Three-dimensional Imaging Sonar Signal Processing System Based on Blade Server

Xiangrui Zeng, Jingyang Liu, Bo Gao
Science and Technology on Sonar Laboratory, Hangzhou Research Institute of Applied Acoustics, Hangzhou 310012, China
zengxiangrui688@139.com

Abstract. The three-dimensional high-resolution imaging sonar uses multi-beam technology combined with synthetic aperture technology to achieve centimeter-level imaging of small underwater targets. It has a good imaging effect on underwater anchoring, sinking mine and buried mine target detection and underwater topography, and has a wide range of military and civil engineering applications. The system has a large amount of data transmission and computation, including multiple functional modules such as synthetic aperture, high resolution array processing, image processing, and target recognition. The software architecture is complex and real-time system development faces serious challenges. The blade servers have been widely used in the telecommunications, financial and other big data processing fields for the high efficiency, stability and autonomy. This paper introduces a cluster 3D imaging signal processing system based on x240m5 blade server. The message queue communication (ZMQ) is used to realize the stable transmission of large data volume. At the same time, the mixed programming technology of C language and Matlab language combines the convenience of real-time and software integration, which greatly shortens the development cycle. The system runs stably and realizes the imaging recognition of the small spheres and their clump weights in the simulated target water.

1. Introduction
With the deepening of marine development, underwater small target detection related technologies have been attached more and more importance. We have developed a three-dimensional high-resolution imaging sonar by combining multiple beams with synthetic apertures. It enables the target exploration to extend from the traditional two-dimensional data to the three-dimensional volume date to obtain the target stereoscopic scale image, which is more convenient for target recognition. The technology can be widely used in civil fields such as offshore oil pipeline laying, error detection, topography and geomorphologic mapping, and ship-sunken salvage. It can be used for mine detection in the military, especially for the buried mines which are difficult to detect[1].

The three-dimensional imaging sonar includes multiple functional modules such as synthetic aperture, high-resolution array processing, image processing, and target recognition. The complexity of the software architecture leads to huge data transmission and computation, and real-time system development faces severe challenges. This paper introduces a blade server-based cluster signal processing system, which uses message queue[2] communication (ZMQ) to achieve stable transmission of large data volume. Moreover, the mixed programming technology of C language and Matlab language takes both timeliness and software integration convenience into account. The system runs stably and realizes imaging detection of hollow ball targets and their clump weights in water.
2. General overview of the system

The system is mainly composed of signal processor, underwater wet end, display console and integrated processor cabinet. The underwater wet end consists of an underwater launching bay, a receiving compartment, an electronic compartment and a power compartment. The transmitting cabin transmits a frequency-modulated signal; the receiving cabin receives the echo signal, and the electronic cabin transmits the received echo data to the signal processor. The underwater wet end is connected to the display console, the signal processor, the task computer, and the integrated processing cabinet through the gigabit network switch. The display console system is connected to the underwater wet end, the signal processor, and the integrated processing cabinet through a Gigabit network cable. It is used to send various data and commands on the wet end and the dry end, and is used for updating and displaying real-time images and commands. The integrated processing cabinet contains data recorders, task computers, GPS and other equipment. The system structure is shown in Figure 1:

![Figure 1 Overall composition of the system](image)

Due to the large amount of data required for three-dimensional images the requirements for network transmission capabilities and the memory for computing are quite demanding. The use of pure DSP signal processing boards has been unable to meet the requirements of ultra-large data processing. Only multi-processing parallelism can solve the problem of large-scale data computing. Its hardware platform is based on blade servers, and it can meet the needs as a multi-CPU parallel processing platform. The blade server has the advantages of small size, good reliability, large amount of data transmitted by the internal network bus, strong real-time computing capability, and suitable for real-time processing of high-speed broadband signals. Communication between the various blades within the system is performed using the message queue (ZMQ) protocol.

Three-dimensional imaging algorithm: Three-dimensional synthetic aperture sonar[3] (SAS) can solve the problem that the detection range and detection accuracy cannot be improved at the same time. The principle is to use a small aperture array to move linearly along the space to virtual large aperture array. By calculating the spatial position and phase relationship, the signal is coherently superimposed to obtain the equivalent large aperture. In particular, it has a good effect on the underwater target detection of isolated target detection and point target under diffuse scattering background.

The system consists of high-frequency side-scan synthetic aperture sonar and low-frequency three-dimensional imaging sonar[4]. The two sonars work at the same time, with no bottom sweeping blind spots. The high-frequency side-scan synthetic aperture sonar signal processing consists of a side sweep pulse pressure, a side scan synthetic aperture, a side scan image processing, and an image feed display module. The low-frequency three-dimensional imaging sonar signal processing part mainly consists of three-dimensional pulse pressure, three-dimensional synthetic aperture, three-dimensional beam forming, three-dimensional image processing, three-dimensional pulse pressure beam forming, three-dimensional pulse pressure image processing and image sending and displaying modules. The block diagram of the system signal processing algorithm is shown in Figure 2.
A server node is used to receive, organize and pack wet end data. Accumulated dates are collected respectively for three-dimensional pulse pressure and side-sweep pulse pressure. Due to the large amount of data, simply using C language to program is too complicated, and the use of Matlab to achieve its functions can not meet the performance requirements. The combination of C language and Matlab language programming technology takes into account the real-time and software integration convenience, greatly shortening the development cycle and the number of codes. The display console is a combination of a variety of complex images. It is particularly important to control the beat of the image transmission and display of each module when images are transmitting and displaying. The message queue is used to make a sufficient buffer between the beats, so that the data will not be blocked, enabling the correctness, completeness and fluency of the images.

3. X240m5 blade server performance

The blade server is capable of inserting multiple card-type server units in a standard-height rack-mount chassis, achieving high availability and high density. Each of the "blades" is actually a system motherboard. It can start its own operating system Windows, Linux, etc. through the "onboard" hard drive. Each motherboard runs its own system, serves different user groups and is not related to each other. Administrators can assemble these motherboards into a single server cluster. In group mode, all motherboards are connected to provide a high-speed network environment while sharing resources to serve the same user base. Inserting new "blades" into the cluster improves overall performance. Each "blade" is hot-swappable and the system can be easily replaced to reduce system maintenance time. The blade server reduces the number of external cables, greatly reducing the hidden dangers caused by cable connection failures and improving system reliability [5].

![Figure 2: Signal Processing System Flow](image)

![Figure 3: Lenovo HPC Knife Box and Blade Content](images)
performance and large memory capacity. The x240m5 also supports TruDDR4 memory and has been vigorously tested and verified to allow servers to support higher speeds than other baselines. The x240m5 has two 10GbE interfaces, four 1GbE adapters, two ports, four ports and eight port 10GbE adapters, a 2-port 40GbE adapter, a 2-port 4-port 8/16Gb Fibre Channel adapter, and a 2-port QDR/FDR inBand adapter, meeting the requirements of large data transmission.

The underwater wet end data is sent to the blade server through the Gigabit network, and the intermediate data is transmitted between the nodes through zero message queue (ZMQ) communication. Pulse compression, three-dimensional synthetic aperture processing, beamforming, and side-scan imaging are sent to the display console. The content of each node signal processing module is shown in Figure 3(b).

4. Message queue based data transmission 240m5 blade

4.1. Message queue overview
A message queue is a server located between an application server and a database. As a buffer, the message queue server receives the database operation commands sent by the application server, and then sends them to the database server for execution according to its own configuration. Because the message queue server is much faster than the database server, it can process and return data quickly. Message queue has good scalability. In the case of high transmission, delaying writing to the database can effectively alleviate the pressure on the database [6]. Message Queuing can ensure reliable delivery of messages under various network conditions, and can overcome the problems caused by network line differences and instability.

4.2. ZMQ transmission
Zero message queue (ZMQ) communication is considered to be the "fastest" message queue in history. It is a simple and easy-to-use transport layer that makes Socket programming simpler, cleaner, and more efficient. The protocol is faster than TCP and is suitable for large clusters. Ultra-high information throughput, self-developed and persistent are based on the C language development of real-time processing between the tasks of the communication. Zero message queue (ZMQ) is not a separate service or program. It is just a set of components [7].

It encapsulates network communication, message queue, thread scheduling and other functions, and provides a simple API to the upper layer. The application implements high-performance network communication by clamping API files and calling API functions. As shown in Figure 4, zero message queue (ZMQ) makes high-performance web applications extremely simple and user-friendly, allowing for flexible scaling between multiple threads, cores, and host boxes. Zero message queue (ZMQ) is similar to a series of interfaces of Socket. The difference between it and Socket is that ordinary Socket is end-to-end (1:1 relationship), and Zero message queue (ZMQ) can be N:M relationship. People know more about BSD sockets in point-to-point connections [8]. Point-to-point connections require explicit connection establishment, destruction of connections, and selection protocols such as TCP and UDP. Zero message queue (ZMQ) blocks these details and makes network programming easier.

![Figure 4 hierarchy](image_url)
connections. The connection will exhibit a certain message pattern, which is determined by the type of socket that created the connection. ZMQ does not provide a function like zmq_accept(), which automatically starts receiving connections when the socket is bound to the endpoint.

The difference in data transmission: Zero message queue (ZMQ) sockets transmit messages, TCP transmits bytes, and UDP transmits frames. ZMQ sockets perform I/O operations in the background. They are sent to a local buffer queue whether receiving or sending messages. The size of this memory queue is configurable. The TCP protocol can only perform peer-to-peer connections, while ZMQ can perform one-to-many, many-to-many, many-to-one operations. Zero message queue (ZMQ) is sent and received as a whole. It will not only receive a part of the message, and it will not be sent immediately but has a certain delay[9]. With the TCP protocol, when the receiver is processing too slowly, the message will accumulate at the publisher. Using the zero message queue (ZMQ) communication protocol, it is possible to discard received data in the future, preventing clogging and processing errors.

Zero message queue (ZMQ) communication has four models, one-to-one pairing model (EXC-PAIR), request answering model (REQ-REP), publish-subscription model (PUB-SUB), and push-pull model (PUSH-PULL). The three-dimensional signal processing signal processing system mainly uses the publish-subscribe model (PUB-SUB) mode. The sender directly distributes data in one direction, and does not care whether all data is sent to the destination address. If the receiving end does not receive it in time, the relevant data will be directly discarded, preventing the data transmission and image transmission process from being disordered.

![ZMQ Communication Diagram](image)

Figure 5 ZMQ communication has 4 models

Each computing node (blade) in the three-dimensional imaging sonar signal processing system activates two Gigabit Ethernet ports and two 10 Gigabit Ethernet ports. Two Gigabit Ethernet ports do link aggregation to implement link backup of the Gigabit network, and connect to the external Gigabit Ethernet switch EN2091, IP address segment 10.10.1.65~78. Two 10 Gigabit optical ports are connected to the link aggregation. External 10 Gigabit optical fiber switch EN4093R. The inter-blade communicates through 10 Gigabit optical port and uses distributed communication middleware ZMQ to complete data transmission in the form of distribution-subscription (PUB-SUB). The pulse compression calculation results are distributed to different blades for side-scan synthesis aperture and three-dimensional pulse pressure beamforming. In order to meet the real-time effect of the system, the three-dimensional synthetic apertures are separately distributed to three blade servers for calculation. The data transmission of each node is transmitted by zero message queue (ZMQ) communication, which satisfies the requirements of system not losing packets and real-time processing.

5. C language and Matlab language hybrid programming technology

The three-dimensional imaging signal processing system uses the server to process the underwater wet end data, and the algorithm is complicated and the amount of information to be processed is huge. As a professional numerical calculation software, Matlab software has powerful algorithm calculation. The C language is more flexible and convenient to use, and the calculation speed is faster. C language and Matlab language programming can be combined to greatly reduce the amount of code, quickly realize applications from simulation to engineering, and meet the real-time performance of the system.

The C language and Matlab program interface in Linux system is actually a dynamic link library named sig.so, which is generated by the Library Compiler in the Matlab software platform. When the C language needs it, it is referenced. The specific implementation of the C language and Matlab
language mixed development is described below. Taking one of them as an example, this function is an interface program when C language calls the Matlab language function in 3D signal processing, and it is the initial function that must be loaded at the beginning of the call.

```cpp
if(!pp_sw_porc Initialize())              //Function initialization
{
    std::cout <<"InitializeFail"<<std::endl;
    return -1
}
pp_sw_porc Terminate();                   //Need to terminate at the end

Called here is the 3D pulse pressure signal processing function. The input is input data in, copy function Cpy, and synthetic aperture number TD. The signal processing point number N, the pulse pressure length pulsetime, and the output result is placed in the out.

```cpp
void pp_sw_porc(short *in, mwArray *out,mwArray Cpy,short TD,unsigned int N,short pulsetime)
{  
    int i,j,data_len;
    double m_pp;
    data_len = N;
    m_pp = pulsetime;
    mwArray param1(TD,data_len, mxINT16_CLASS); //The parameter size is TD*data_len
    mwArray in1(1,1,mxDOUBLE_CLASS);                   //The parameter size is 1*1
    param1.SetData(in, TD*data_len);                //Put the data in in into param1
    in1.SetData(&m_pp,1);                //Put the data in m_pp into in1
    3d_PulsePress(1,*out,param1,Cpy,in1);       //Out is the output value
}
```

When C language calls Matlab program, it needs to convert data into mwArray data type. It is the bridge between C language and Matlab program. The data is placed in the mwArray data type param1 by the function SetData. The parameters are put into the mwArray data type in1 to complete the calculation of the Matlab program. The calculation result is put into the out output.

The C language calls the data in the out to perform subsequent operations, thereby implementing the Matlab program call.

6. Lake test exploration effect

This signal processing system is used to perform a detection experiment on a hollow ball target of a twenty-centimeter diameter with clump weights in the Qiandao Lake underwater acoustic test site. The current measured sound velocity is 1450 m/s and the test water depth is 50 m.

Figure 6 Test ball
The exploring test results are shown in Figure 7. Both the three-dimensional imaging and the side-scan imaging clearly show the suspended hollow balls and the clump weights in the water. The stratum is clearly visible and can achieve high resolution requirements. The system achieved the desired results.

7. Conclusion
This paper studies a three-dimensional imaging sonar signal processing system based on blade server design. The relatively simple hardware structure is used to realize the powerful data processing function of the three-dimensional sonar signal big data flow. The application of the blade server significantly reduces the failure rate of the hardware device compared with other signal processing devices, reduces the development complexity, and ensures the real-time requirements of the system. The lake test successfully detected the test hollow balls in the suspended water and the clump weights in the bottom, and the underwater geomorphological features were clearly visible. Fully achieving various performance indicators, the test verifies the feasibility of the algorithm. It has a good application and development prospects in military mine detection, marine mapping and civil salvage. In order to better verify the capabilities of the system, further sea trials are needed.

Acknowledgments
This work was financially supported by funding support for the CSIC innovative application research project.

References
[1] Hayes M P, Gough P T. Synthetic aperture sonar: a review of current status[J]. IEEE Journal of Oceanic Engineering, 2009, 34(3): 207-224.
[2] Sha Weiping, Rui Ting, Gao Qiyu, Zhang Sai, Zou Junhua Design of multi-thread programming based on ZeroMQ [J]. Jiangsu Science & Technology Information, 2017.
[3] Hong Yifan, Song Kunpo, Xia Shunren, Cong Weihua. 3D imaging sonar image post-processing and visualization[J]. Computer Engineering, 2011, 37(22): 196-197.
[4] Alan J. Hunter. Underwater Acoustic Modelling for Synthetic Aperture Sonar. University of Canterbury. Doctor’s thesis, June 2006: 20-80.
[5] Nicholas Harold Klausner. Underwater Target Detection Using Multiple Disparate Sonar Platforms [D]. Colorado State University 2010.
[6] Hayden J. Callow. Signal Processing for Synthetic Aperture Sonar Image Enhancement [D], University of Canterbury, 2003.
[7] Weiwei Jiang, Naikuo Chen, Shihua Geng. The Research of Heartbeat Detection Technique for Blade Server [J]. International Conference of Pioneering Computer Scientists. 2009.
[8] Caruso, Jeff. Neterion’s 10 Gigabit Ethernet adapter chosen for blade server [J]. Network World, 2009.
[9] Anonymous. Juniper Networks selects BLADEto develop Junos based blade server switches [J]. Telecomworldwire, 2009.