of coefficients. This results in more recent data being encoded with higher fidelity than older data. In the case of an image, the coordinates and scale for regions of interest could be transmitted along with the encoded image data, which would be used to derive the applied weighting function.

3.2.4 Imagery

(a) Pillow Lava (Southern Mid-Atlantic Ridge)
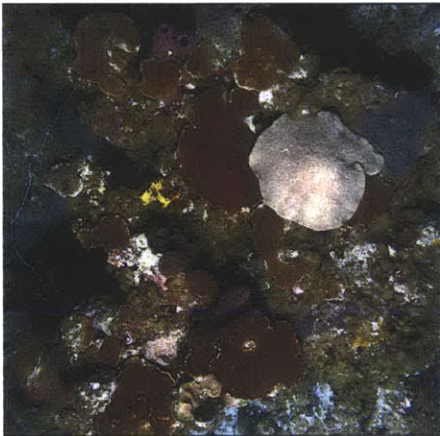
(b) Coral Reef (Puerto Rico)
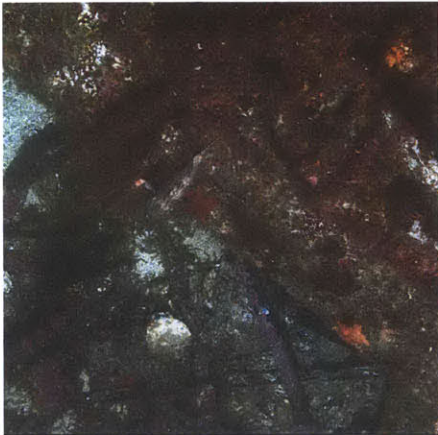
(c) Airplane (Santa Barbara, CA)

(d) Fish and Sand (Santa Barbara, CA)

Figure 3-10: Representative imagery, captured by the SeaBED AUV. The four images shown are used to illustrate the performance of SPIHT on typical underwater imagery, relative to the current state of the art.

SPIHT was originally designed for photo compression, and can be used on high dimensional datasets as well as scalar data. Two-dimensional data like imagery is simply transformed with the 2D form of the DWT, and then SPIHT coded following a similar process as the 1D version. To encode color images, each color plane is encoded independently. As humans are more sensitive to changes in luminosity than changes in chromaticity, encoding in either the YUV or Lab color space simplifies allocating bits to the most important data, with the U and V color planes being encoded with a much smaller allowance than the luminance plane.

Fig. 3-10 displays four images, captured by the SeaBED AUV, which are representative of the types of images desired by human operators during an AUV mission. These images were resampled to 1024×1024 pixel source images and then coded using SPIHT, JPEG 2000, and progressive JPEG at three different levels of quality. Fig. 3-11 displays the reconstruction error versus number of bytes for each of the images using each coder. The JPEG 2000 data has visible nonlinearities indicating discrete quality 'checkpoints,' while the errors associated with SPIHT follow a smooth reduction curve as the size of the transmitted file increases. We also see from these graphs that JPEG is largely incapable of encoding large images at the low sizes available through SPIHT.

Fig. 3-12 displays the same metrics as Fig. 3-11 but encodes versions of the images in Fig. 3-10 that have been resampled to a smaller size of 256×256 pixels. Again, there are visible discrete quality 'checkpoints' evident in the progressively coded JPEG 2000 data, while SPIHT coding provides a smooth compression curve. JPEG performs adequately at larger filesizes; however the error increases substantially at lower file sizes reached. It was hypothesized that the lower quality (smaller q) JPEG coders would perform better than the higher quality JPEG coders at lower numbers of bits. However, somewhat counter-intuitively, the higher-quality JPEG compression coders typically resulted in better image quality (less RMS Error) even for small target filesizes. JPEG is simply not suited to compression at these ratios.

The mean reconstruction error across all four images of Fig. 3-10 is displayed in Fig. 3-13. The solid lines correspond to the 1024×1024 pixel source images of
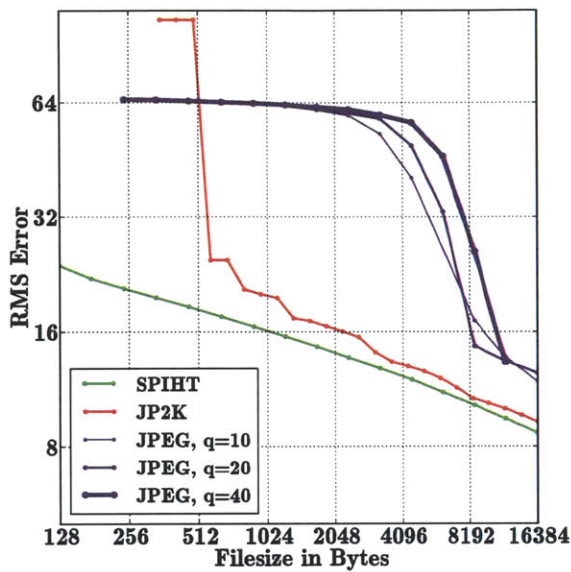
(a) Pillow Lava
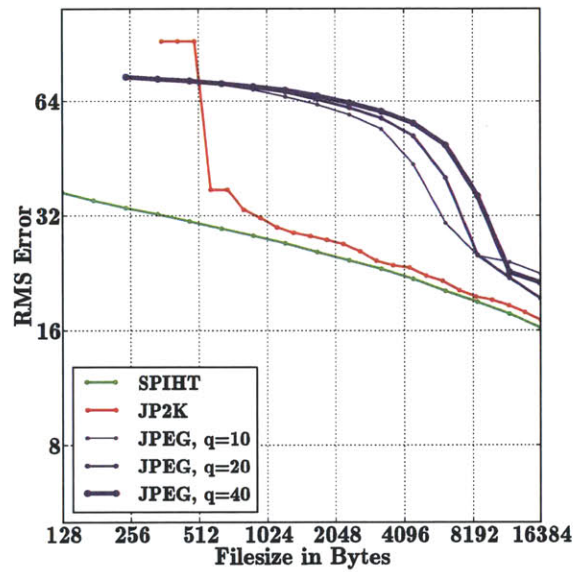
(b) Coral Reef
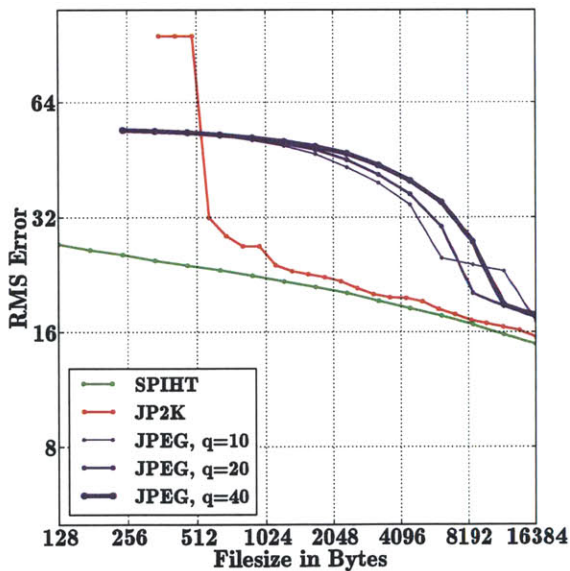
(c) Airplane
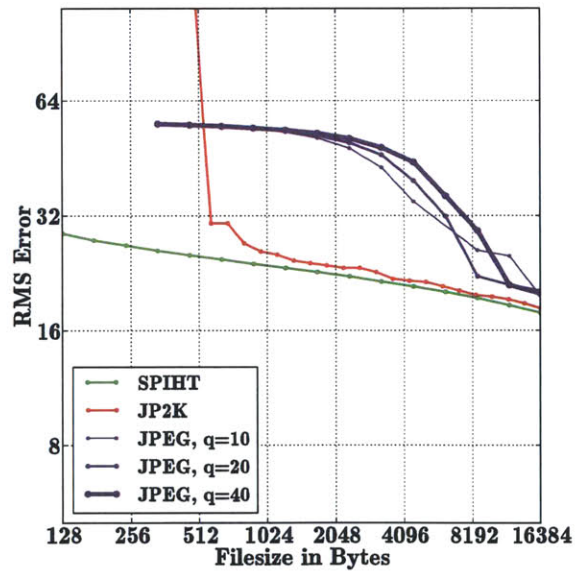
(d) Fish and Sand

Figure 3-11: Reconstruction error versus number of bytes for the four representative images shown in Fig. 3-10, encoded from 1024×1024 pixel source images. Comparing the discrete quality 'checkpoints' visible as 'bumps' in the progressively coded JPEG 2000 (JP2K) data, with the smooth progression of the SPIHT coding, we learn that JPEG is largely incapable of encoding large images at these low sizes.
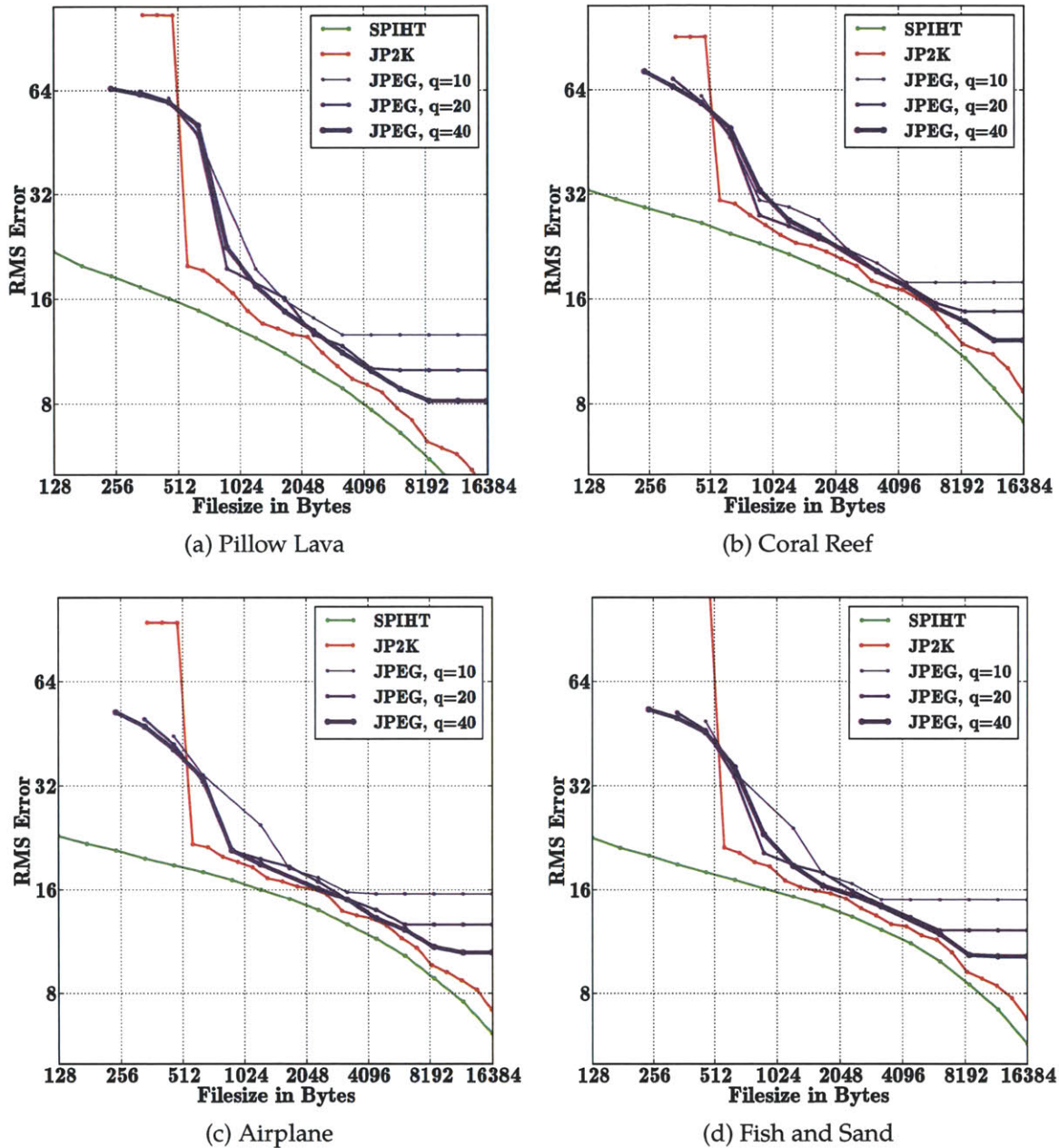
Figure 3-12: Reconstruction error versus number of bytes for the four images shown in Fig. 3-10, encoded from small, 256×256, pixel source images. Discrete quality 'checkpoints' are very evident as 'bumps' in the progressively coded JPEG 2000 (JP2K) data, while SPIHT coding provides a smooth compression curve. JPEG performs adequately at larger filesizes, until reaching the target encoding quality. Counter-intuitively, the higher-quality (larger q) JPEG compression typically has better image quality even at lower filesizes.
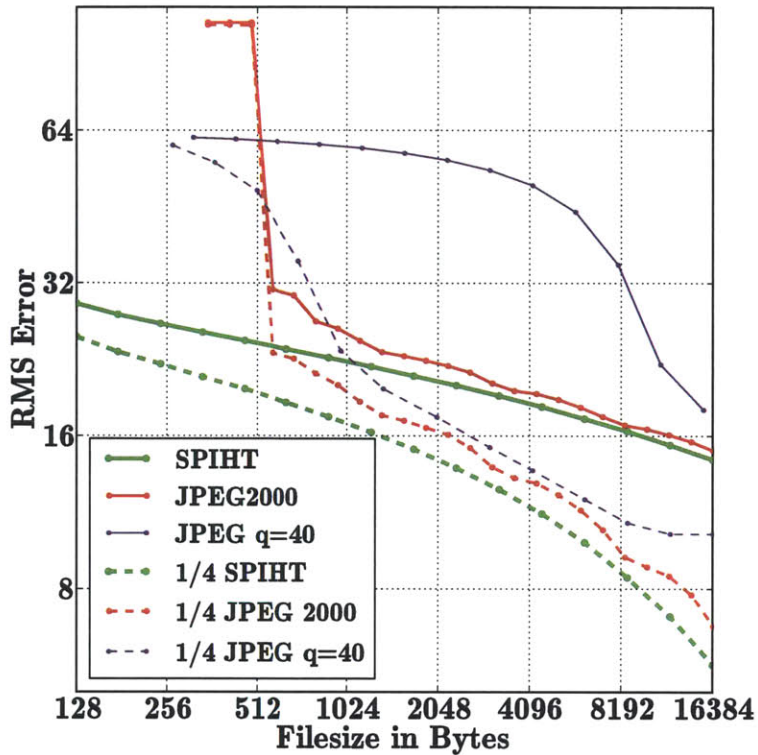
Figure 3-13: The mean reconstruction error across all four images from Fig. 3-10. The solid lines indicate versions of the images that were 1024×1024 pixels in size, the dashed lines represent images that are 256×256 pixels. Beyond the benefits of the fully embedded coding, SPIHT clearly provides the highest compression performance for both small and large imagery.

Fig. 3-11, while the dashed lines represent the 256×256 pixel source images of Fig. 3-12. This summary chart shows that beyond the aforementioned benefits of fully embedded coding, SPIHT clearly provides higher compression performance than JPEG or JPEG 2000 for both small and large imagery target sizes.

3.3 Image Synthesis
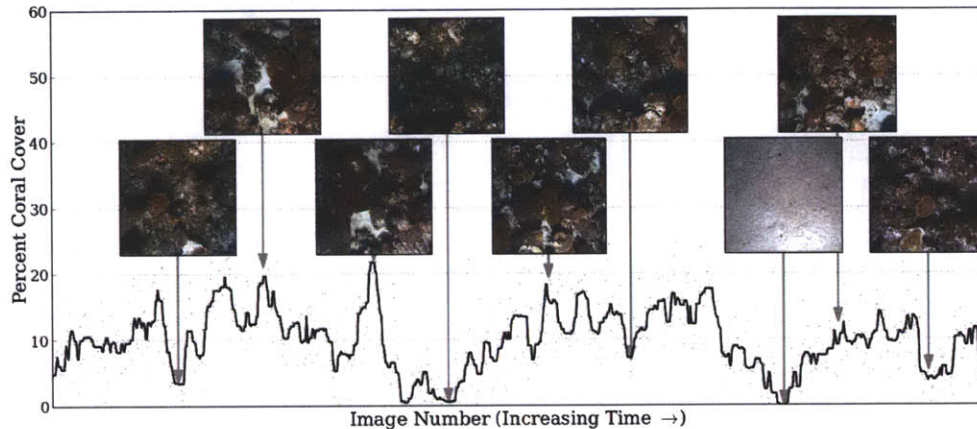
Figure 3-14: Percentage of coral cover for each image. Black line is the 11-point median-filtered percentage, computed from the individual samples. Note that areas dominated by sand and rubble both correspond to areas of low coral cover.

Extensive anthropogenic damage to shallow-water coral has been well documented [38][29], yet it is much more challenging to study the health of coral reefs below diver depths. AUVs allow scientists to not only reach deeper than a single diver, but cover more area as well [4]. Unfortunately, transmitting imagery at high enough quality to distinguish healthy coral from rubble requires significant subsampling of the captured imagery, as fine-scale texture information suffers from significant blurring in wavelet compressed imagery at high compression.

Wavelet compressors are highly efficient at encoding intra-image redundancy, having amongst the highest known compression ratios on single images. Given the limited bandwidth available, and the significant quantity of imagery that AUVs capture, it would be nice to also take advantage of inter-frame redundancy. One way to do that is to compute scalar metrics (e.g. percent coral cover) and transmit those instead of the individual images. Such a technique allows the content of imagery to be communicated much faster than a compressed image, describing entire datasets in a few hundreds of bytes. When such a metric is the goal of data acquisition (e.g. measuring the percentage of coral cover), this scalar metric ap-

proach is an extremely efficient way of communicating information about imagery acquired subsea. Fig. 3-14 plots the percentage of each image covered in coral for a dataset acquired by the author as part of a study on the long-term health of deep-sea coral reefs near Puerto Rico. Distinctive changes in imagery correspond well with changes in the percentage of coral cover, as illustrated.

Still, this representation sacrifices information about the distribution of coral within these specific images, and about the character of each image more generally. I now present a middle ground between these two extremes of imagery and scalar image statistics nicknamed Image Synthesis.

3.3.1 Image Synthesis

Image Synthesis employs inter-image repetition in texture space to summarize information about the contents of an image, allowing an extremely compact representation to be communicated while still allowing estimations of scalar texture-based metrics such as the percentage of coral cover based on the received image. The resulting imagery can be transmitted in much less space than the compression techniques described previously in this chapter, consuming only tens of bytes per image on average. To achieve high compression ratios I employ both inter-image and intra-image redundancy in texture space by describing each new image as a set of previously seen image texture patches. This allows the surface reproduction of imagery with rich textures, at the cost of a decreased ability to communicate previously unseen imagery. A single image compressed using each of these techniques is shown in Fig. 3.1, along with a time-series representation of the percentage coral cover. Parts (a), (b), and (c) of this figure each represent embedded-wavelet compressed versions of the image at different sizes. Part (d) represents the image compressed using the image synthesis method. The largest SPIHT image (a) clearly provides a better representation of the true image than the synthesized image (d). Comparing the synthesized image (d) to a SPIHT image of comparable file size (c), the image synthesis technique is more appealing. Additionally, the percentage of coral cover could be trivially computed from the synthesized image, whereas it is

nearly impossible to determine from the comparably sized SPIHT image. Part (e) shows that computing the percentage of coral cover from synthesized images (blue line) provide a reasonable estimate of the true coral cover (black line). Note that, alternatively, the percentage of coral cover for every image could be transmitted as a scalar time-series (green line).

The procedure for compressing a sequence of images using Image Synthesis consists of four steps, depicted in Fig. 3.2. First, source imagery is segmented and classified into areas of similar texture. Second, this segmented image is drastically subsampled, resulting in a low-resolution map of texture blocks within the image. Third, each of these masks are encoded in a non-sequential but deterministic order, using an arithmetic coder. Finally, when each mask is received on the surface, a texture synthesis procedure is used to synthesize an image similar to the one compressed. I proceed with further description of each of these steps in turn.

3.3.2 Image Segmentation

Segmentation and classification of seafloor imagery remains an active research topic in underwater robotics. Pizarro, Rigby et al.[82][83] have presented results obtained with a 'bag-of-features' approach, using SIFT descriptors as their feature space. Loomis[62] obtained high classification ratios using only 5x5 patches as textons, and by developing a classifier that relied on boosting. The approach pursued here for this dataset is inspired by, though different from, the efforts of Kaeli et al[56] using morphological image processing on a similar data set. I document below the segmentation and classification used on this dataset in the interest of completeness, though the recent work described above would provide a more appropriate starting point for future implementers seeking a flexible implementation. The segmentation and classification procedure described here results in a classification mask containing regions in one of four classes: sand, rubble, M. Annularis coral, and unclassifiable (such as shadows). For all examples in this section, the dimensions of the source imagery are 504×504 pixels.

Each image is processed independently. Initially it is converted to an eight-bit
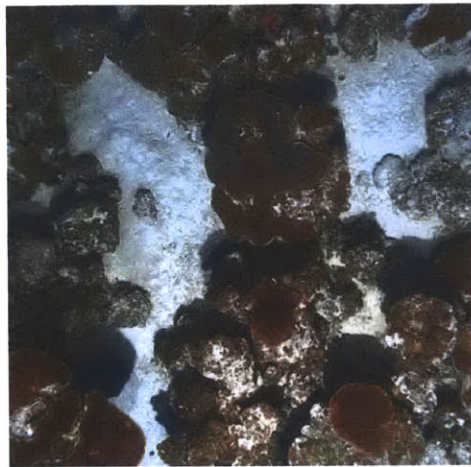
(a) SPIHT (400x400 @ 0.04bpp) (20,000 Bytes)
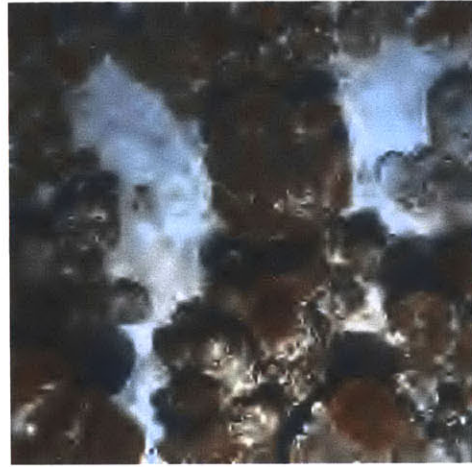
(b) SPIHT (400x400 @ 0.004bpp) (2,000 Bytes)
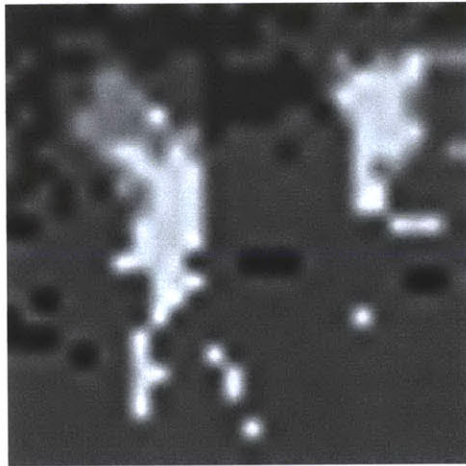
(c) SPIHT (400x400 @ 0.0004bpp) (200 Bytes)
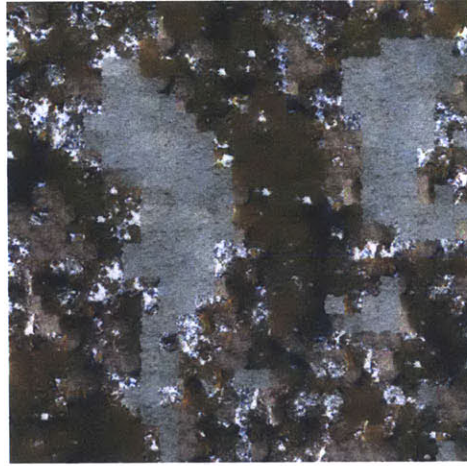
(d) Synthesized (116 Bytes)
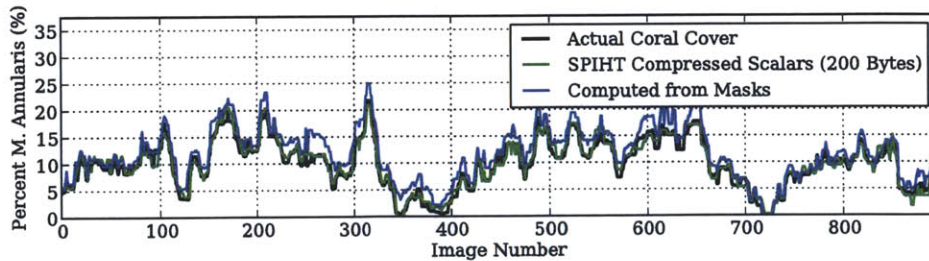
(e) Percent Coral Cover (200 Bytes)

Table 3.1: Coral reef imagery, encoded using the full spectrum of techniques described by this thesis. Note that calculating the percentage of coral cover from the 200 Byte SPIHT-encoded image would be impossible, yet is trivial for the synthesized image. Communicating the same statistic as scalar data allows the entire dataset to be represented in the same number of bytes. Each of the time series' has been filtered with an 11-point median filter for clarity.
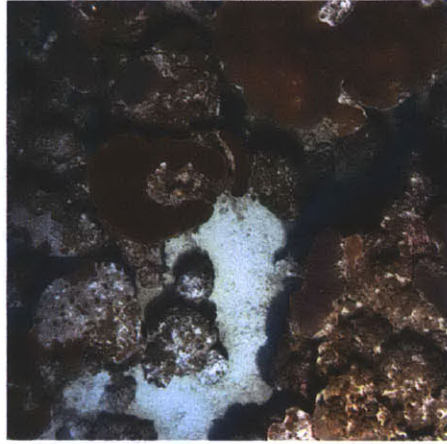
(a) Source Image
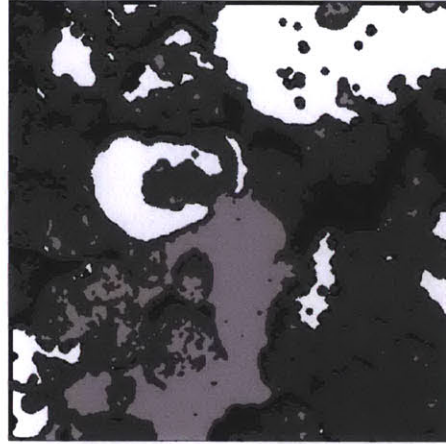
(b) Segmented + Classified Mask

(c) Subsampled Classification Mask

(d) Entropy Coding
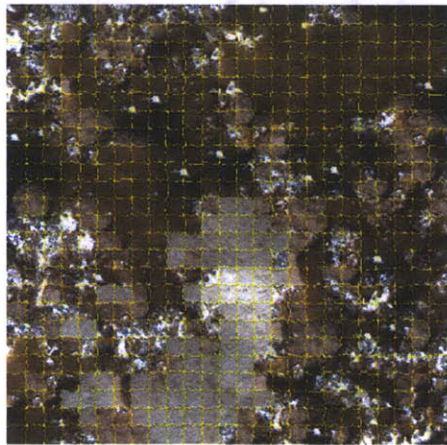
(e) Synthesized Image, with Cuts
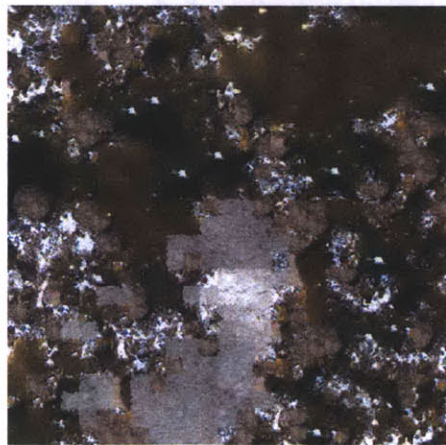
(f) Final Synthesized Image

Table 3.2: The original image is shown in (a). The image following segmentation and classification is shown in (b). Part (c) next displays the subsampled image, (d) represents the relative sizes of the data as encoded naively (8 bits per pixel), using quantization alone (2 bits per pixel), and using arithmetic coding (0.07 bits per pixel). The reconstructed image is shown in (e) and (f), with and without inter-tile boundaries.
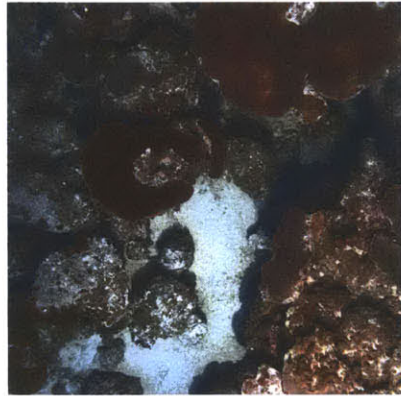
(a) Source Image
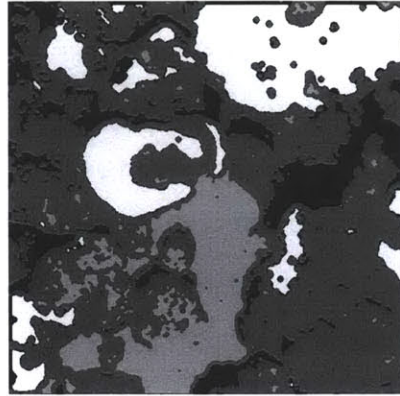
(b) Classification Results
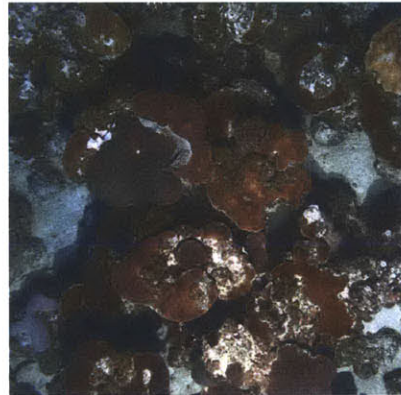
(c) Source Image
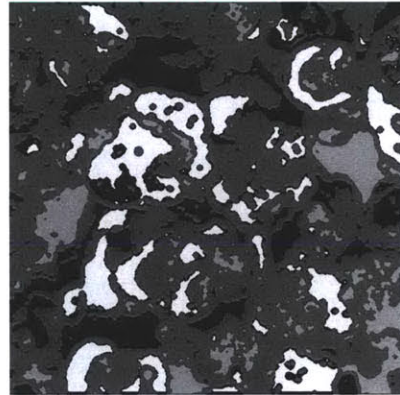
(d) Classification Results
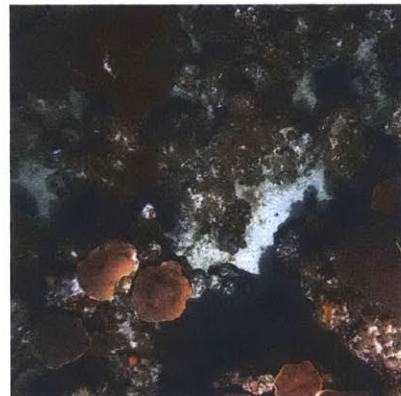
(e) Source Image
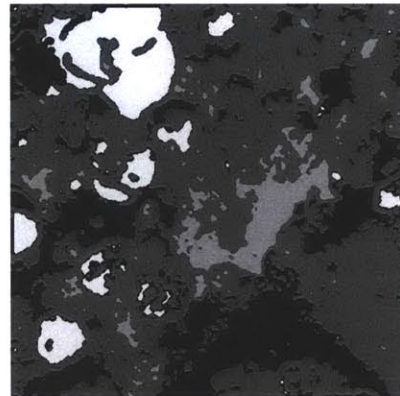
(f) Classification Results

Table 3.3: Segmentation and classification results, computed using hand-tuned local image statistics.

$Y'C_BC_R$ colorspace, separating luminance and chrominance data. For each resulting channel, the local mean and variance are computed for a 5×5 neighborhood at each pixel. A sequence of binary thresholding operations over these statistics, and comparisons, form the basis for the classifications. After each class is identified, it is smoothed with a morphological Alternating Sequence Filter (ASF)[25] to smooth the classification. This smoothing is done with a circular structural element as described in Kaeli et al[56]. Classification was performed sequentially on a per-class basis, using manually tuned thresholds. While the thresholds were manually selected, the final classification procedure performed well on each of the 919 images in the dataset.

The resulting image masks for three distinct images are shown in Fig. 3.3. After a classification mask has been produced for each image, the mask is subsampled. This subsampling is not performed through simple decimation or interpolation, but by computing the dominant class within a grid of fixed-size windows. Grid cells overlap adjacent grid cells by a constant amount, as seen in Fig. 3-15.

Figure 3-15: A schematic of the grid used for subsampling the classification mask, illustrating the overlap between adjacent cells. Note that the lack of overlap at image edges means that a simple interpolation would bias the resulting mask.

This procedure accounts for overlap between adjacent texture patches that will occur during the final phase of image synthesis. As a result of this overlap, subsampling a 504×504 pixel image with a 24×24 pixel window that overlaps by four pixels yields a square mask 25 pixels per side, rather than $\frac{504}{24} = 21$ pixels on a

side as might be expected. Within each window, the most common identified class (statistical mode) is taken as the subsampled tile class. For the imagery presented here, 24×24 pixel grid cells were used, overlapping by 4 pixels.

3.3.3 Arithmetic Coding of Texture Masks

At this point, a low-resolution mask of texture classes has been generated. The simplest approach would be to represent each texture class with an integer, and sequentially encode the mask pixels. For the four-class images shown here, each mask pixel could be represented in only two bits, resulting in a size of $\lceil \frac{25 \times 25 \times 2}{8} \rceil = 157$ bytes per mask for the parameters used here. Compressing an image with dimensions (h_i, w_i), using patches of dimension (h_p, w_p) that overlap by o pixels in each dimension, each containing one of c texture classes, would require $\lceil \frac{h_i - o}{h_p - o} \rceil \lceil \frac{w_i - o}{w_p - o} \rceil \log_2 c$ bits. Each texture class is not uniformly probable, however, and the pixels are not statistically independent of each other. Given these facts, the use of entropy coding offers the ability to shrink this image representation even further.

Arithmetic coding allows for the optimal encoding of a sequence of symbols when the probability distribution of those symbols is known to both the encoder and decoder. Adaptive arithmetic coding[129] provides for near-optimal encoding by learning the probability distribution as the sequence is encoded or decoded. The most simple adaptive coders simply generate a frequency table of coded symbols, and use that as the probability distribution for the next symbol. More elaborate probabilistic models can be built by tracking the conditional frequencies of symbols based on, for instance, the previously encoded symbol. In the sequence *AAAABBBBAAAAAAAA*, for instance, *B* is not a particularly likely symbol. If we track conditional probabilities, we see that *B* has a much higher likelihood of appear immediately after another *B*, and can build a better probabilistic model of our sequences.

The choice of an appropriate adaptive model can have significant impact on the efficiency of the arithmetic coding. Table 3.4 shows the results of encoding a sequence of 919 texture masks using an adaptive arithmetic coder, with each of

81

Context	Mean (Bytes)	Median (Bytes)	Std. Dev. (Bytes)
m_{i-1}	49.60	51.0	14.65
m_{i-1}, m_{i-2}	49.44	51.0	14.62
m_{i-1}, m_{i-w}	40.98	42.0	11.89

Table 3.4: Comparison of Arithmetic Coding Models. Table shows average size in bytes for transmitting a single image using the specified adaptive model.

three different adaptive models. The first model conditions probabilities upon the texture class of the previous pixel. The second model conditions the probabilities upon both of the previous two pixels. The third, and most effective model, conditions the probabilities on the previous pixel and the pixel directly above the current pixel. Note that it is not possible to condition on future pixels, as the decoder will not have access to those pixels until after decoding the current one. As seen in Table 3.4, conditioning probabilities upon the pixel directly above the current one in addition to the previous pixel results in a significant improvement in encoding efficiency. The encoded size of each mask is plotted in Fig. 3-16 for the first and third adaptive models.

Source images with large regions of constant textures will have less entropy in the texture mask. This can significantly increase the efficiency of the arithmetic coder, resulting in a smaller compressed representation of the image. As one example, Fig. 3.5 consists of almost entirely two texture classes, rubble and sand, arranged as two large regions. The texture mask, before and after downsampling to 25x25, are also shown in Fig. 3.5. After arithmetic coding, the texture mask consumes only 27 bytes. The average size for this dataset, in contrast, was 41 bytes. The amount of entropy of the textures in the compressed imagery is easily computed from the transmitted imagery, and could be used subsea for image selection, computed on the surface for received images, or transmitted as a separate scalar statistic.

Rather than encode these texture masks in the order that the source images were captured, I encode the masks in a deterministic manner that provides samples from across the entire sequence, gradually filling in the entire time-series of images
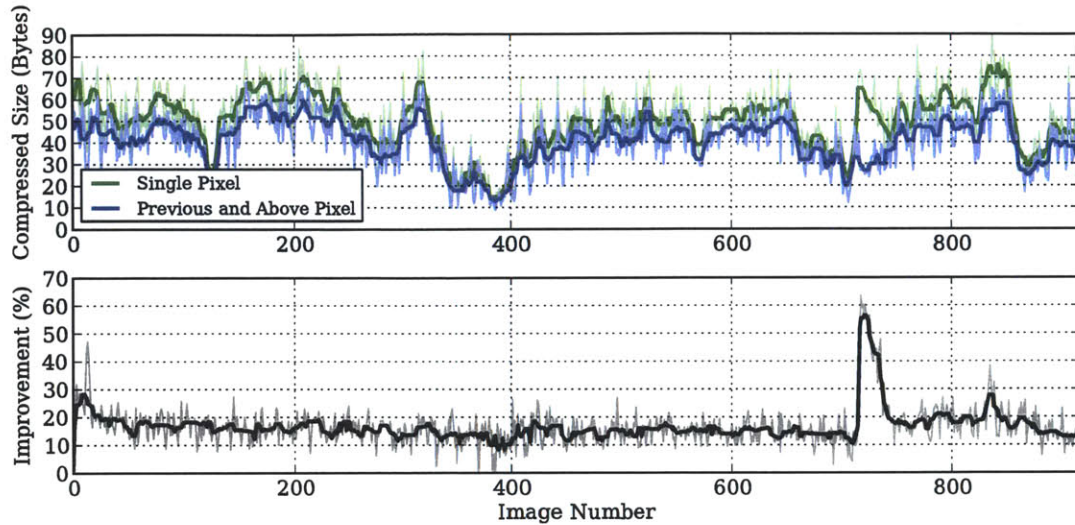
Figure 3-16: The number of bytes required to transmit each image using arithmetic coding is shown for two different adaptive models. By utilizing both the previous neighbor and the neighbor above each new pixel, the data may be compressed further (blue) than when only utilizing the previous neighbor (green). Note the much higher performance of the model incorporating the vertical neighbor when coding a sequence of images near the end, each of which is entirely sand. Both signals are filtered with an 11-point median filter for clarity.

rather than encoding each image in turn. This allows surface operators to quickly get a rough idea of an entire dataset, rather than a clear view of the beginning of a dataset. Specifically, for a sequence of n images, I iterate through the sequence with a step size of $\frac{n-1}{2}$, beginning with the first image. When I reach the end of the sequence, I reduce the step size by half, and continue from the beginning of the sequence, skipping images which have already been encoded.

3.3.4   Image Synthesis

After receipt of a texture mask, a new image is synthesized by the recipient to match the form of the mask. In Efros and Freeman[30] a method for synthesizing textures, nicknamed "Image Quilting", is presented. The synthesized textures are generated from patches of a source texture, selected to meet some minimum error criterion. Much of the seafloor consists of large swaths of repetitive textures - sand and rubble on a macro scale, and the fine-grained textures of coral and on a microscale.

83

(a) Source image exhibiting large constant-texture regions

(b) Texture mask for source image

(c) Downsampled texture mask
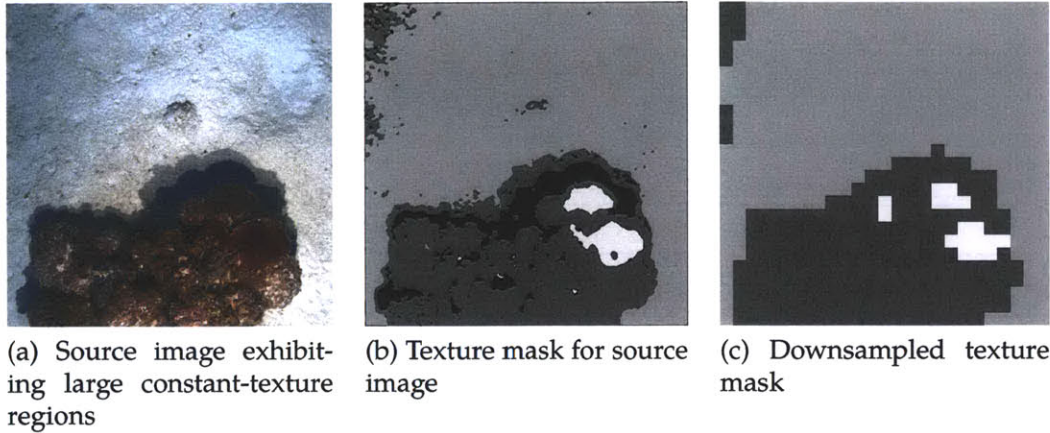
Table 3.5: Constant-texture images, like this one, have less entropy in the texture mask. This results in greater coding efficiency during the arithmetic coding stage.

This texture synthesis approach can be used effectively on individual textures to generate high resolution samples from low-resolution samples, as in Fig. 3.6, 3.7, and 3.8.

To synthesize an image which is compatible with the received texture mask, then, each patch must be drawn from a source texture corresponding to the class of that pixel in the mask. As each texture mask is decoded by the recipient, it is used to synthesize an image with high-resolution textures, closely approximating the original image. The quilting process, adapted from Efros and Freeman[30], is completed as follows:

1. For a received texture mask with dimensions $h \times w$, a patch size of t, and overlap o, initialize an empty destination image with dimensions $(h \cdot (t - o) + o, w \cdot (t - o) + o)$.

2. Go through the received texture mask in raster scan order. For each pixel, search the source texture corresponding to that pixel's class for a set of patches that match the patches it overlaps within some error tolerance. Randomly pick one of these patches.

3. Compute the error surface between the newly chosen patch, and the patches it overlaps. Find the minimum cost path along this surface and make that the

(a) Montastraea Annularis (192 × 192)

(b) Synthesized Montastraea Annularis texture

(c) Synthesized Montastraea Annularis texture with Minimum-Error Cut boundaries

(d) Montastraea Annularis (256 × 256)

(e) Synthesized Montastraea Annularis texture

(f) Synthesized Montastraea Annularis texture with Minimum-Error Cut boundaries

(g) Montastraea Annularis (256 × 256)

(h) Synthesized Montastraea Annularis texture

(i) Synthesized Montastraea Annularis texture with Minimum-Error Cut boundaries

Table 3.6: Synthesis Results for Montastraea Annularis Textures. Synthesized texture is (504 × 504) pixels. Textures are synthesized from 24 × 24 pixel patches.

(a) Rubble (192×192)

(b) Synthesized Rubble texture

(c) Synthesized Rubble texture with Minimum-Error Cut boundaries

(d) Rubble (256×256)

(e) Synthesized Rubble texture

(f) Synthesized Rubble texture with Minimum-Error Cut boundaries

(g) Synthesized Combined Rubble texture, (24×24) patches, (504×504)

(h) Synthesized Combined Rubble texture with Minimum-Error Cut boundaries
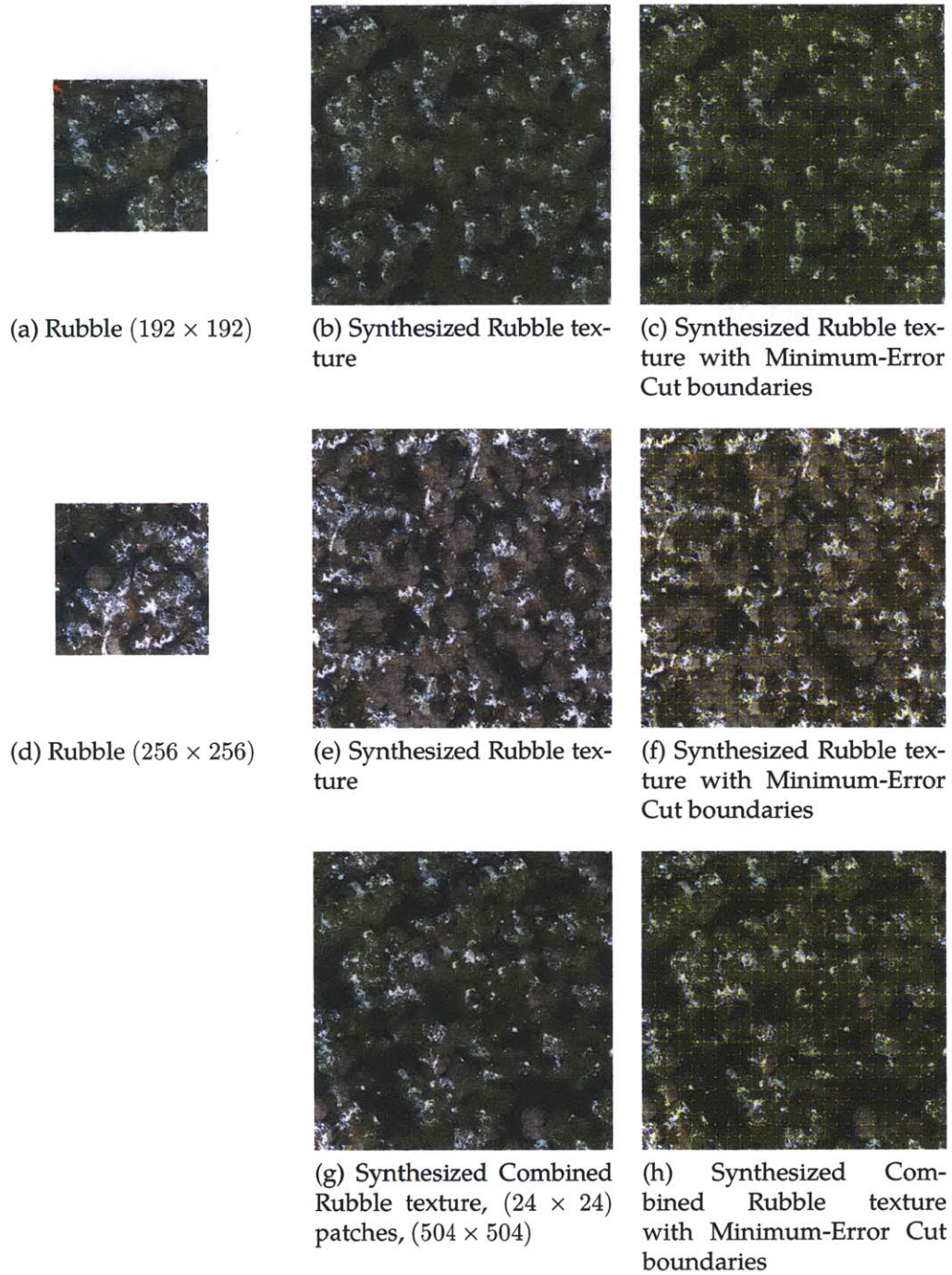
Table 3.7: Synthesis Results for Rubble Textures. Synthesized texture is (504×504) pixels. Textures are synthesized from 24×24 pixel patches. The third row represents synthesis from a combination of both rubble textures (a) and (d).

(a) Gorgonian (192×192)

(b) Synthesized Gorgonian texture

(c) Synthesized Gorgonian texture with Minimum-Error Cut boundaries

(d) Brain Coral (256×256)

(e) Synthesized Brain Coral texture

(f) Synthesized Brain Coral texture with Minimum-Error Cut boundaries

(g) Sand (256×256)

(h) Synthesized Sand texture

(i) Synthesized Sand texture with Minimum-Error Cut boundaries
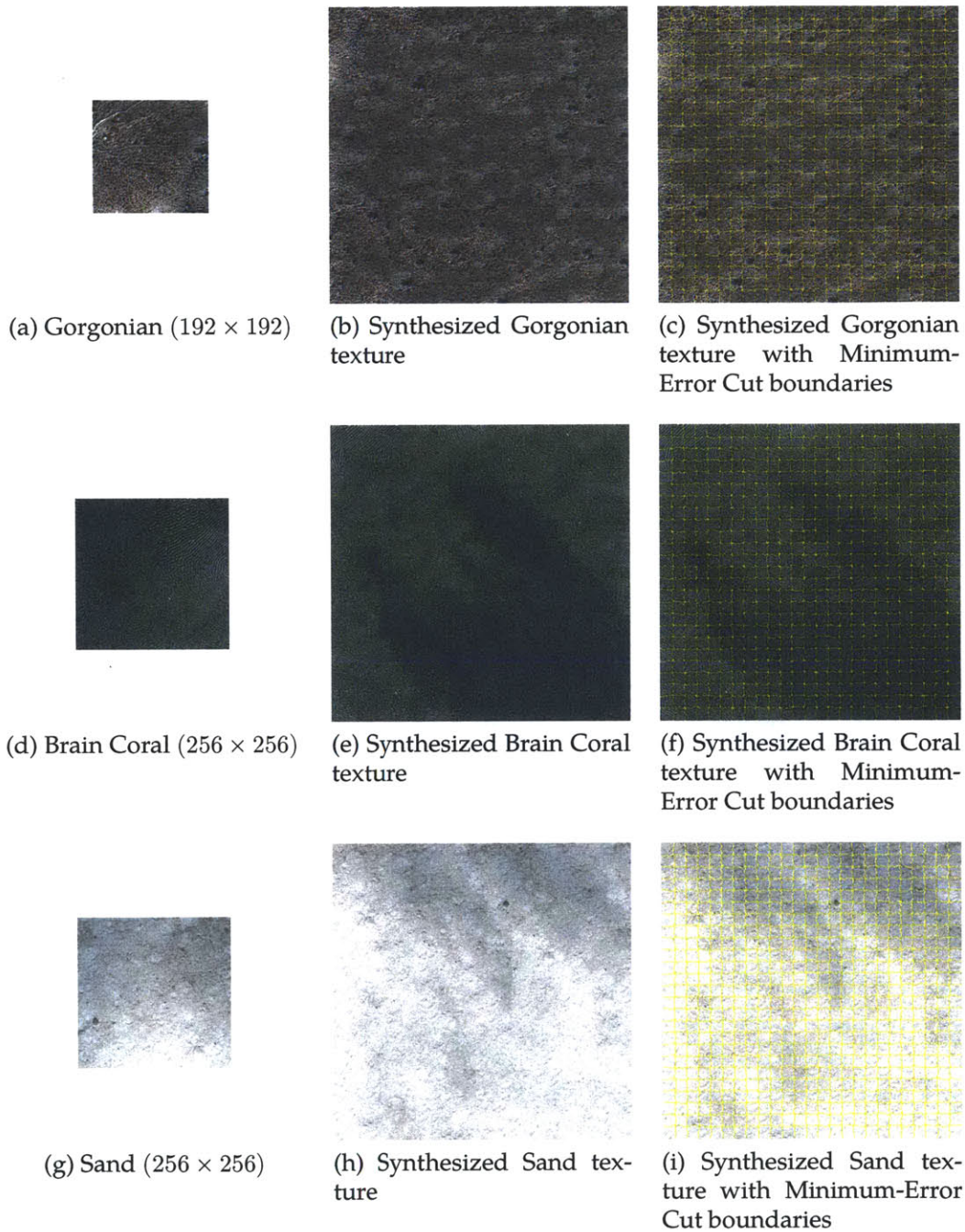
Table 3.8: Synthesis Results for Miscellaneous Textures. Synthesized texture is (504×504) pixels. Textures are synthesized from 24×24 pixel patches, with four pixels of overlap.

boundary of the new patch. Paste the patch onto the destination image.

For both the second and third stages of this algorithm, the error between patches was computed as the L^2 in a scaled $Y'C_BC_R$ space, where the Y channel error is scaled by 1.3, and each chrominance channel error is scaled by 0.85, more heavily weighting luminance errors over chrominance errors. Three images generated using this full technique, including the minimum cost paths between adjacent patches, are shown in Fig. 3.9. Note that s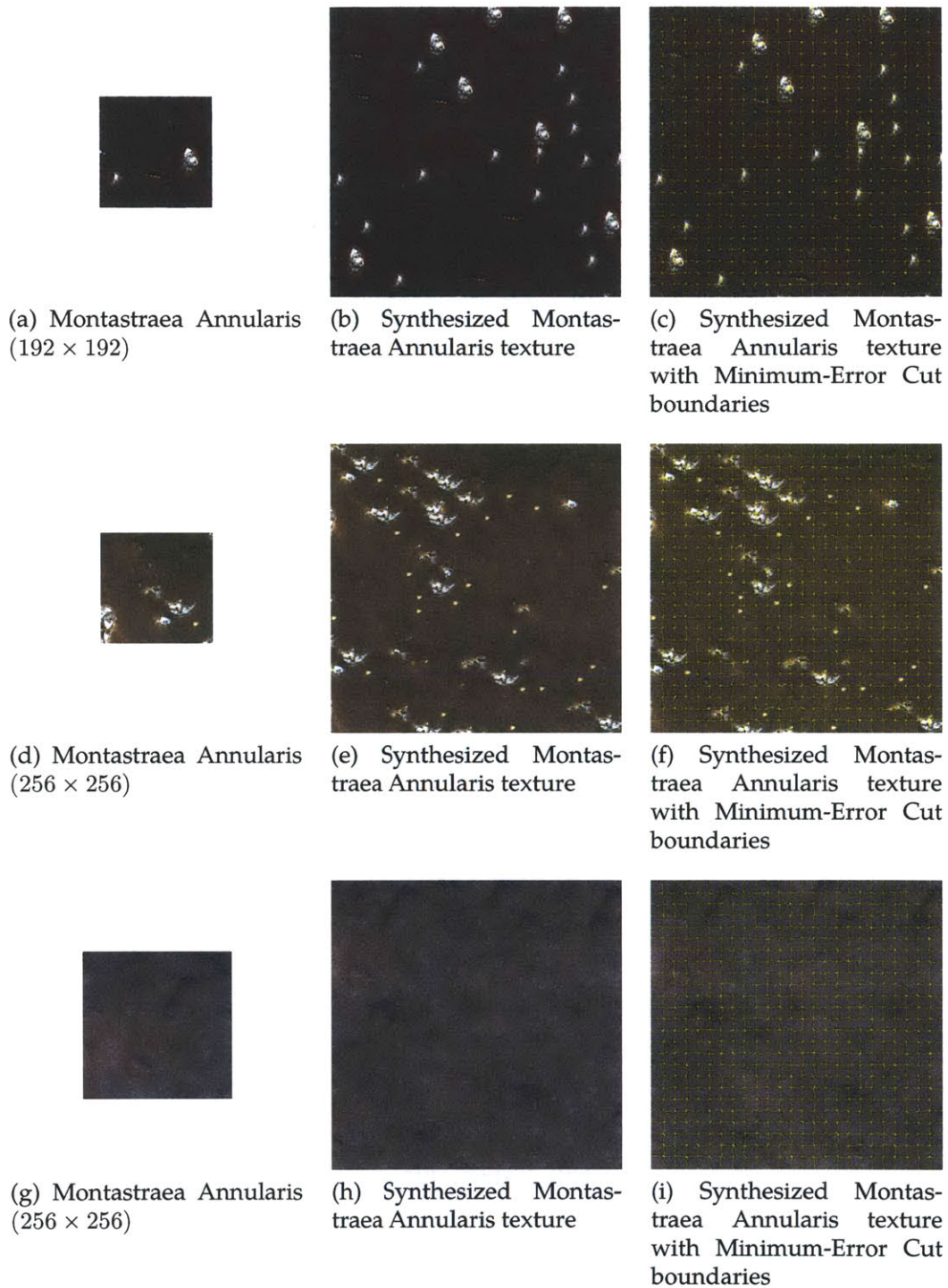ince the current encoding does not transmit any color information explicitly, there may be a mismatch in the color or appearance of the synthesized image relative to the first image. The nearness of the match between the synthesized image and the original image depends upon how broad the set of textures are within one texture class. Not transmitting color information keeps the compressed size down as far as possible (.0004 bits per pixel per color channel for this dataset, on average). Were color information sent up as well, it could be used as a constraint during the texture synthesis phase as described in Efros and Freeman's original paper[30]. In this way, the color information of a low resolution SPIHT image could be combined with the texture information of a synthesized image, for example.

3.4 Discussion

In this chapter I have laid out several options for compressing telemetry from AUVs, including methods applicable to both environmental data and imagery. For imagery, I have illustrated a range of encoding options: treating summary statistics as time-series, communicating texture information via Image Synthesis, or transmitting full images with embedded wavelet compression. While selection of the appropriate technique from these options must be done with an understanding of the problem domain, the techniques are complimentary. For typical AUV missions, a combination of all the approaches described in this thesis may be appropriate.
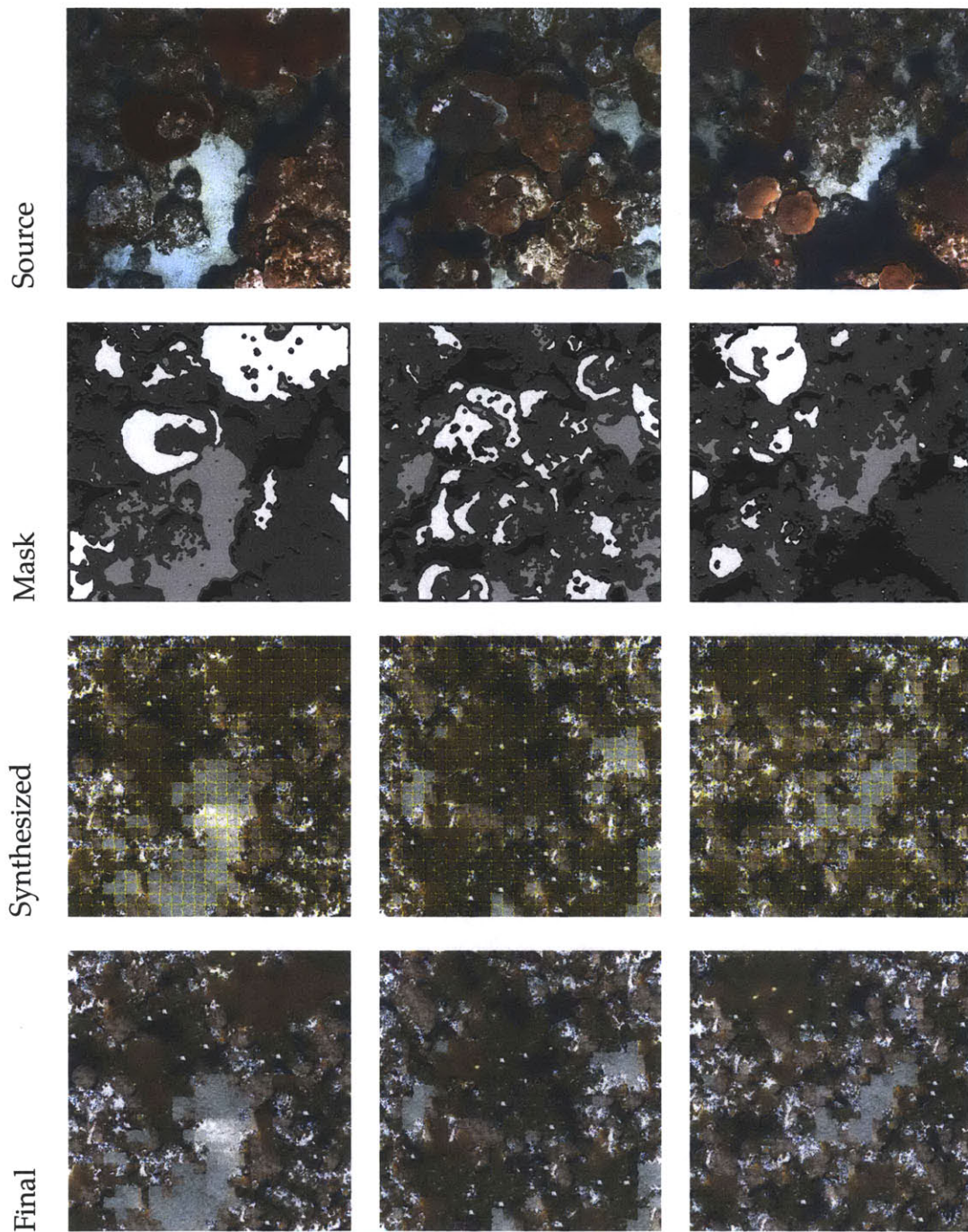
Table 3.9: Three full 504×504 pixel images synthesized using this approach from 24×24 pixel patches, with four pixels of overlap between adjacent patches.

CAPTURE Architecture

Having outlined the approach to compression and relay communication underlying this work, I describe here how to integrate those two components into a full end-to-end communication architecture for relaying acquired data from an AUV across an acoustic network of marine vehicles. This architecture is nicknamed CAPTURE—a Communications Architecture using Progressive Transmission via Underwater Relays and Eavesdroppers. CAPTURE relies on progressive transmission to communicate data as a sequence of gradually improving "previews". High-quality versions of these previews, up to an error-free reconstruction, can be requested by operators immediately, or at any later time over the course of a mission. CAPTURE has been designed to facilitate efficient multi-hop relay communication across a small group of vehicles, where the vehicles involved may be from different manufacturers, or have different software architectures.

4.1 Overview

CAPTURE consists of four distinct components, shown in Fig. 4-1. First, a set of data is acquired by the AUV and registered as a transmittable resource with the telemetry system, via a platform-specific driver. Examples of possible resources include a single image, or a time-series of measurements from a single sensor. The platform-specific drivers isolate the telemetry system from the specific capabilities or limitations of each host vehicle. Second, new resources are automatically selected for compression and transmission to the surface, or existing resources are selected for further transmission based on requests from the surface. Automatic
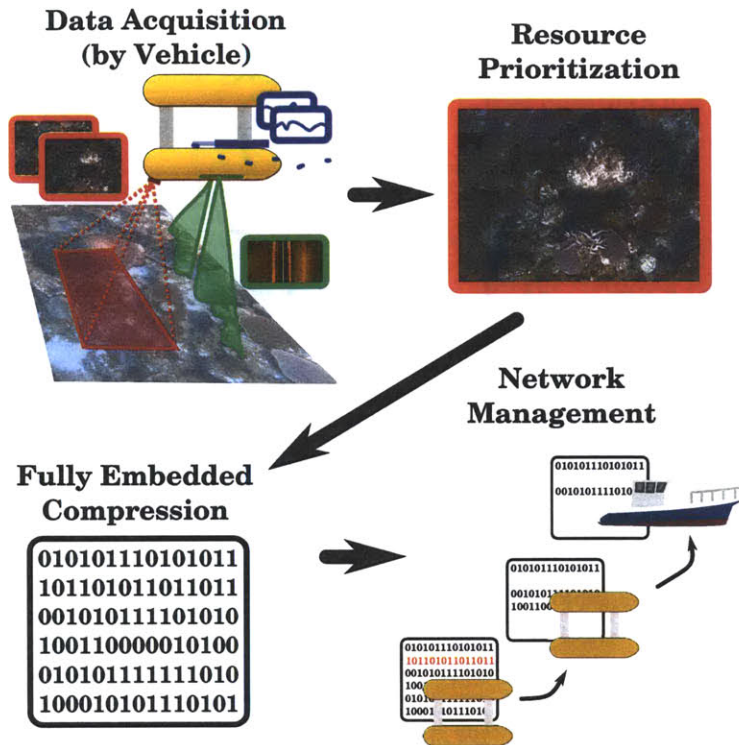
Figure 4-1: High-level overview of data flow through the four main components of CAPTURE. Platform drivers connect acquired data into the CAPTURE system, which is then winnowed down, compressed, and eventually transmitted to the surface.

selection provides an avenue for high-level algorithms, such as mine identification or other interest operators, to guide the selection of interesting telemetry. Third, selected resources are compressed using progressive coding methods. Progressive coding methods, specifically those that are fully embedded, ensure that an approximation to the data can be reconstructed with each newly received bit of data. Finally, the transmission of the resource to the surface is managed to ensure end-to-end delivery. When multiple underwater vehicles are available, intermediate vehicles can relay data to the surface as hops in the route, or help through 'eavesdropping'. The flow of data between the four subsystems is shown in detail in Fig. 4-2, and each subsystem is described in detail in the following subsections.
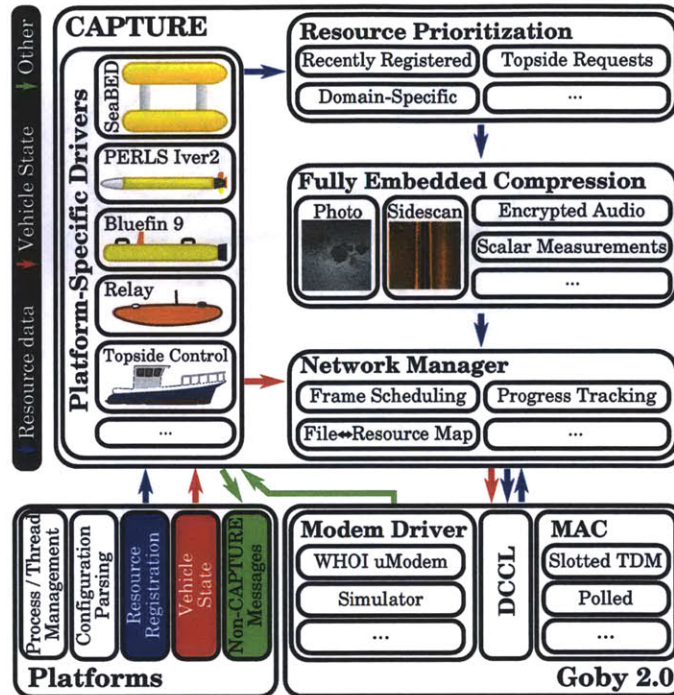
Figure 4-2: A detailed view of the interactions between CAPTURE components, particularly the role of the platform drivers. Blue indicates the path of a resource (such as an image) through CAPTURE; red and green indicate vehicle state and control messages bypassing the majority of CAPTURE components.

4.2 Platform Drivers

The platform drivers provide an interface to the existing software on each different vehicle platform. Software architectures vary significantly from vehicle to vehicle, as do sensing and computation capabilities. Platform drivers smooth over these architectural differences by providing:

- An interface for data transmission and reception via the modem,

- Configuration of resource registration and prioritization,

- Handling of non-CAPTURE acoustic traffic, such as command and control messages, and

- Logging support via LCM[50].

Physical connections to the vehicle's acoustic modem vary, but most are connected to an RS-232 serial port. Each modem manufacturer provides a different

software interface to the end user—there does not yet exist a common API shared by multiple manufacturers such as the Hayes / AT Command Set that dictated the course of terrestrial telephone modems. Webster et al. previously developed a modem abstraction layer for the WHOI MicroModem[127]. More recently, the Goby Autonomy Project[100] has made advances in developing a generic abstraction for acoustic modems and implementing drivers for physical modem hardware. These drivers allow software to operate independent of the modems' underlying proprietary languages. Goby was used in the field experiments to provide a low-level vendor-neutral interface to the acoustic modem.

AUVs will require some configuration, such as information about any acoustic range-based navigation systems that are in use, or the specification of a fixed MAC communication cycle. Each modem requires a unique integer identifier, typically specified as part of the configuration. That configuration is performed through the platform driver. The platform driver is also responsible for registering existing sensors, such as cameras, sonars and CTDs, as resource generators. The importance of different resources will vary by mission and vehicle, so their prioritization may require pre-mission configuration by users. Some vehicles may only register a single camera and transmit imagery. Other vehicles may switch between multiple sensors, such as a camera and a CTD, selecting between the resources during the prioritization phase. Command and control messages, such as vehicle aborts or mission changes, are also delivered by the driver to appropriate handlers.

4.3 Resource Prioritization

Modern AUV platforms generate orders of magnitude more data than could possibly be transmitted to the surface—the first task facing any telemetry system is to prioritize which data should be transmitted. At any given time, surface operators can choose whether to request refinement of a specific resource or whether to allow the vehicle to automatically select new resources for transmission. For vehicles with multiple sensors of interest, it is also necessary to multiplex the transmissions

between those sensors. These steps can be quite simple, such as always sending the most recent resource registered by a single sensor. More complex missions may involve significant computation in this step, such as identifying seafloor mines through image analysis. Yogesh Girdhar has published a sequence of papers[39, 40, 41, 42] on identifying a subset of images that appropriately describe an entire collection - similar to the problem of selecting an appropriate subset for a slideshow. These papers include both offline methods, useful for identifying a summary subset after a dive, and online methods. The online identification method (nicknamed ONSUM) builds the summary set as new images are being acquired, making it ideally suited to telemetry prioritization. These methods have been tested with several datasets, including one from a small shallow-water AUV. Thompson et al[118] also have earlier work on optimal prioritization of telemetry for the Zoë autonomous rover, based on Hidden Markov Models.

Multiplexing of multiple sensors on a single vehicle could be done with a round robin scheduling-based approach, priority queues, or computed metrics. While a single image is easy to consider as a distinct 'resource', transmitting environmental sensor data currently requires breaking data into fixed-length segments for transmission. This is best done with long sequences of data at time to maximize compression, as discussed in Sec. 3.2.2.

4.4 Progressively Encoded Compression

After identifying a resource for transmission to the surface, that resource must be compressed to maximize the throughput of the channel. CAPTURE relies on progressively coded compression methods—preferably fully embedded ones. CAPTURE transmits enough data to the surface to reconstruct a low-quality "preview" of each automatically selected resource before moving onto a new resource. Due to the progressive nature of the encoding, each new piece of data received on the surface will allow an increasingly higher-quality representation of the resource to be reconstructed. This serves two equally important purposes. If the "preview"

piques the operator's interest, the operator can request more encoded data from that resource to refine the already-transmitted data with no wasted transmissions. Every byte sent up for the preview will be used as the basis for the higher-quality version. If, on the other hand, the resource is uninteresting, the operator may be able to determine that after only a few transmissions and avoid wasting further bandwidth to deliver a full preview.

4.5 Multi-Hop Networking
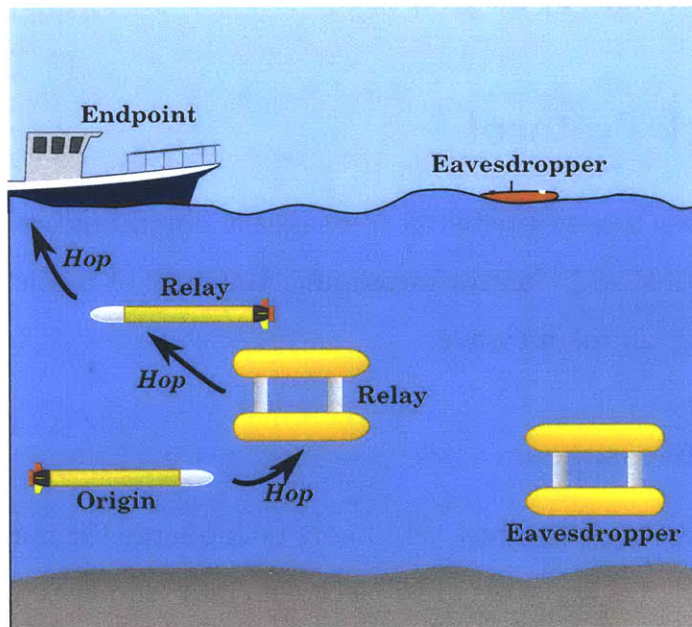
Figure 4-3: A large CAPTURE network, including multiple hops and eavesdroppers. The vehicle selecting resources for transmission is known as the 'origin', and the surface ship is the 'endpoint'.

A CAPTURE network consists of multiple nodes, including an origin, endpoint, zero or more ordered vehicles, and possibly some eavesdroppers, as shown in Fig. 4-3. Resources, such as photographs, are captured by the origin and relayed by hops to the endpoint. The network can operate in either an automatic selection mode where transmitted resources are automatically selected by the origin, or in a refinement mode. When in automatic selection mode, enough data is relayed for

the endpoint to reconstruct a low-quality preview of a resource. When the origin learns that the endpoint has received enough segments to reconstruct a preview, it will automatically select a new resource for transmission. Since the origin waits until the endpoint has confirmed reception before moving on to a new resource, more data may be transmitted than is required to generate a preview. The endpoint, typically a manned surface ship, can request that the network instead operate in refinement mode. In refinement mode, data continues to be transmitted for a specific, previously-transmitted resource, selected by the endpoint. The origin and hops will relay additional data from that resource until the network is put back into automatic selection mode by the endpoint.

4.6 Network Protocol

CAPTURE uses two types of network messages to communicate information between nodes: Chunk and Control messages. The bulk of traffic in a CAPTURE network consists of Chunk messages.

4.6.1 Chunk Messages

Even after compression, resources will likely be too large for transmission by today's acoustic modems, and thus must be broken into segments. Chunk messages consist of a single segment of data, along with the identifier for the segment's position within the resource, and a unique identifier for the resource itself. Chunk messages are designed to stand alone—any vehicle receiving a message can uniquely identify the resource the segment belongs to, and the segment's position within the resource, without any additional knowledge. Segments are of a fixed size, which must be agreed upon within the network before deployment. The segment size should be based on the Maximum Transmission Unit (MTU) supported by the modem hardware. For the WHOI MicroModem, this could be 256, 512 or 2048 bits depending on the level of error correction that is applied. In plain English, an example Chunk message could be: *"The 4th segment of SeaBED's 33rd resource consists*

of the following data ...". Control messages, sometimes referred to as acknowledgements, are significantly more complicated.

4.6.2 Control Messages

The second type of message used in a CAPTURE network is a Control message. Control messages contain a variety of data used to synchronize the state of the network between nodes - including acknowledgement and routing information. Control messages include the current resource identifier being transmitted by the network, just like Chunk messages, but otherwise serve a different purpose. Each network node tracks the segments, for each resource, that it knows each other node to possess. The primary purpose of control messages is to convey partial estimates of these 'segment masks' between CAPTURE nodes, acting as a selective acknowledgement. In particular, the message indicates the highest known index of the endpoint's contiguously received segments, and encodes a bitmask indicating the segments beyond that which are known to be possessed by network hops or the endpoint. Control messages also identify whether the network is operating in refinement mode or automatic selection mode. One possible control message might be: *"The route consists of SeaBED, vehicle A, vehicle B, and the endpoint. SeaBED's 33rd resource is being refined by request from the surface. The endpoint has received the first 9 contiguous segments. Beyond the 9th segment, the hops and endpoint are known to have received the following segments: ...".*

Control messages also include the current route from the origin to the endpoint, and a revision ID. The endpoint can alter this route or select a different vehicle as the origin by incrementing the revision ID. The route consists of the hardware ID's, in order, for the nodes currently in the network: $\langle origin, hop_A, ..., endpoint \rangle$. The overhead of this routing information would be substantial in traditional networks, but adds minimal overhead for small numbers of vehicles. For networks with small numbers of vehicles, a single network node can be identified by a few bits, and routes can be expressed in a byte or two.

4.6.3 Message Handling

Since the ocean is a broadcast medium, messages may 'skip' any individual hop in a network, or even be communicated directly from origin to endpoint. There is no guarantee or requirement that each message be communicated along every node in the route. When any message is received, some components of a message may be ignored depending on the source of the message. In particular, some data is not assumed to be valid unless it comes from *upstream*, closer to the origin, or *downstream*, closer to the endpoint. For example, both Chunk and Control messages contain a resource ID. If the network is believed to be in automatic selection mode, that resource ID is taken to be the currently active resource only if it came from upstream. On the other hand, if the network is in refinement mode, the resource ID will be taken as the active ID only if it came from downstream. This allows the origin to control the transmission of automatically selected resources, yet also propagates resources requested from the surface towards the origin when operating in refinement mode.

When a Chunk message is received, the data segment it contains is stored at the appropriate offset in the local copy of the resource. The receiving node also stores that the transmitter has the segment.

Any node receiving a Control message first incorporates the included segment masks into their own segment mask. If the message was transmitted by the immediate downstream neighbor, the current autonomy mode is also stored from the message. Finally, if the route revision in the message is higher than that of the currently stored route, the local copy of the routing information is updated.

4.6.4 Transmission Scheduling

Which messages are transmitted by a network node depend upon the node's role in the CAPTURE network, as shown in Table 4.1 below.

When transmitting a Chunk message, the segment masks for downstream nodes should be used to select what is transmited. Early resource segments that have not

	Chunk	Control
Endpoint		×
Hop	×	×
Origin	×	
Eavesdropper	×	×

Table 4.1: Message types transmitted by each of the four node classs.

been received by any nodes closer to the endpoint are the highest priority. In particular, nodes should start by transmitting the earliest segments for the active resource that a downstream node is not believed to possess, and continue in-order transmission of any later segments not held by downstream nodes. When a Control message is received from a downstream node, this process starts over by transmitting the earliest segment now known to not be received.

Using the simulation parameters described in Section 2.3, Fig. 4-4 illustrates the sensitivity of a CAPTURE network to how frequently acknowlegement messages are transmitted relative to segments of data. When the ratio is high and acknowledgements are sent infrequently, the odds of too much data being transmitted before moving on are high. However, transmission rates seems to be relatively insensitive to this scheduling for values near the minima of one control message for every four to eight chunk messages.

4.6.5   Implementation

Each of the autonomous platforms had a platform driver developed to fit the needs of their specific software environments. A number of revisions to Goby[100] were made as part of this work, which allowed it to be used as a software abstraction layer for the acoustic modem on each vehicle. These revisions have now been incorporated into Goby v2.0. The implementation of the CAPTURE network protocol relied on two packed message types, representing the Chunk and Control messages. These messages were constructed as 512-bit messages, to fit the requirements of the physical layer. The specific message definitions that were used are shown in Fig. 4-5 and Figure 4-6.

99
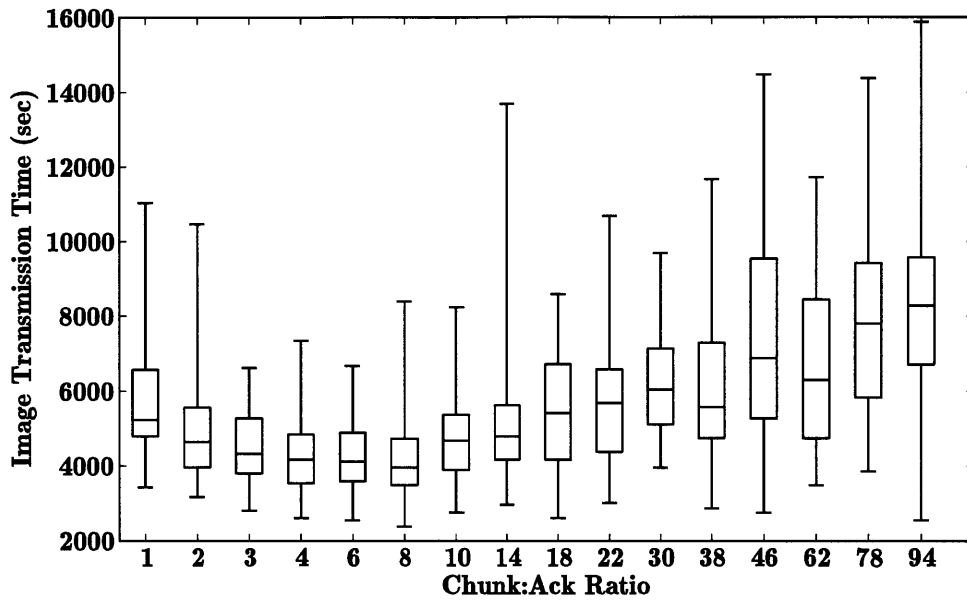
(a) Per-Image Time
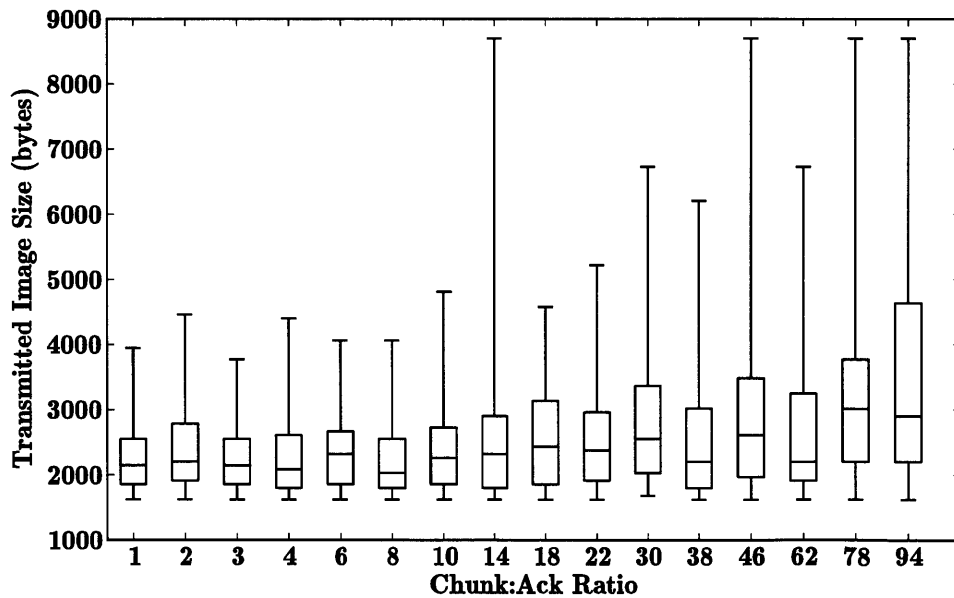
(b) Preview Size

Figure 4-4: Illustration of network sensitivity to the ratio of chunk messages to control messages. Fig. 4-4a shows the mean time required to receive each resource preview for a fixed preview size of 1600 bytes. Fig. 4-4b shows the final average size of the previews. In total, the link was 12km long, and consisted of five evenly spaced nodes (four hops).

0	8	16	24	31
Message ID	Time of Launch	Resource Origin / ID		
...	Segment ID		56 Byte Segment...	
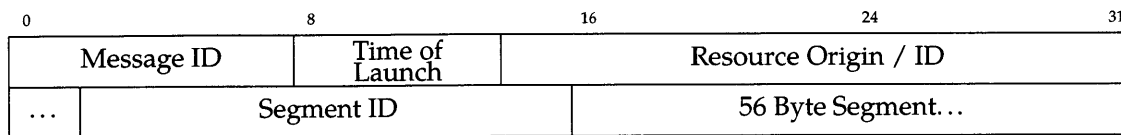
Figure 4-5: Definition for Chunk messages used during 2011 field experiments in Buzzards Bay. The numerical scale across the top displays the number of bits. Each subsequent row represents additional bits which continue from the row above.

0	8	16	24		31
Message ID	Time of Launch	RID	Route	Mode	...
...Resource Origin / ID			End-to-end Reception Count		
54 Byte Acknowledgement Masks ...					
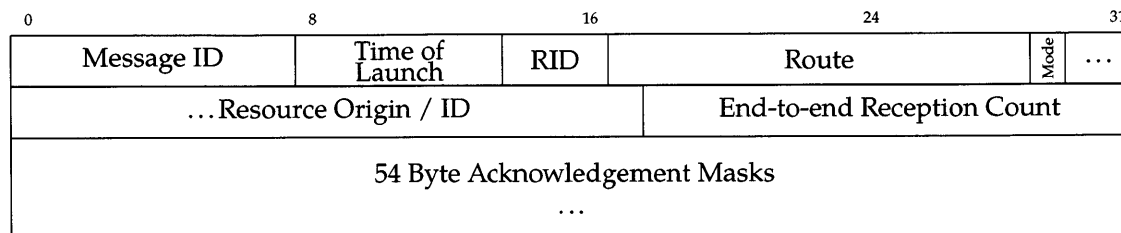
Figure 4-6: Definition for Control messages used during 2011 field experiments in Buzzards Bay. The numerical scale across the top displays the number of bits. Each subsequent row represents additional bits which continue from the row above.

Since the entire route is encoded in the control packet, which currently is twelve bits long (plus three to allow changing the route), this implementation supports routes containing up to four vehicles, and networks containing seven vehicles in total. This could easily be expanded for longer routes, consuming only a few additional bits.

The Chunk and Control messages both contain a time-of-launch field, allowing the second of transmission to be encoded in a message. All vehicles in the Buzzard's Bay experiment were equipped with a high precision, low-drift clock [33]. By synchronizing each vehicle's clock at the surface, all nodes can passively measure the one-way-travel-time (OWTT) of each acoustic broadcast by simply comparing the encoded time-of-launch and the observed time-of-arrival. Since the sound speed profile is well known in water, the inter-vehicle range can be easily computed. Over time, vehicles within the network can augment each other's navigation estimates using these additional range constraints.

101

CHAPTER 5

Field Results

CAPTURE has been field tested in three distinct experiments, and four different network configurations, as shown in Fig. 5-1. All told, these experiments involved six distinct autonomous platforms, including two different SeaBED AUVs, two different OceanServer Iver AUVs, and a Bluefin 9 AUV. In addition, four manned surface ship platforms have been used, involving researchers from NOAA, MIT, WHOI, Northeastern University, University of Michigan, and Bluefin Robotics Corporation.

In February of 2010, an early version of the CAPTURE architecture was tested on Lucille, a SeaBED-class[104] AUV owned by NOAA, during a research expedition aboard the NOAA Ship Oscar Elton Sette. A single dive was performed near Rota, an island in the Northern Marianas Archipelago[72], ranging in depth between 100 and 350 meters. No specific constraints were put on the surface ship, which remained within 600 meters of the vehicle throughout the dive.

In late May of 2011, CAPTURE was extended to operate on a Bluefin 9 AUV equipped with a "backseat driver" computation stack running the MOOS software suite. That vehicle is part of ongoing Mine Counter-Measures development, seeking to identify seafloor mine-like objects and transmit their sonar signatures to the surface for confirmation[73].

In August of 2011, CAPTURE was tested on three autonomous platforms and one manned platform operating simultaneously. Two OceanServer Iver AUVs with payload and navigation suites custom-developed by the University of Michigan[32] provided long-range mid-water-column survey capability, while a SeaBED AUV provided the ability to capture detailed low-altitude photographic surveys. A pho-
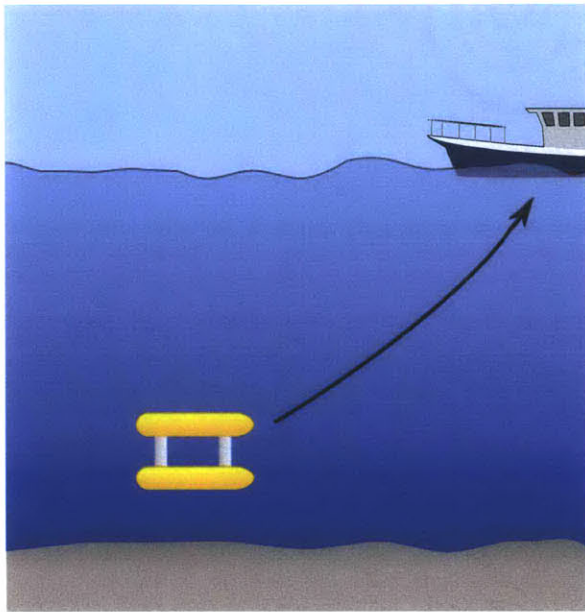
(a) One Hop
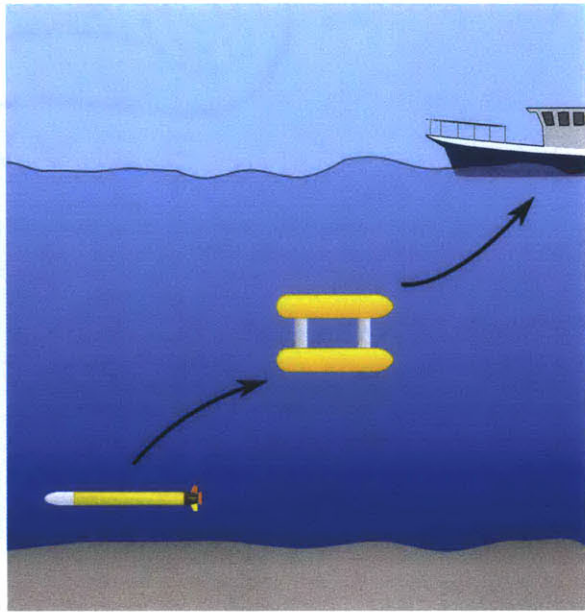
(b) Two Hop
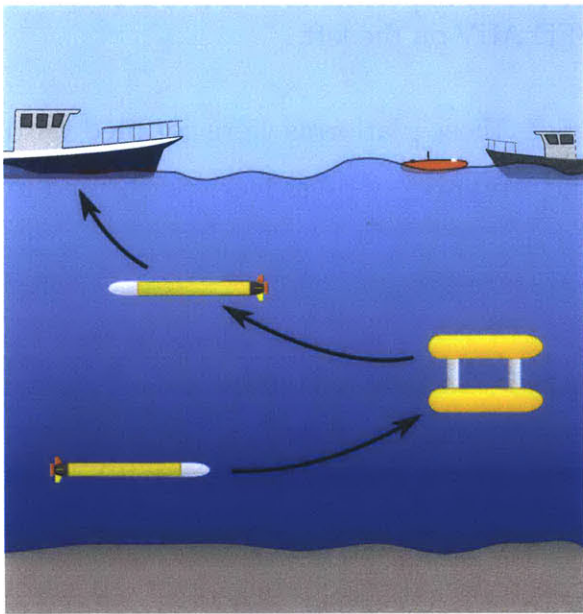
(c) Three Hop
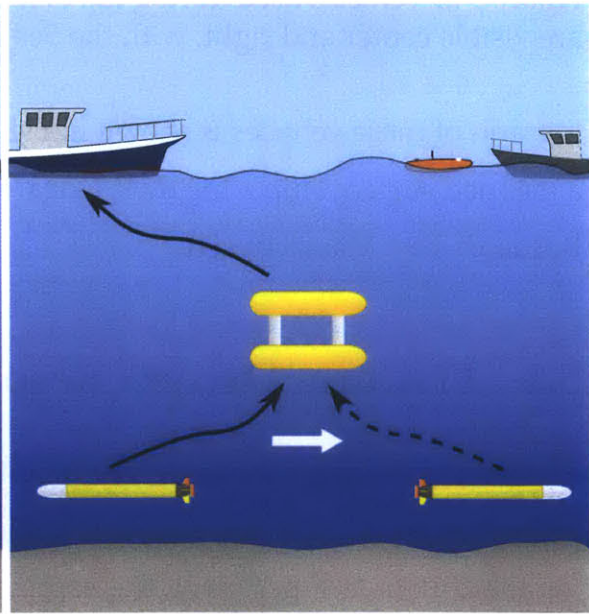
(d) Route Switch

Figure 5-1: Network configurations which have successfully been used in the field with CAPTURE. In the fourth example, the vehicle responsible for initiating transmissions was changed mid-dive, in response to a request from the surface.

Figure 5-2: Vehicles used during the CAPTURE '11 Experiment. The two Iver AUVs are visible center and right, with the SeaBED AUV on the left.

tograph of these vehicles is shown in Fig. 5-2. These platforms were coupled with a manned surface ship – the R/V Tioga, and a number of dives were performed in Buzzards Bay, Massachusetts.

5.1 Platform Driver / Resource Acquisition

The Lucille AUV used during the 2010 field experiment is equipped with a five megapixel Prosilica color camera, featuring a CCD with high dynamic range. During the 2010 field experiment, this camera captured one color image every five seconds at a resolution of 2048×2048 pixels. Those raw, Bayer RGGB encoded, images were processed and converted to the Y'UV colorspace onboard the AUV's main control computer, resulting in 1024×1024 pixel square full color images.

The Bluefin 9 AUV used during the brief mine counter-measure experiment is equipped with a MarineSonic sidescan sonar system, which generates 2D imagery

Figure 5-3: Bluefin 9 AUV prior to deployment, and the sonar imagery transmitted during the dive.

in a proprietary TIFF-like format after a fixed number of scanlines. A platform driver was developed to support reading the imagery from the sonar, and to interface with the onboard MOOS autonomy software. Goby software was used to abstract the interface with the on-board WHOI MicroModem. During a very short mission, there was time to transmit a single sonar image to the surface from the AUV, shown in Fig. 5-3.

5.2   Resource Prioritization

To date, our field experiments have relied on a single-resource queuing model to identify the next resource for transmission. The Lucille AUV used during the 2010 field experiment, and shown in Fig. 5-4, has a single onboard CPU used for both CAPTURE and vehicle control. To minimize the risk of overloading the onboard
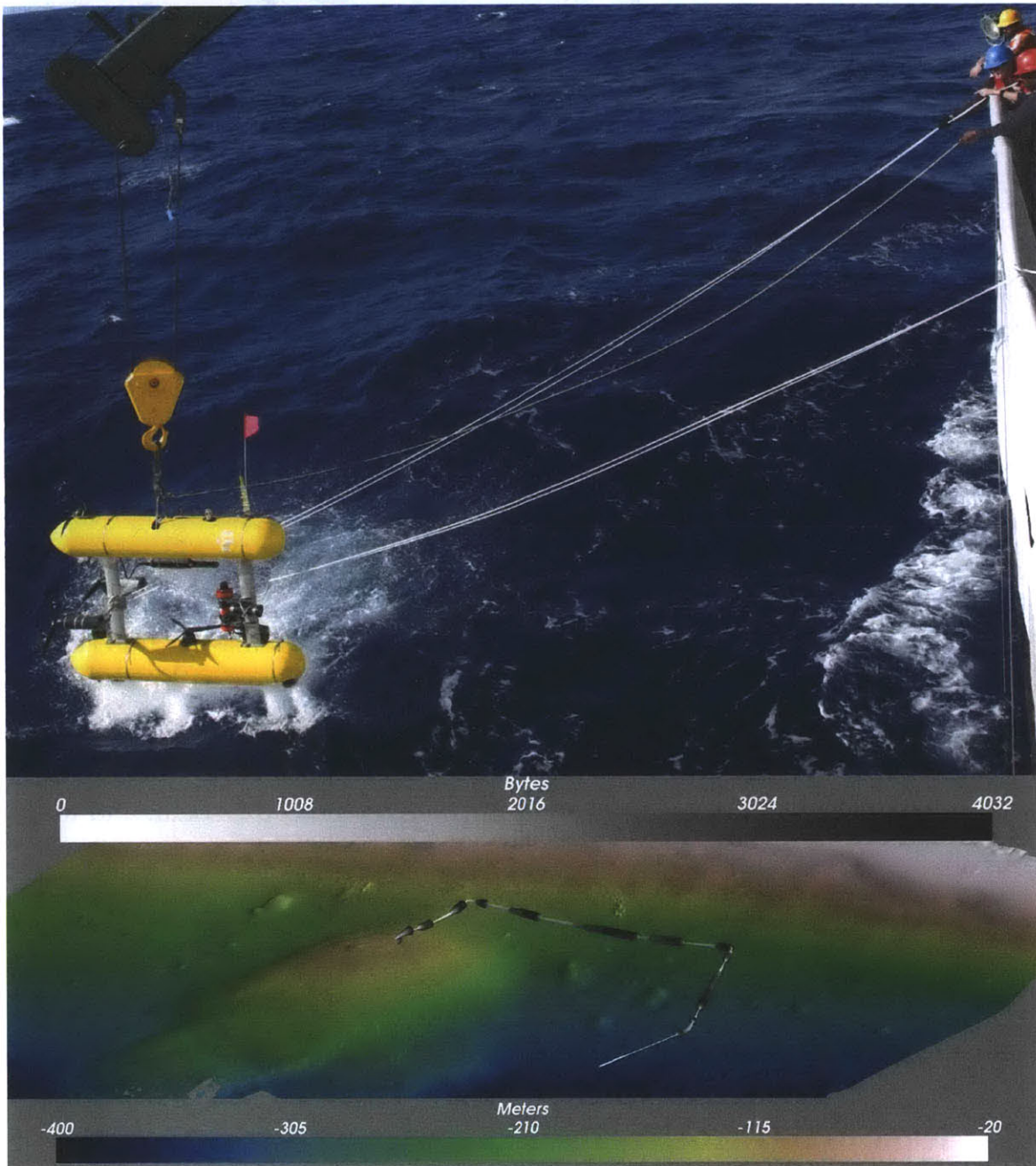
Figure 5-4: *Top*: Lucille, a SeaBED AUV, prior to launch near Rota, 2500km south of Tokyo. *Bottom*: Transmission progress overlaid on bathymetry.

CPU's limited resources, the most recently captured photograph was compressed every three minutes. This led to several images being compressed but not transmitted, but ensured that new data was always available for transmission. When CAPTURE was ready to transmit a new resource to the surface, the most recently compressed new image was selected for transmission.

5.3 Progressive Encoding

The photographic and sonar imagery acquired by the Lucille and Bluefin AUV respectively, were compressed using SPIHT compression, in conjunction with the Cohen-Daubechies-Feauveau 9/7 wavelet[21]. The MarineSonic sonar source imagery was a grayscale image of 1024×960 pixels in a proprietary format. For the color photographic imagery captured by the Lucille AUV, 50% of the encoded data stream was allocated to luminance data, and 50% to chrominance data. In retrospect, allocating a higher proportion to luminance data would have resulted in more visually pleasing imagery.

During the 2010 field experiment, a total of fifteen color photographs were received over the course of a 3.75 hour period. Of the fifteen successfully received images, four were captured during descent or ascent and were completely black as a result. The eleven non-black images received are shown in Fig. 5-5 and 5-6. The fifteen images were transmitted over a 3.75 hour period, resulting in about fifteen minutes per image, or approximately 35 bits per second achieved. While this low number is largely due to packet loss and scheduling in real-world conditions, the modem also varied the level of forward error correction it applied, between encodings with maximum theoretical burst rates of 520 and 5400 bits per second, to obtain richer statistics on transmission success.

During the August 2011 CAPTURE field experiment, extremely murky water conditions prevented capturing photographs, and pre-captured imagery was used instead. In addition, one test was performed with a non-progressively encoded dataset. A short segment of audio, Neil Armstrong's first words on the surface of

Figure 5-5: The first six color images captured by the SeaBED-class AUV, compressed in-situ, and transmitted to surface operators.
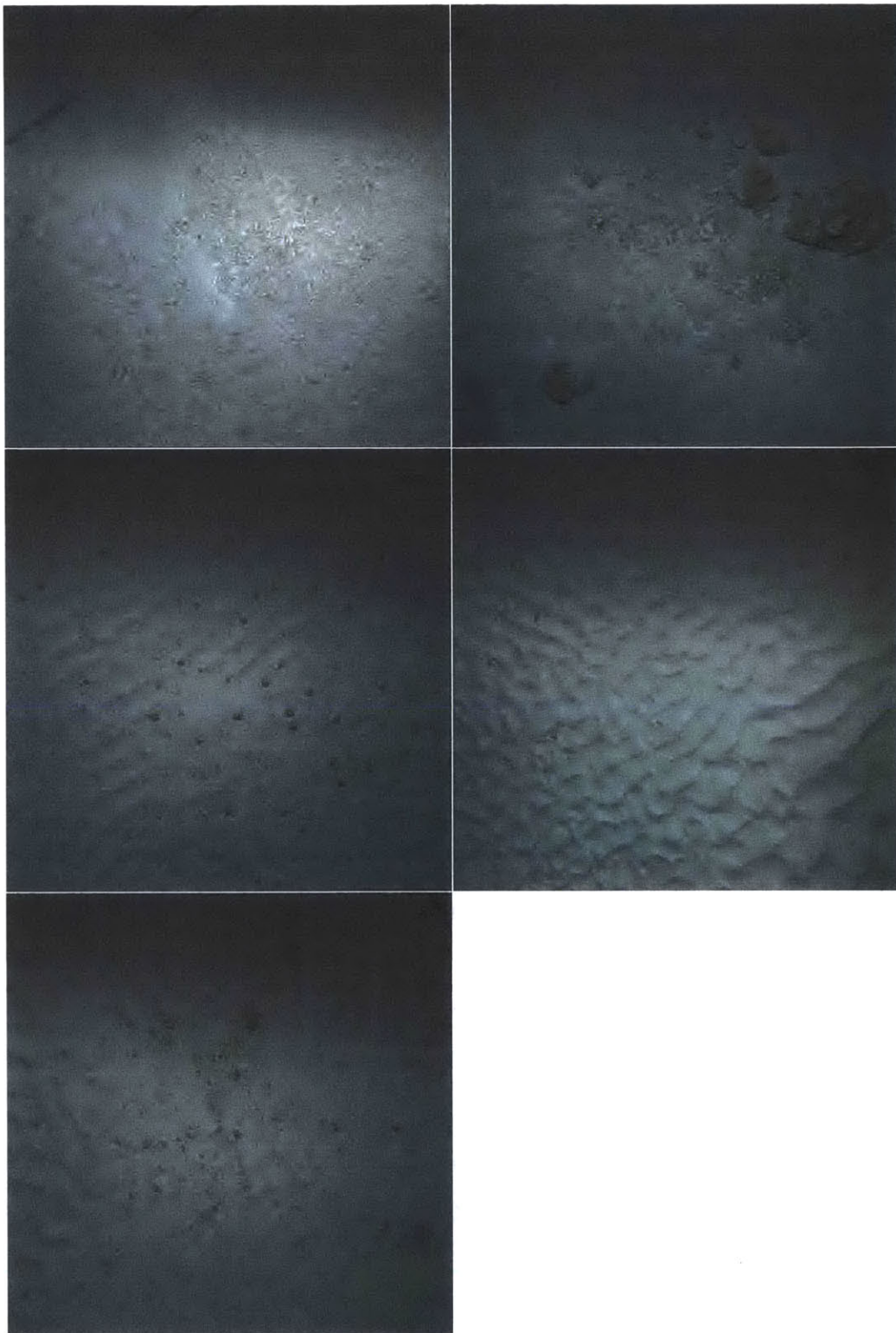
Figure 5-6: The final five non-black images returned by the SeaBED vehicle.

the moon, was compressed with the Speex voice codec to 4368 bytes. The audio was then encrypted using AES with a 256 bit key. Once the full set of encoded packets had been received, the audio was decoded and successfully played.

5.4 Relay Communication

(a) 41 Segments   (b) Log. difference   (c) 97 Segments
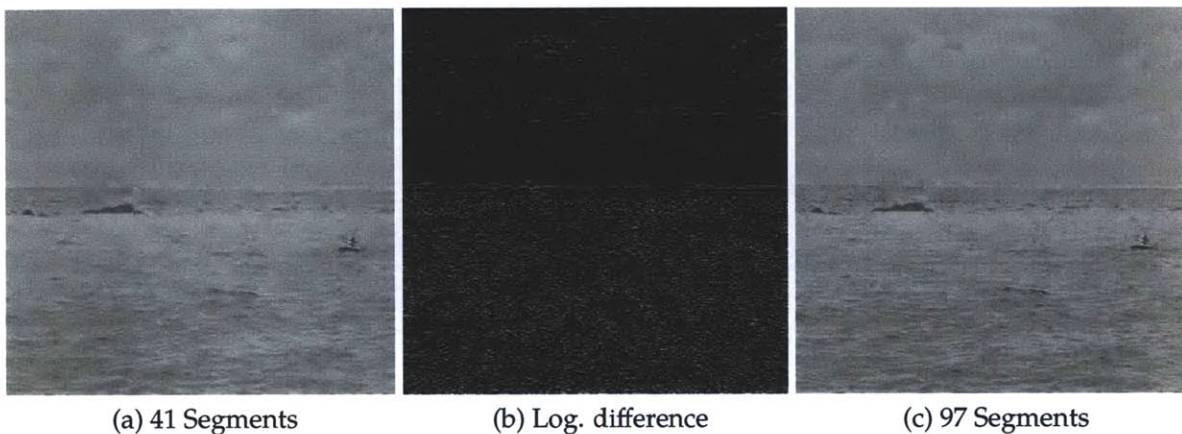
Figure 5-7: A transmitted grayscale photo prior to, and after requesting additional refinement. The difference in magnitude is shown between the two versions on a logarithmic scale to highlight changes.

Three separate successful CAPTURE dives were performed during the most recent field experiment, each testing different capabilities of the networking protocol. During one trial, data was communicated across a two-hop network as shown in Fig. 5-1b. After six preview images were sequentially transmitted as 2048 byte previews, the fourth transmitted image was identified by the surface operator as warranting further refinement. Upon request, the transmitting vehicle went back and provided additional data to refine the image, as shown in Fig. 5-7. The origin eventually transmitted 5529 contiguous bytes of the image before being instructed to return to automatic selection.

A total of seven images were eventually transmitted, each decoded progressively, with gradually improving reconstructions over the course of the transmission. CAPTURE was also tested in the three-hop linear network, depicted in Fig. 5-1c, successfully relaying four images across the heterogeneous network.

In the final experiment, a two-hop network was employed as shown in Figure 5-1d. Four grayscale images were transmitted from an Iver AUV, via a SeaBED AUV, to the surface. The surface operator then requested a route change, granting the other Iver AUV the responsibility for transmitting resources. That vehicle transmitted the pre-loaded encrypted speech, followed by another two grayscale photos.

CHAPTER 6

Discussion

Relaying high-resolution scientific data from submerged AUVs over long horizontal distances faces numerous obstacles. In this thesis I have presented an analysis of these obstacles, along with the design of a unified solution in the form of CAPTURE. A networking and compression infrastructure, CAPTURE supports transmission and interactive refinement of sonar and photographic data, along with scalar environmental measurements.

6.1 Contributions

The specific contributions and characteristics of this work can be divided into those relating to compression and data selection, and those related more directly to the CAPTURE networking architecture. To demonstrate the viability of this architecture, I presented both simulated results and real-world results from running CAPTURE software in three field experiments, in diverse environments, on SeaBED, OceanServer and Bluefin AUVs, each employing significantly different software architectures.

6.1.1 Compression

While wavelets have previously been recognized as appropriate for underwater image coding, there are no known examples of using wavelet-based source coding for scalar telemetry. This work additionally represents the first application of fully embedded encodings to AUV telemetry, allowing decoding to halt after any number of contiguous packets while still producing the highest quality reproduction for that

112

number of bits. I also presented a novel method for highlighting data regions of interest prior to compressing data with these wavelet coders.

To fill the capability gap between transmitting individual images and transmitting summary statistics, this thesis presents a novel compression technique based on image synthesis. This strategy provides a very compact representation for imagery, utilizing inter-image redundancy while communicating both the visual 'gist' of an image in texture space and allowing computation of texture statistics on the surface.

CAPTURE takes a hybrid approach to data selection, incorporating both autonomous prioritization and feedback from human operators. There are clear opportunities to incorporate high-level autonomy algorithms during the data selection process. New resources are automatically selected for compression and transmission to the surface barring specific requests from human operators. As humans are able to view the data being collected by an AUV in real time, they are better enabled to recognize anomolies and features of interest. Human feedback is explicitly incorporated into CAPTURE by allowing the identification of scientifically valuable images or data segments for additional refinement, and allowing refinement up to an arbitrarily high-quality reconstruction. This work represents the first example of human-driven data-quality selection for scalar AUV telemetry of underwater AUVs.

6.1.2 Networking

The ocean imposes challenges on underwater networks including high latency, intermittent communication, the lack of instantaneous end-to-end connectivity, and a broadcast medium. This thesis uniquely employs a strategy of comprehensive data storage at every node and a broadcast-based selective acknowledgement protocol to combat these challenges. Relying on a "store and forward" architecture, rather than a simple relay chain, increases the performance of the relay link in poor conditions.

Most underwater networking research assumes the use of relatively low-cost,

and low-complexity, static communication nodes. These nodes may be augmented by one or two AUVs, but the fixed nodes represent the backbone of the network. Fixed seafloor nodes cost 5-10 times less than even low-cost AUVs, yet untethered AUV are the most practical option for accessing some environments. This thesis presents an approach targeted to very small relay networks of AUVs, using the small network size as an advantage.

6.2 Future Work and Limitations

Geographic routing

Recently there has been significant interest in geographically aware routing protocols[131, 135], which learn and construct routing tables based on known node locations. To that end, the one-way-travel-time capabilities in the current implementation of CAPTURE allow every vehicle to determine the range from any overheard transmitter with each transmission. That information is not currently used, but could be used in the future to aid in routing.

Incorporate network coding

When using random linear network coding[64][18][63], instead of a single packet being transmitted, a linear combination of several packets is transmitted along with the linear coefficients. After enough of these random packets have been received, it becomes possible to decode all of the original packets that were linearly combined. Used strategically, this decreases the frequency of ARQ required.

Facilitate handling of multiple data sources

While there are no architectural barriers to transmitting data from multiple data sources, the current implementation does not provide any way to multiplex or differentiate between multiple sensors. There are clear opportunities for both automatic and user-guide approaches to selection of a data source – as well as opportunities to incorporate more advanced data interest detectors

114

on a per-sensor basis, such as image recognition algorithms.

Preview size selection

In the current implementation of CAPTURE, the size of the preview transmitted before the origin moves to a new resource is defined by the resource type. Incorporating automatic selection of the preview size would allow the origin to transmit higher quality previews for images that it believes have a higher likelihood of being interesting. Simple image statistics such as entropy or luminance would likely be sufficient for this purpose.

User-driven Region of Interest selection

SPIHT does not include any explicit information in the bitstream about which wavelet coefficients are being encoded, making it challenging to refine only specific regions of an image. Other embedded wavelet compressors, including WDR[120], do support this capability at a small cost to image quality.

Artifact Reduction for Image Synthesis

Transmitting texture information as a low-resolution grid results in blocking artifacts in the reconstructed image. Additionally, texture class boundaries are currently handled no differently than constant-texture areas during image synthesis. Explicit handling of class boundaries could improve the second issue, and the former could be improved by denser encoding of texture information. There is also a body of literature on blocking artifact reduction in Vector Quantization that may be relevant.

Progressive Synthesis Mask Transmission

At an average of 40 bytes, individual texture masks are easy to transmit in a single transmission. For higher resolution masks, with a larger number of texture classes, the size of the texture mask could grow significantly. As it increases in size, a progressive transmission scheme for individual texture class masks based on shape-adaptive SPIHT coding techniques (e.g. [66]) could be worth investigating.

Synthesis/Wavelet Hybrid Image Compression

Image Synthesis is effective at encoding repetitive background textures (like sand), but does not encode objects, or unrecognized texture areas. A hybrid approach could be developed that uses Image Synthesis to compress the background of each image, and embedded wavelet compression to encode any 'significant' foreground objects, based on some significance metric.

6.3 Modem Suggestions

Although most testing and development took place on the WHOI Micro-Modem, this work is designed to be independent of the specific physical layer used for communication. The modem provides as robust a physical layer as can be hoped given the underwater environment, yet the modem's interface to higher-level applications imposes several limitations. Many of these limitations stem from the age of the underlying hardware, which is in the process of a major revision. Three key changes that would make the Micro-Modem easier to use with AUVs are:

- A greater independence between MTU and FEC

- The ability to include custom header metadata in packets

- A "raw" modulation mode and "best-effort" decoding

The Maximum Transmission Unit of the WHOI Micro-Modem depends upon the level of FEC selected for channel coding. Depending on the specific level of coding selected, the MTU may be 32, 64, or 256 bytes. This variability makes it challenging to select an FEC level based solely on the current channel quality, as it imposes constraints on the higher level network layers. Indeed, 32 bytes is a relatively small MTU, and imposes fragmentation on all but the most trivial of messages. Even with 64 byte frames, a few bytes of header metadata stands out as a significant cost. In [7] an optimal packet size is derived for multi-hop relay networks, based upon simulated results with two realistic MAC protocols. The optimum size varies with the

116

specific simulation parameters, but is generally a few hundred to a few thousand bytes.

Packet transmissions from the Micro-Modem contain a heavily protected meta-data header, including a 7-bit source ID and 7-bit destination ID. Even when the bulk of a message is lost, this header information frequently remains decodable. The ability to include custom heavily protected metadata in this header would allow routing information and other critical metadata to receive protection, even when the FEC applied to the rest of the frame is low.

When a packet is received by the Micro-Modem a CRC, or "checksum", is computed. If the computed checksum does not match one encoded in the packet, the packet is dropped and not presented to the user. This makes it challenging to apply error correction customized to your dataset (though not impossible, as described further below). If the modem reported a best-estimate decoding in the case of failed CRC's, possibly along with hard decisions from the equalizer, FEC could be applied in software as well as in the modem firmware.

As a direct result of this work, each of these interface suggestions is being actively considered for incorporation into the next version of the WHOI Micro-Modem, and some are already being implemented.

Bibliography

[1] Ian F. Akyildiz, Dario Pompili, and Tomasso Melodia. Underwater Acoustic Sensor Networks: Research Challenges. In *Ad Hoc Networks*, volume 3, pages 257 – 279. Elsevier, March 2005.

[2] Douglass A. Alexander, Robert G. Deen, Paul M. Andres, Payam Zamani, Helen B. Mortensen, Amy C. Chen, Michael K. Cayanan, Jeffrey R. Hall, Vadim S. Klochko, Oleg Pariser, Carol L. Stanley, Charles K. Thompson, and Gary M. Yagi. Processing of mars exploration rover imagery for science and operations planning. *J. Geophys. Res.*, 111(E2):E02S02–, February 2006. ISSN 0148-0227. URL http://dx.doi.org/10.1029/2005JE002462.

[3] Camilla S. Andresen, Fiammetta Straneo, Mads Hvid Ribergaard, Anders A. Bjork, Thorbjorn J. Andersen, Antoon Kuijpers, Niels Norgaard-Pedersen, Kurt H. Kjaer, Frands Schjoth, Kaarina Weckstrom, and Andreas P. Ahlstrom. Rapid response of helheim glacier in greenland to climate variability over the past century. *Nature Geosci*, 5(1):37–41, January 2012. ISSN 1752-0894. URL http://dx.doi.org/10.1038/ngeo1349.

[4] Roy A. Armstrong, Hanumant Singh, Juan Torres, Richard S. Nemeth, Ali Can, Chris Roman, Ryan Eustice, Lauren Riggs, and Graciela Garcia-Moliner. Characterizing the deep insular shelf coral reef habitat of the hind bank marine conservation district (us virgin islands) using the seabed autonomous underwater vehicle. *Continental Shelf Research*, 26(2):194–205, February 2006.

[5] A. Baggeroer. Acoustic telemetry–an overview. *Oceanic Engineering, IEEE Journal of*, 9(4):229 – 235, oct 1984. ISSN 0364-9059. doi: 10.1109/JOE.1984. 1145629.

[6] A. Balasubramanian, B.N. Levine, and A. Venkataramani. Replication routing in DTNs: A resource allocation approach. *Networking, IEEE/ACM Transactions on*, 18(2):596 –609, april 2010. ISSN 1063-6692. doi: 10.1109/TNET. 2009.2036365.

[7] S. Basagni, C. Petrioli, R. Petroccia, and M. Stojanovic. Choosing the packet size in multi-hop underwater networks. In *OCEANS 2010 IEEE - Sydney*, pages 1 –9, may 2010. doi: 10.1109/OCEANSSYD.2010.5603664.

[8] P.-P. Beaujean and E. Carlson. Combined vehicle control, status check and high-resolution acoustic images retrieval using a high-frequency acoustic

modem on a hovering auv. In *OCEANS 2010*, pages 1 –10, sept. 2010. doi: 10.1109/OCEANS.2010.5664037.

[9] Michael Benjamin, Henrik Schmidt, Paul Newman, and John Leonard. Nested autonomy for unmanned marine vehicles with moos-ivp. *Journal of Field Robotics*, 27(6):834–875, November 2010.

[10] C. Benton, J. Kenney, R. Nitzel, R. Blidberg, S. Chappell, and S. Mupparapu. Autonomous undersea systems network (ausnet) - protocols to support ad-hoc auv communications. In *Autonomous Underwater Vehicles, 2004 IEEE/OES*, pages 83 – 87, june 2004. doi: 10.1109/AUV.2004.1431197.

[11] Vaduvur Bharghavan, Alan Demers, Scott Shenker, and Lixia Zhang. MACAW: A medium access protocol for wireless LAN's. In *Proc. ACM SIG-COMM Conference (SIGCOMM '94)*, pages 212–225, August 1994.

[12] Andy Bowen, Dana Yoerger, and Louis Whitcomb. Hybrid ROV for 11,000 Meter Operations. In *Symposium on Underwater Technology and Workshop on Scientific Use of Submarine Cables and Related Technologies*, April 2007.

[13] Leonid M. Brekhovskikh and IUP Lysanov. *Fundamentals of ocean acoustics*. AIP series in modern acoustics and signal processing. AIP Press/Springer, 2003. ISBN 9780387954677. URL http://books.google.com/books?id= q5dT1t88niQC.

[14] Robert W. Buccigrossi and Eero P. Simoncelli. Image compression via joint statistical characterization in the wavelet domain. *Image Processing, IEEE Transactions on*, 8(12):1688 –1701, dec 1999. ISSN 1057-7149. doi: 10.1109/ 83.806616.

[15] Nathalie A. Cabrol, Edmond A. Grin, Michael H. Carr, Brad Sutter, Jeffrey M. Moore, Jack D. Farmer, Ronald Greeley, Ruslan O. Kuzmin, David J. Des-Marais, Marc G. Kramer, Horton Newsom, Charles Barber, Ivan Thorsos, Kenneth L. Tanaka, Nadine G. Barlow, David A. Fike, Mary L. Urquhart, Brian Grigsby, Frederick D. Grant, and Olivier de Goursac. Exploring gusev crater with spirit: Review of science objectives and testable hypotheses. *J. Geophys. Res.*, 108(E12):8076–, December 2003. ISSN 0148-0227. URL http://dx.doi.org/10.1029/2002JE002026.

[16] Richard Camilli and Anthony Duryea. Characterizing marine hydrocarbons with in-situ mass spectrometry. In *Oceans 2007, Proceedings of the MTS/IEEE*, pages 1–7, Oct 2007.

[17] N. Chirdchoo, Wee seng Soh, and Kee Chua. RIPT: A receiver-initiated reservation-based protocol for underwater acoustic networks. *Selected Areas in Communications, IEEE Journal on*, 26(9):1744 –1753, december 2008. ISSN 0733-8716. doi: 10.1109/JSAC.2008.081213.

[18] Mandar Chitre and Wee-Seng Soh. Network coding to combat packet loss in underwater networks. In *Proceedings of the Fifth ACM International Workshop on UnderWater Networks*, WUWNet '10, pages 5:1–5:6, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0402-3. doi: http://doi.acm.org/10.1145/1868812. 1868817. URL http://doi.acm.org/10.1145/1868812.1868817.

[19] Mandar Chitre, Shiraz Shahabudeen, Lee Freitag, and Milica Stojanović. Recent advances in underwater acoustic communications & networking. In *Proceedings of the IEEE Oceans 2008 Conference*, Quebec City, QC, September 2008.

[20] R.F.W. Coates. *Underwater acoustic systems*. Macmillan new electronics series. Macmillan, 1990. URL http://books.google.com/books?id=ju89AQAAIAAJ.

[21] A. Cohen, Ingrid Daubechies, and J.-C. Feauveau. Biorthogonal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, 45(5):485–560, 1992. ISSN 1097-0312. doi: 10.1002/cpa.3160450502. URL http://dx.doi.org/10.1002/cpa.3160450502.

[22] T. Collins and P. Atkins. Error-tolerant SPIHT image compression. *Vision, Image and Signal Processing, IEE Proceedings*, 148(3):182 –186, jun 2001. ISSN 1350-245X. doi: 10.1049/ip-vis:20010376.

[23] T. Collins, P.R. Atkins, and J. Davies. Video transmission through a shallow underwater acoustical channel. In *Data Compression: Methods and Implementations, IEE Colloquium on*, pages 6/1–6/4, 1999.

[24] R. A. DeVore and B. J. Lucier. Wavelets. In *Acta Numerica 1*, pages 1–56. Cambridge University Press, 1992.

[25] E.R. Dougherty. *An Introduction to Morphological Image Processing*. SPIE Optical Engineering Press, 1992.

[26] Christiane N. Duarte, Gerald R. Martel, Christine Buzzell, Denise Crimmins, Rick J. Komerska, Sai S. Mupparapu, Steven G. Chappell, D. Richard Blidberg, and Rober Nitzel. a common control language for dynamic tasking of multiple autonomous vehicles. In *Proc. of the 14th Intern. Symp. on Unmanned Untethered Submersible Technology UUST'05*, Durham, NH, aug 2005.

[27] M. Dunbabin, P. Corke, I. Vasilescu, and D. Rus. Data muling over underwater wireless sensor networks using an autonomous underwater vehicle. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2091 –2098, may 2006. doi: 10.1109/ROBOT.2006. 1642013.

[28] Robert L. Eastwood, Lee E. Freitag, and Josko A. Catipovic. Compression techniques for improving underwater acoustic transmission of images and data. In *Proceedings of MTS/IEEE OCEANS 1996: Prospects for the 21st Century*, Fort Lauderdale, FL, September 1996.

[29] Peter J. Edmunds and Robin Elahi. The demographics of a 15-year decline in cover of the caribbean reef coral montastraea annularis. *Ecological Monographs*, 77(1):3–18, feb 2007. ISSN 0012-9615. doi: 10.1890/05-1081. URL http://dx.doi.org/10.1890/05-1081.

[30] Alexei A. Efros and William T. Freeman. Image quilting for texture synthesis and transfer. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '01, pages 341–346, New York, NY, USA, 2001. ACM. ISBN 1-58113-374-X. doi: http://doi.acm.org/10.1145/383259. 383296. URL http://doi.acm.org/10.1145/383259.383296.

[31] B. Elder, A.D. Bowen, M. Heintz, M. Naiman, C. Taylor, W. Seller, J.C. Howland, and L.L. Whitcomb. Jason 2: a review of capabilities. In *EOS: Trans. Amer. Geophysical Union Fall Meeting Supplement*, 2003. Abstract.

[32] Ryan M. Eustice, Hunter C. Brown, and Ayoung Kim. An overview of auv algorithms research and testbed at the university of michigan. In *Proceedings of the IEEE/OES Autonomous Underwater Vehicles Conference*, pages 1–9, Woods Hole, MA, October 2008. doi: http://dx.doi.org/10.1109/AUV.2008.5290531. In Press.

[33] Ryan M. Eustice, Hanumant Singh, and Louis L. Whitcomb. Synchronous-clock one-way-travel-time acoustic navigation for underwater vehicles. *Journal of Field Robotics, Special Issue on State of the Art in Maritime Autonomous Surface and Underwater Vehicles*, 28(1):121 – 136, Jan 2011.

[34] Kevin Fall. A delay-tolerant network architecture for challenged internets. In *SIGCOMM 2003*, Karlsruhe, Germany, aug 2003.

[35] B.P. Foley, K. DellaPorta, D. Sakellariou, B. Bingham, R. Camilli, R. Eustice, D. Evagelistis, V. Ferrini, M. Hansson, K. Katsaros, D. Kourkoumelis, A. Mallios, P. Micha, D. Mindell, C. Roman, H. Singh, D. Switzer, and T. Theodoulou. New methods for underwater archaeology: the 2005 chios ancient shipwreck survey. *Hesperia*. Accepted, To Appear.

[36] Lee Freitag, Matt Grund, Sandipa Singh, James Partan, Peter Koski, and Keenan Ball. The WHOI Micro-Modem: An Acoustic Communcations and Navigation System for Multiple Platforms. In *Oceans 2005, Proceedings of the MTS/IEEE*, volume 2, pages 1086 – 1092, 2005. URL http://acomms.whoi.edu/60x%20Reports%20and%20Papers/MicromodemOceans2005.pdf.

[37] J.J Garcia-Luna-Aceves and Chane L. Fullmer. Floor acquisition multiple access (fama) in single-channel wireless networks. *Mob. Netw. Appl.*, 4:157–174, October 1999. ISSN 1383-469X. doi: http://dx.doi.org/10.1023/A:1019146831447. URL http://dx.doi.org/10.1023/A:1019146831447.

[38] Toby A. Gardner, Isabelle M. Côté, Jennifer A. Gill, Alastair Grant, and Andrew R. Watkinson. Long-term region-wide declines in caribbean corals.

Science, 301(5635):958–960, 2003. doi: 10.1126/science.1086050. URL `http://www.sciencemag.org/content/301/5635/958.abstract`.

[39] Y. Girdhar and G. Dudek. Online navigation summaries. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 5035 –5040, may 2010. doi: 10.1109/ROBOT.2010.5509464.

[40] Y. Girdhar and G. Dudek. ONSUM: A system for generating online navigation summaries. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 746 –751, oct. 2010. doi: 10.1109/IROS.2010.5650315.

[41] Y. Girdhar and G. Dudek. Online visual vocabularies. In *Computer and Robot Vision (CRV), 2011 Canadian Conference on*, pages 191 –196, may 2011. doi: 10.1109/CRV.2011.32.

[42] Y. Girdhar and G. Dudek. Offline navigation summaries. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 5769 –5775, may 2011. doi: 10.1109/ICRA.2011.5980094.

[43] Aditi Goel, Aiswariya G. Kannan, Ian Katz, and Radim Bartoš. Improving efficiency of a flooding-based routing protocol for underwater networks. In *Proceedings of the third ACM international workshop on Underwater Networks*, WuWNeT '08, pages 91–94, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-185-9. doi: http://doi.acm.org/10.1145/1410107.1410125. URL `http://doi.acm.org/10.1145/1410107.1410125`.

[44] Jill R. Goldschneider. *Lossy Compression of Scientific Data via Wavelets and Vector Quantization*. PhD thesis, University of Washington, Seattle, WA, June 1997.

[45] Matt Haag, Emmanuel Agu, Rick Komerska, Steven G. Chappell, and Radim Bartovš. Status packet deprecation and store-forward routing in ausnet. In *Proceedings of the 1st ACM international workshop on Underwater networks*, WUWNet '06, pages 86–92, New York, NY, USA, 2006. ACM. ISBN 1-59593-484-7. doi: http://doi.acm.org/10.1145/1161039.1161056. URL `http://doi.acm.org/10.1145/1161039.1161056`.

[46] David F. Hoag and Vinay K. Ingle. Underwater image compression using the wavelet transform. In *OCEANS '94. Proceedings*, volume 2, pages 533–537, Sep 1994.

[47] D.F. Hoag, V.K. Ingle, and R.J. Gaudette. Low-bit-rate coding of underwater video using wavelet-based compression algorithms. *Oceanic Engineering, IEEE Journal of*, 22(2):393 –400, apr 1997. ISSN 0364-9059. doi: 10.1109/48.585958.

[48] Geoffrey A. Hollinger, Sunav Choudhary, Parastoo Qarabaqi, Christopher Murphy, Urbashi Mitra, Gaurav S. Sukhatme, Milica Stojanovic, Hanumant

Singh, and Franz Hover. Communication protocols for underwater data collection using a robotic sensor network. In *International Workshop on Wireless Networking for Unmanned Autonomous Vehicles*, 2011. URL http://cres.usc.edu/cgi-bin/print_pub_details.pl?pubid=741.

[49] Geoffrey A. Hollinger, Urbashi Mitra, and Gaurav S. Sukhatme. Autonomous data collection from underwater sensor networks using acoustic communication. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, pages 3564–3570, San Francisco, CA, sep 2011.

[50] Albert S. Huang, Edwin Olson, and David Moore. LCM: Lightweight communications and marshalling. In *Int. Conf. on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan, Oct. 2010.

[51] I. Iglesias, A. Song, J. Garcia-Frias, M. Badiey, and G.R. Arce. Image transmission over the underwater acoustic channel via compressive sensing. In *Information Sciences and Systems (CISS), 2011 45th Annual Conference on*, pages 1 –6, march 2011. doi: 10.1109/CISS.2011.5766128.

[52] Michael Jakuba. A communications and control architecture for developing enhanced autonomy in deep sea AUVs. Provided by Author, 2010.

[53] Michael Jakuba, James Kinsey, Dana R. Yoerger, Richard Camilli, Chris Murphy, Daniel Steinberg, and Asher Bender. Exploration of the gulf of mexico oil spill with the sentry autonomous underwater vehicle. In *Proceedings of the International Conference on Intelligent Robots and Systems (IROS) Workshop on Robotics for Environmental Monitoring (WREM)*, San Francisco, 2011.

[54] E.P.C. Jones, L. Li, J.K. Schmidtke, and P.A.S. Ward. Practical routing in delay-tolerant networks. *Mobile Computing, IEEE Transactions on*, 6(8):943 –959, aug. 2007. ISSN 1536-1233. doi: 10.1109/TMC.2007.1016.

[55] Philo Juang, Hidekazu Oki, Yong Wang, Margaret Martonosi, Li-shiuan Peh, and Daniel Rubenstein. Energy-efficient computing for wildlife tracking : Design tradeoffs and early experiences with zebranet. *ASPLOSX Proceedings of the 10th international conference on Architectural support for programming languages and operating systems*, 37(10):96–107, 2002. URL http://portal.acm.org/citation.cfm?id=605408.

[56] J.W. Kaeli, H. Singh, and R.A. Armstrong. An automated morphological image processing based methodology for quantifying coral cover in deeper-reef zones. In *OCEANS 2006*, pages 1 –6, sept. 2006. doi: 10.1109/OCEANS.2006.306820.

[57] Phil (KA9Q) Karn. MACA - a new channel access method for packet radio. In *9th ARRL Computer Networking Conference*, London, Ontario, Canada, 1990.

123

[58] Jon Kristensen and Karstein Vestgard. Hugin-an untethered underwater vehicle for seabed surveying. In *OCEANS, 1998. Proceedings of MTS/IEEE*, volume 1, pages 118–123, September 1998.

[59] A. Kukulya, A. Plueddemann, T. Austin, R. Stokey, M. Purcell, B. Allen, R. Littlefield, L. Freitag, P. Koski, E. Gallimore, J. Kemp, K. Newhall, and J. Pietro. Under-ice operations with a remus-100 auv in the arctic. In *Autonomous Underwater Vehicles (AUV), 2010 IEEE/OES*, pages 1 –8, sept. 2010. doi: 10.1109/AUV.2010.5779661.

[60] C. Kunz, C. Murphy, H. Singh, C. Pontbriand, R. Sohn, S. Singh, T. Sato, C. Roman, K. Nakamura, M. Jakuba, R. Eustice, R. Camilli, and J. Bailey. Toward extraplanetary under-ice exploration: Robotic steps in the arctic. *Journal of Field Robotics*, 26(4), 2009.

[61] Qingzhong Li, Wenjin Wang, Bing Wang, and Xiaoling Geng. An efficient underwater color image coding method based on improved wavelet difference reduction. In *Audio, Language and Image Processing, 2008. ICALIP 2008. International Conference on*, pages 215 –219, july 2008. doi: 10.1109/ICALIP. 2008.4590000.

[62] Nicholas C Loomis. *Computational imaging and automated identification for aqueous environments*. PhD thesis, Woods Hole Oceanographic Institution and Massachusetts Institute of Technology, Cambridge, MA, Jun 2011. URL http://hdl.handle.net/1721.1/67589.

[63] D.E. Lucani, M. Medard, and M. Stojanovic. Random linear network coding for time-division duplexing: Field size considerations. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1 –6, 30 2009-dec. 4 2009. doi: 10.1109/GLOCOM.2009.5425257.

[64] D.E. Lucani, M. Stojanovic, and M. Medard. Random linear network coding for time division duplexing: When to stop talking and start listening. In *INFOCOM 2009, IEEE*, pages 1800 –1808, april 2009. doi: 10.1109/INFCOM. 2009.5062100.

[65] E.R.B. Marques, J. Pinto, S. Kragelund, P.S. Dias, L. Madureira, A. Sousa, M. Correia, H. Ferreira, R. Goncalves, R. Martins, D.P. Horner, A.J. Healey, G.M. Goncalves, and J.B. Sousa. AUV control and communication using underwater acoustic networks. In *OCEANS 2007 - Europe*, pages 1 –6, june 2007. doi: 10.1109/OCEANSE.2007.4302469.

[66] K. Martin, R. Lukac, and K.N. Plataniotis. SPIHT-based coding of the shape and texture of arbitrarily shaped visual objects. *Circuits and Systems for Video Technology, IEEE Transactions on*, 16(10):1196 –1208, oct. 2006. ISSN 1051-8215. doi: 10.1109/TCSVT.2006.882388.

[67] R. Martins, P.S. Dias, E.R.B. Marques, J. Pinto, J.B. Sousa, and F.L. Pereira. IMC: A communication protocol for networked vehicles and sensors. In *OCEANS 2009 - EUROPE*, pages 1 –6, may 2009. doi: 10.1109/OCEANSE. 2009.5278245.

[68] Sebastian H. Mernild, Glen E. Liston, Christopher A. Hiemstra, Konrad Steffen, Edward Hanna, and Jens H. Christensen. Greenland ice sheet surface mass-balance modelling and freshwater flux for 2007, and in a 1995âĂŞ2007 perspective. *Hydrological Processes*, 23(17):2470–2484, 2009. ISSN 1099-1085. doi: 10.1002/hyp.7354. URL http://dx.doi.org/10.1002/hyp.7354.

[69] A. E. Mohr, E. A. Riskin, and R. E. Ladner. Graceful degradation over packet erasure channels through forward error correction. In *Proc. Data Compression Conf DCC '99*, pages 92–101, 1999. doi: 10.1109/DCC.1999.755658.

[70] A. E. Mohr, E. A. Riskin, and R. E. Ladner. Unequal loss protection: graceful degradation of image quality over packet erasure channels through forward error correction. *IEEE Journal on Selected Areas in Communications*, 18(6):819–828, 2000. doi: 10.1109/49.848236.

[71] M. Molins and M. Stojanovic. Slotted fama: a mac protocol for underwater acoustic networks. In *OCEANS 2006 - Asia Pacific*, pages 1 –7, may 2006. doi: 10.1109/OCEANSAP.2006.4393832.

[72] Chris Murphy and Hanumant Singh. Wavelet compression with set partitioning for low bandwidth telemetry from auvs. In *Proceedings of the Fifth ACM International Workshop on UnderWater Networks (WUWNET) Conference*, 2010.

[73] Chris Murphy and Hanumant Singh. Fully embedded wavelet compression for low bandwidth image telemetry. In *MOOS Development and Applications Working Group*, volume 2, Cambridge, MA, July 2011.

[74] Christopher Alden Murphy. Lossy compression and real-time geovisualization for ultra-low bandwidth telemetry from untethered underwater vehicles. Master's thesis, MIT / WHOI Joint Program, September 2008.

[75] K. W. Nicholls, E. P. Abrahamsen, J. Buck, P. A. Dodd, C. Goldblatt, G. Griffiths, K. J. Heywood, N. E. Hughes, A. Kaletzky, G. F. Lane-Serff, S. D. McPhail, N. W. Millard, K. I. C. Oliver, J. Perrett, M. R. Price, C. J. Pudsey, K. Saw, K. Stansfield, M. J. Stott, P. Wadhams, A. T. Webb, and J. P. Wilkinson. Measurements beneath an antarctic ice shelf using an autonomous underwater vehicle. *Geophysical Research Letters*, 33, 2006. L08612, doi:10.1029/2006GL025998.

[76] Amrita A. Nimbalkar and Dario Pompili. Reliability in underwater inter-vehicle communications. In *Proceedings of the third ACM international workshop on Underwater Networks*, WuWNeT '08, pages 19–26, New York, NY,

125

USA, 2008. ACM. ISBN 978-1-60558-185-9. doi: http://doi.acm.org/10.1145/1410107.1410112. URL http://doi.acm.org/10.1145/1410107.1410112.

[77] A. Patil and M. Stojanovic. A node discovery protocol for ad hoc underwater acoustic networks. In *OCEANS, 2011 IEEE - Spain*, pages 1 –9, june 2011. doi: 10.1109/Oceans-Spain.2011.6003416.

[78] William A. Pearlman and Amir Said. Set partition coding: Part I of set partition coding and image wavelet coding systems. In *Foundations and Trends in Signal Processing*, pages 95–180. Now Publishers, Jun 2008.

[79] B. Peleato and M. Stojanovic. Distance aware collision avoidance protocol for ad-hoc underwater acoustic sensor networks. *Communications Letters, IEEE*, 11(12):1025 –1027, december 2007. ISSN 1089-7798. doi: 10.1109/LCOMM.2007.071160.

[80] Costas Pelekanakis, Milica Stojanovic, and Lee Freitag. High rate acoustic link for underwater video transmission. *OCEANS 2003. Proceedings*, 2:1091–1097, Sep 2003.

[81] C. Perkins, E. Belding-Royer, and S. Das. Ad hoc On-Demand Distance Vector (AODV) Routing. RFC 3561 (Experimental), July 2003. URL http://www.ietf.org/rfc/rfc3561.txt.

[82] O. Pizarro, P. Rigby, M. Johnson-Roberson, S.B. Williams, and J. Colquhoun. Towards image-based marine habitat classification. In *OCEANS 2008*, pages 1 –7, sept. 2008. doi: 10.1109/OCEANS.2008.5152075.

[83] O. Pizarro, S.B. Williams, and J. Colquhoun. Topic-based habitat classification using visual data. In *OCEANS 2009 - EUROPE*, pages 1 –8, may 2009. doi: 10.1109/OCEANSE.2009.5278260.

[84] D. Pompili, T. Melodia, and I.F. Akyildiz. Distributed routing algorithms for underwater acoustic sensor networks. *Wireless Communications, IEEE Transactions on*, 9(9):2934 –2944, september 2010. ISSN 1536-1276. doi: 10.1109/TWC.2010.070910.100145.

[85] John G. Proakis. *Digital Communications*. McGraw-Hill, 4 edition, 2000.

[86] A. Rajala, M. O'Rourke, and D.B. Edwards. AUVish: An application-based language for cooperating AUVs. In *OCEANS 2006*, pages 1 –6, sept. 2006. doi: 10.1109/OCEANS.2006.307128.

[87] K. Rajan, F. Py, C. McGann, J. Ryan, T. O'Reilly, T. Maughan, and B. Roman. Onboard adaptive control of auvs using automated planning and execution. In *International Symposium on Unmanned Untethered Submersible Technology (UUST)*, Durham, NH, August 2009.

[88] J. Rice and D. Green. Underwater acoustic communications and networks for the us navy's seaweb program. In *Sensor Technologies and Applications, 2008. SENSORCOMM '08. Second International Conference on*, pages 715 –722, aug. 2008. doi: 10.1109/SENSORCOMM.2008.137.

[89] J.A. Rice and Chee Wei Ong. A discovery process for initializing underwater acoustic networks. In *Sensor Technologies and Applications (SENSORCOMM), 2010 Fourth International Conference on*, pages 408 –415, july 2010. doi: 10. 1109/SENSORCOMM.2010.110.

[90] Joseph A Rice. Us navy seaweb development. In *MOBICOM 2007: WUWNet*, pages 3–4, Montréal, Quebéc, Canada, 2007.

[91] Joseph A. Rice, Robert K. Creber, Christopher L. Fletcher, Paul A. Baxley, Kenneth E. Rogers, and Donald C. Daviso. Seaweb underwater acoustic nets. In *SSC San Diego Biennial Review*. Space and Naval Warfare Systems Center, San Diego, 2001.

[92] Julio K Rosenblatt. *DAMN: A distributed architecture for mobile navigation*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, 1997.

[93] Julio K Rosenblatt, Stefan B Williams, and Hugh Durrant-Whyte. Behavior-based control for autonomous underwater exploration. *International Journal of Information Sciences*, 145(1-2):69–87, 2002.

[94] Subhasis Saha. Image Compression - from DCT to Wavelets : A Review. *ACM Crossroads: Data Compression*, Spring 2000. URL http://www.acm.org/crossroads/xrds6-3/sahaimgcoding.html.

[95] Amir Said and William A. Pearlman. A new, fast, and efficient image codec based on set partitioning in hierarchical trees. *Circuits and Systems for Video Technology, IEEE Transactions on*, 6(3):243–0250, Jun 1996. URL http://www.cipr.rpi.edu/research/SPIHT/.

[96] David Salomon. *Data Compression: The Complete Reference*, chapter 4.5. Springer, 4 edition, 2007.

[97] Craig Sayers, Angela Lai, and Richard Paul. Visual imagery for subsea teleprogramming. In *Proc. IEEE Robotics and Automation Conference*, 2005.

[98] Toby Schneider and Henrik Schmidt. pacommshandler: Acoustic networking in moos with the whoi micro-modem. In *MOOS Developers and Working Group Meeting*, Cambridge, MA, August 2010.

[99] Toby Schneider and Henrik Schmidt. The Dynamic Compact Control Language: A compact marshalling scheme for acoustic communications. In *Proceedings of MTS/IEEE OCEANS 2010*, Sydney, Australia, 2010.

[100] Toby Schneider and Henrik Schmidt. Unified command and control for heterogeneous marine sensing networks. *Journal of Field Robotics*, 27(6):876–889, 2010. ISSN 1556-4967. doi: 10.1002/rob.20346. URL http://dx.doi.org/10.1002/rob.20346.

[101] R. Scopigno and H.A. Cozzetti. Mobile slotted aloha for vanets. In *Vehicular Technology Conference Fall (VTC 2009-Fall), 2009 IEEE 70th*, pages 1 –5, sept. 2009. doi: 10.1109/VETECF.2009.5378792.

[102] Keith Scott and Scott Burleigh. Bundle Protocol Specification. RFC 5050 (Experimental), November 2007. URL http://www.ietf.org/rfc/rfc5050.txt.

[103] Jerome M. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *Signal Processing, IEEE Transactions on*, 41(12):3445–3462, Dec 1993.

[104] Hanumant Singh, Ali Can, Ryan Eustice, Steve Lerner, Neil McPhee, Oscar Pizarro, and Chris Roman. Seabed AUV offers new platform for high-resolution imaging. *EOS, Transactions of the AGU*, 85(31):289,294–295, August 2004.

[105] Sandipa Singh, Sarah E. Webster, Lee Freitag, Louis L. Whitcomb, Keenan Ball, John Bailey, and Craig Taylor. Acoustic communication performance of the whoi micro-modem in sea trials of the nereus vehicle to 11,000 m depth. In *OCEANS 2009, MTS/IEEE Biloxi - Marine Technology for Our Future: Global and Local Challenges*, pages 1 –6, oct. 2009.

[106] S.M. Smith, J.C. Park, and A. Neel. A peer-to-peer communication protocol for underwater acoustic communication. In *OCEANS '97. MTS/IEEE Conference Proceedings*, volume 1, pages 268 –272 vol.1, oct 1997. doi: 10.1109/OCEANS.1997.634374.

[107] Robert Sohn, Claire Willis, Susan Humphris, Timothy Shank, Hanumant Singh, Henrietta Edmonds, Clayton Kunz, Ulf Hedman, Elisabeth Helmke, Michael Jakuba, Bengt Liljebladh, Julia Linder, Christopher Murphy, Koichi Nakamura, Taichi Sato, Vera Schlindwein, Christian Stranne, Maria Tausenfreund, Lucia Upchurch, Peter Winsor, Martin Jakosson, and Adam Soule. Explosive volcanism on the ultraslow-spreading Gakkel ridge, Arctic Ocean. *Nature*, 453(7199):1236–1238, June 2008.

[108] Milica Stojanović. Recent advances in high speed underwater acoustic communications. *IEEE Journal of Oceanic Engineering*, 21(2):125–136, April 1996.

[109] Milica Stojanović. On the relationship between capacity and distance in an underwater acoustic communication channel. In *SIGMOBILE Mobile Computing and Communications Review*, volume 11, pages 34 – 43, October 2007.

[110] Milica Stojanović. Capacity of a relay acoustic channel. In *OCEANS 2007*, pages 1 –7, 29 2007-oct. 4 2007. doi: 10.1109/OCEANS.2007.4449214.

[111] Roger P. Stokey, Lee E. Freitag, and Matthew Grund. A Compact Control Language for auv acoustic communication. *Oceans 2005 - Europe*, 2:1133–1137, June 2005.

[112] Roger P. Stokey, Alexander Roup, Chris von Alt, Ben Allen, Ned Forrester Tom, Austin, Rob Goldsborough, Mike Purcell, Fred Jaffre, Greg Packard, and Amy Kukulya. Development of the remus 600 autonomous underwater vehicle. In *OCEANS, 2005. Proceedings of MTS/IEEE*, volume 2, pages 1301–1304, Washington, 2005.

[113] Fiammetta Straneo, Gordon S. Hamilton, David A. Sutherland, Leigh A. Stearns, Fraser Davidson, Mike O. Hammill, Garry B. Stenson, and Aqqalu Rosing-Asvid. Rapid circulation of warm subtropical waters in a major glacial fjord in east greenland. *Nature Geosci*, 3(3):182–186, March 2010. ISSN 1752-0894. URL http://dx.doi.org/10.1038/ngeo764.

[114] Kristen Stubbs, Pamela J. Hinds, and David Wettergreen. Autonomy and common ground in human-robot interaction: A field study. *Intelligent Systems, IEEE*, 22(2):42 –50, march-april 2007. ISSN 1541-1672. doi: 10.1109/MIS.2007.21.

[115] M. Suzuki, T. Sasaki, and T. Tsuchiya. Digital acoustic image transmission system for deep-sea research submersible. In *OCEANS '92. 'Mastering the Oceans Through Technology'. Proceedings.*, volume 2, pages 567 –570, oct 1992. doi: 10.1109/OCEANS.1992.607839.

[116] A. Syed, Wei Ye, and J. Heidemann. Comparison and evaluation of the t-lohi MAC for underwater acoustic sensor networks. *Selected Areas in Communications, IEEE Journal on*, 26(9):1731 –1743, december 2008. ISSN 0733-8716. doi: 10.1109/JSAC.2008.081212.

[117] J.L. Talavage, T.E. Thiel, and D. Brady. An efficient store-and-forward protocol for a shallow-water acoustic local area network. In *OCEANS '94. 'Oceans Engineering for Today's Technology and Tomorrow's Preservation.' Proceedings*, volume 1, pages I/883 –I/888 vol.1, sep 1994. doi: 10.1109/OCEANS.1994.363865.

[118] David R. Thompson, Trey Smith, and David Wettergreen. Information-optimal selective data return for autonomous rover traverse science and survey. In *Proc. IEEE Int. Conf. on Robotics and Automation*, 2008.

[119] William H. Thorp. Deep-ocean sound attenuation in the sub- and low-kilocycle-per-second region. *The Journal of the Acoustical Society of America*, 38(4):648–654, 1965. doi: 10.1121/1.1909768. URL http://link.aip.org/link/?JAS/38/648/1.

[120] Jun Tian and Raymond Wells. Embedded image coding using wavelet difference reduction. In Pankaj Topiwala, editor, *Wavelet Image and Video Compression*, volume 450 of *The Kluwer International Series in Engineering and Computer Science*, pages 289–301. Springer Netherlands, 2002. ISBN 978-0-306-47043-1.

[121] Nicholas Tolimieri, M. Elizabeth Clarke, Hanumant Singh, and Chris Goldfinger. Development of the seabed auv to monitor populations of west-coast groundfish in untrawlable habitat: an example with rosethorn rockfish, sebastes helvomaculatus. In Press.

[122] L. Toni, L. Rossi, N. Agoulmine, and J.-G. Fontaine. Virtual unequal error protection in underwater image transmission. In *OCEANS 2010 IEEE - Sydney*, pages 1 –6, may 2010. doi: 10.1109/OCEANSSYD.2010.5603526.

[123] Laura Dubreuil Vall, Daniel Sura, and Milica Stojanovic. Towards underwater video transmission. In *Proceedings of the Sixth ACM International Workshop on Underwater Networks*, WUWNet '11, pages 4:1–4:5, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-1151-9. doi: http://doi.acm.org/10.1145/2076569.2076573. URL http://doi.acm.org/10.1145/2076569.2076573.

[124] Martin Vetterli. Wavelets, approximation, and compression. *IEEE Signal Processing Magazine*, pages 59 – 73, September 2001.

[125] James S. Walker. Lossy image codec based on adaptively scanned wavelet difference reduction. *Optical Engineering*, 39(7):1891–1897, 2000. ISSN 00913286. doi: DOI:10.1117/1.602573. URL http://dx.doi.org/10.1117/1.602573.

[126] James S. Walker, Truong Q. Nguyen, and Ying-Jui Chen. A low-power low-memory system for wavelet-based image compression. In *Optical Engineering, Research Signpost*, 2003.

[127] Sarah E Webster, Ryan M Eustice, Chris Murphy, Hanumant Singh, and Louis L. Whitcomb. Toward a platform-independent acoustic communications and navigation system for underwater vehicles. In *Proc. IEEE/MTS OCEANS Conf. Exhib.*, pages 1–7, Biloxi, MS, October 2009.

[128] H. Wiemann, M. Meyer, R. Ludwig, and Chang Pae O. A novel multi-hop ARQ concept. In *Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st*, volume 5, pages 3097 – 3101 Vol. 5, may-1 june 2005. doi: 10.1109/VETECS.2005.1543917.

[129] Ian H. Witten, Radford M. Neal, and John G. Cleary. Arithmetic coding for data compression. *Commun. ACM*, 30:520–540, June 1987. ISSN 0001-0782. doi: http://doi.acm.org/10.1145/214762.214771. URL http://doi.acm.org/10.1145/214762.214771.

[130] G.G. Xie and J.H. Gibson. A network layer protocol for UANs to address propagation delay induced performance limitations. In *OCEANS, 2001. MT-S/IEEE Conference and Exhibition*, volume 4, pages 2087 –2094 vol.4, 2001. doi: 10.1109/OCEANS.2001.968320.

[131] Peng Xie, Jun-Hong Cui, and Li Lao. VBF: Vector-based forwarding protocol for underwater sensor networks. In *Proceedings of IFIP Networking 2006*, Coimbra, Portugal, may 2006.

[132] Dana R. Yoerger, Albert M. Bradley, Steve C. Martin, and Louis L. Whitcomb. The sentry autonomous underwater vehicle: Field trial results and future capabilities. *AGU Fall Meeting Abstracts*, 1:A1674, December 2006.

[133] Dana R. Yoerger, Michael V. Jakuba, Albert M. Bradley, and Brian Bingham. Techniques for deep sea near bottom survey using an autonomous vehicle. *International Journal of Robotics Research*, 26(1):41–54, 2007.

[134] Caidan Zhao and En Cheng. Error-tolerant image coding for underwater acoustic transmissions. In *Communications, Circuits and Systems Proceedings, 2006 International Conference on*, volume 1, pages 123 –125, june 2006. doi: 10.1109/ICCCAS.2006.284600.

[135] Yibo Zhu, Zhong Zhou, Zheng Peng, Michael Zuba, and Jun-Hong Cui. An efficient geo-routing aware mac protocol for underwater acoustic networks. *ICST Transactions on Mobile Communications and Applications*, 11(7–9):e6, 9 2011. doi: 10.4108/icst.trans.mca.2011.e6.

[136] F. Zorzi, M. Stojanovic, and M. Zorzi. On the effects of node density and duty cycle on energy efficiency in underwater networks. In *OCEANS 2010 IEEE - Sydney*, pages 1 –6, may 2010. doi: 10.1109/OCEANSSYD.2010.5603904.

[137] M. Zorzi, P. Casari, N. Baldo, and A. Harris. Energy-efficient routing schemes for underwater acoustic networks. *Selected Areas in Communications, IEEE Journal on*, 26(9):1754 –1766, december 2008. ISSN 0733-8716. doi: 10.1109/JSAC.2008.081214.