PUPIL: Programmable Ultrasound Platform and Interface Library

Robert Rohling, Wilson Fung, and Pedram Lajevardi

Department of Electrical and Computer Engineering
University of British Columbia, Vancouver, Canada
`rohling@ece.ubc.ca www.ece.ubc.ca/~{}rohling`

Abstract. A new programmable ultrasound machine and software interface is described. The software interface takes advantage of the open architecture of the new machine to provide real-time access to the digital image formation pipeline and control of the parameters of acquisition. The first application of the system seeks to enhance the visibility of a needle in an image-guided procedure. The enhancement algorithm detects the needle in an ultrasound image and automatically steers the ultrasound beam in the perpendicular direction. This direction produces the strongest echoes and raises the contrast of the needle in subsequent images. The results show improved visibility of a needle in both phantoms and real tissue. The results also demonstrate the flexibility and performance of the system and its suitability for a wide range of research.

1 Introduction

Ultrasound is a popular medical imaging modality because it offers the ability to visualize soft-tissue and blood flow in real-time with a low-cost, portable and safe system. There is a wide range of research on medical ultrasound including fundamental work on hardware and signal processing, as well as a growing number of applications in diagnostics, therapeutics and surgery [1]. Examples of new applications include computer-guided needle-biopsies, brachytherapy, and neurosurgery [2]. In most of these applications a commercial ultrasound machine is used and ultrasound images are acquired from a digital video frame-grabber connected to the analog video output of the machine [3]. Direct access to real-time digital images is normally not available. There are several disadvantages of video acquisition over direct digital access. The first is that there is a mismatch between the frame rate of the analog video standard (approximately 30 images/sec for NTSC and PAL) and the actual frame rate of the ultrasound machine (typically 1–100 images per second). Another disadvantage is the loss in quality to the digital content as it passes through the analog conversion. The video format was developed in the 1950's with resolution and image-quality tradeoffs determined by the needs of color television. Moreover, the analog format requires multiple pieces of data to be fused together such as B-mode images, Doppler signals and patient information. Complete separation of these data streams is

impossible with frame-grabbed images. DICOM (Digital Imaging and Communication In Medicine) is a digital transfer standard for hospital networks, but does not offer a solution to these drawbacks. In the ultrasound DICOM format, data fusion cannot be undone, and DICOM is neither intended nor capable of real-time image transfer.

A need exists for an ultrasound system with an open architecture for direct digital access to the various ultrasound data streams. There is also a need to provide researchers with the ability to reprogram and reconfigure the ultrasound system to implement new imaging techniques. Several research groups have tackled various aspects of these needs. For example, new multi-media technologies - such as programmable DSP chips - have been successfully adapted to the needs of ultrasound [4]. A notebook computer has also been used as the back-end of a miniature beamforming system to improve portability [5]. Other groups have custom-built systems for fundamental research on original, unprocessed RF (radiofrequency) echo data [6,7,8].

This paper describes a PC-based ultrasound system and software library intended for a wide range of research, including many aspects that compliment the research programs just described. The new system is called PUPIL: Programmable Ultrasound Platform and Interface Library. PUPIL is based on the Ultrasonix 500 - a new ultrasound machine developed by Ultrasonix Medical Corporation (Burnaby, BC, Canada). The Ultrasonix machine is different from most conventional ultrasound machines because it uses a standard personal computer (PC) to perform many of the image processing tasks instead of the usual set of customized rack-mounted printed circuit boards. The new approach is possible because the speed of PC's has recently achieved the ability to perform complex image formation calculations in real-time. The use of PC hardware also offers advantages in terms of reliability, cost and ability to incorporate technologies developed for the non-medical market. A version of the machine (500RP) has a research package that provides easy access to almost the entire image formation pipeline. The pipeline starts with original RF echo signals and ends with a fully-formed digital image. Access points include real-time RF data, pre and post-beamformed data, I/Q data and envelope data. Moreover, the research package offers the ability to control the parameters of image formation from software. This essentially "closes-the-loop" for an ultrasound machine incorporated into a larger research application. This paper describes the architecture and operation of PUPIL and relates it to the common needs of ultrasound researchers. A new application has also been developed to demonstrate the performance of PUPIL and the ability to implement feedback to the ultrasound machine.

2   PUPIL Hardware

The various hardware components and sub-components of PUPIL are shown in Figure 1. The main component is the Ultrasonix 500RP. It consists of a standard ultrasound probe connected to an acquisition module for beam forming and echo sampling. The acquisition module is based on a set of FPGA chips so the beam
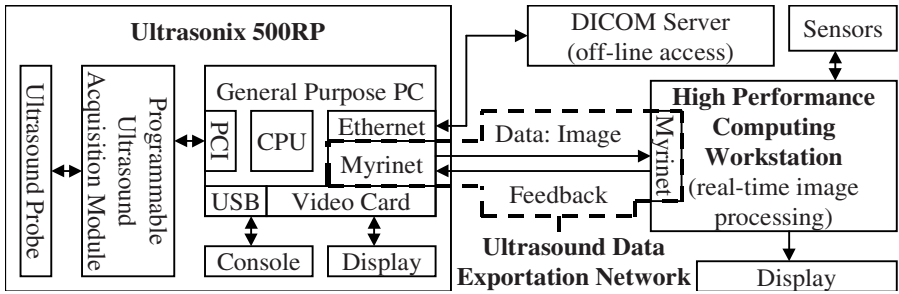
Fig. 1. Hardware components of PUPIL

forming and sampling algorithms can be easily changed. The board is connected through the PCI bus to a general purpose PC with standard USB, Ethernet and video ports. A sonographer operates the ultrasound machine through an external console in the same way as a conventional ultrasound machine. The CPU performs scan conversion, post-processing and displays the images in a Windows-based user interface. In most clinical settings, off-line picture archiving and storage (PACS) is implemented by sending the images to an external server using the DICOM standard.

In our research setting, an ultrasound data exportation network (UDEN) sends ultrasound data to an external high-performance computing workstation (HPCW) in real-time. External devices such as position and force/torque sensors can also be connected and sampled in synchronization with image capture. The HPCW is also capable of real-time display of the ultrasound data stream.

The UDEN uses a pair of Myrinet fibre-optic network communication cards (Myricom Inc., Arcadia, CA). The Myrinet system offers a high-bandwidth and low latency transfer of data to the HPCW. The high bandwidth is needed to accommodate the real-time transfer of ultrasound at native frame rates. The low latency benefits the ability to synchronize ultrasound images with external sensors as well as to feed data back to the ultrasound machine. An alternative is Gigabit Ethernet, but Myrinet offers almost twice the bandwidth and one eight of the latency, so was chosen for PUPIL.

3   PUPIL Software

The 500RP research package is a tool for researchers to customize the user interface, set the parameters of image formation, and capture data. For example, a user can adjust one of the amplifier gains, then capture a pre-scan converted digital image and export it over a LAN. All of this can be done during continuous operation of the ultrasound machine. A total of 966 parameters controlling the image formation and user interface are available. A full listing of the acquisition, control and user interface parameters can be found in Appendix A of an internal technical report [9].

For PUPIL, Ultrasonix also provided a software development kit for the research package (RP-SDK). The RP-SDK enables researchers to perform the same data access and control as the research package but with function calls in C++. In other words, the RP-SDK has control over all the parameters of the console and the RP software. This is done using the Microsoft Windows Messaging System. In addition, the RP-SDK allows direct memory access to the image buffers for the fastest possible transfer. With the RP-SDK, real-time applications can be developed on top of the existing software.

Some real-time applications require only a small level of computational overhead. For these cases, the application can be run directly on the ultrasound machine without the need for the UDEN or the HPCW. For example, a simple image-processing filter can be implemented by running a stand-alone program that acquires an image, filters the data, then displays the filtered image directly on the ultrasound monitor. The program can be run while the ultrasound machine is operating normally because there is spare computing capacity in the 500RP. But for more complex applications, the ultrasound data needs to be exported to external computing hardware. PUPIL provides a layer above the RP-SDK to perform the data export, as well as issue command inputs. The high level software layer of PUPIL allows programs developed to run on the 500RP to be ported and run on the HPCW without modification.

A diagram of the architecture of PUPIL is shown in Figure 2. It is designed to maximize data transfer rate and minimize latency. The **Communication Layers** represent the operation of the Myrinet cards and the **Data Buffers** represent the ultrasound data and input commands in memory. All other boxes represent separate threads to perform the data transfer. The main threads on both the 500RP and the HPCW (**Image Acquirer** and **Manager** respectively) manage internal resources and never deal directly with the data buffers. The **Image Acquirer** works with the RP-SDK to pass the data created by the **Ultrasound Device** to the **Communication Layer**. Inside the HPCW, the **Communication Layer** receives the image and places it in the **Data Buffer**. The **Workers** process the image, the **Viewer** displays the image, and the **Manager** synchronizes the whole process. The **Data Buffer(s)** is written as plural because the API of PUPIL allows developers to freely arrange the pipeline for dual or even triple buffering to further reduce time delays in retrieving the series of images. The bidirectional control arrows include both the verification signals for data transfer and the feedback commands.

With this pipeline configuration, a 552x448 post-scan converted image has a measured transfer time of 8.3msec. If the ultrasound machine produces 20 images per second, there is 41.7msec (50msec - 8.3msec) available for the HPCW to perform real-time image processing and display. This is sufficient time for image processing algorithms with moderate complexity to be implemented. For example, with the HPCW described below, a 17x17 median filter can be implemented in real-time using the Intel Integrated Performance Primitives compiler. Further optimization of the PUPIL software and results of tests with different hardware are described in an internal technical report [9].

In the current implementation, the HPCW has dual Intel Pentium Xeon processors operating at 2.8GHz, and the Myrinet cards operate at 133MHz. With
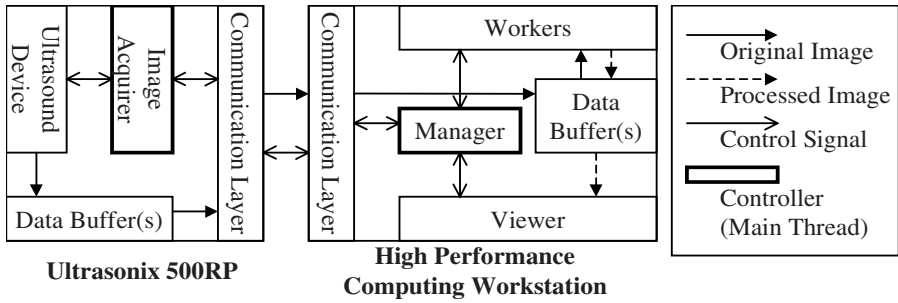
Fig. 2. Data flow in PUPIL

these components, the speed of data transfer is limited by either the Myrinet (240MB/sec) or the 33MHz-64bit PCI bus on the 500RP (264MB/sec). The latency of the network is 9μsec. The hardware components of the HPCW and the UDEN can be easily replaced as faster versions become available. Moreover, some researchers have already investigated the use of clusters for real-time ultrasound signal processing [10]. The architecture of PUPIL is designed so that it can be extended to a cluster configuration with the addition of a Myrinet switch. The optimization of high performance computing clusters is an ongoing research area and new advances can also be applied here.

In summary, the research package and SDK were developed by Ultrasonix with suggestions provided by UBC to allow an ideal interface with PUPIL. The PUPIL software is designed to allow a developer to quickly and easily create new applications using ultrasound images without needing to understand the internals of data acquisition and control. The following application was developed and tested in six weeks.

4   Needle Visualization Application

4.1   Introduction

The first application of PUPIL is on image guidance of a percutaneous needle insertion. The goal is to advance the tip of a needle to a target while observing the needle and target on the ultrasound display. This technique is used for procedures such as biopsies, drug delivery and hyperthermic therapy [11]. One of the common difficulties is visualization of the needle in the image, especially when using thin needles at deep locations in tissue. The reason is that the needle acts as a perfect reflector of ultrasound, so angling the needle away from the probe results in undetected echoes, as shown in Figure 3. In cases where the probe can be angled to nearly perpendicular to the needle, the visibility improves dramatically [12]. But achieving the ideal angle is usually limited to very shallow needle insertions, and is complicated by the need to hold the correct angle steady. Other solutions include coating or roughening the needle to improve detection. More often, the user estimates the needle location by observing both the visible portions of the needle and the motion of nearby tissue during insertion.
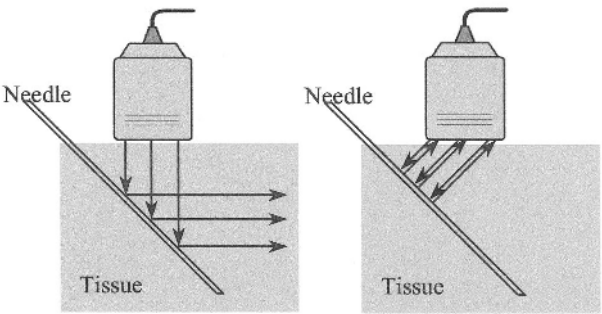
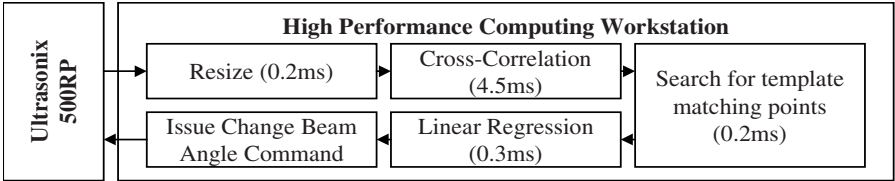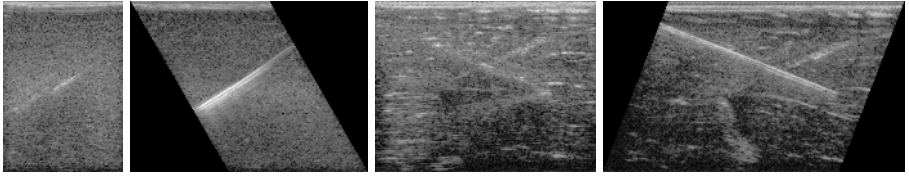**Fig. 3.** Beam steering and needle visibility

Fig. 4. Beam steering algorithm

We propose to achieve better refectivity of the needle by actively steering the beam of a linear array to a direction perpendicular to the needle. To achieve this goal in a fully automated manner, an algorithm must detect the needle in the image, calculate its angle, and steer the beam. Four properties of PUPIL are required for implementation: acquisition of a digital post-scan converted image; real-time image processing; feedback control of the ultrasound machine, and development of an integrated user interface for testing.

4.2   Implementation

The algorithm for beam steering is shown in Figure 4. Recognition of the needle within the image is achieved by using a template of a single line and the correlation coefficient is used as the metric. Note that the algorithm makes no attempt to locate the tip of the needle, only the angle. It also discards features that are not linear, and performs a regression on the correlation results. This makes the recognition problem easier and a robust implementation is readily found.

The main challenge is initialization. The first image has no beam steering so the visibility is poorest and the correlation coefficients are low. Yet the algorithm converges quickly to the correct angle as long as small portion of the needle is visible. For extreme cases where visibility is very poor, the needle is purposefully jiggled, and the correlation is performed on a series of *difference* images that highlight the region nearby the moving needle. Switching between these options is done automatically.

Fig. 5. Needle visibility in agar phantom (left images) and turkey breast (right images)
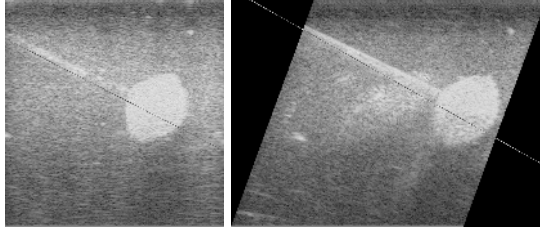
Fig. 6. Needle detection in a breast biopsy phantom with artificial lesion

4.3   Results

The principle of improving needle visibility through beam steering is demonstrated in Figure 5. The figures created from active beam steering are charactized by their trapezoidal shape. The visiblity of the needle is enhanced in both the phantom and real tissue. The automatic detection of the needle within the image is shown in Figure 6. The breast phantom contains large, high contrast lesions that obscure part of the needle. Yet the needle detection algorithm is successful because a portion of the needle is still visible.

Figure 4 also shows the timing for each step of the algorithm operating on the HPCW. The total time is 5.2msec for the algorithm. The time for correlation on difference images (when visibility is very poor) is essentially the same. The transmission time of the image from the 500RP is 8.3msec so the time for a complete cycle is 13.5msec (transmission of the return command is neglibile) [1]. This suggests that considerably more complicated algorithms can run on the current hardware without limiting the output rate of the 500RP.

5   Conclusions

PUPIL is designed as a flexible tool for research into ultrasound image processing and applications. The goal of improving the visibility of a needle in an image-guided procedure has been shown to be effective through active beam steering. The implementation of this application has demonstrated several abilities of

---

[1] The 500RP must wait at least until the the current image is completed before responding to a command, so the response time is not included here.

PUPIL: acquisition of digital data, real-time image processing, and control of the ultrasound machine from software.

Future research with PUPIL will be undertaken in several directions. First, the needle visibility application must be tested in a clinical setting. New image processing algorithms, such as real-time registration and spatial compounding, will soon be implemented. PUPIL will also help incorporate the 500RP into larger image-guided applications such as brachytherapy and neurosurgery. Finally, the ability to control the ultrasound machine through software also opens up new areas of research in computer assisted interventions.

References

1. Nelson, T.R.: Ultrasound into the Future. Journal of Ultrasound in Medicine . Vol. 20(12) (2001) 1263–1264
2. Fenster, A., Surry, K., Smith, W., Gill, J., Downey, D.B.: 3D Ultrasound Imaging: Applications in Image-Guided Therapy and Biopsy. Computers and Graphics. Vol. 26(4) (2002) 557–568.
3. Welch, J.N., Johnson, J.A., Bax, M.R., Badr, R., Shahidi, R.: A Real-Time Freehand 3D Ultrasound System for Image-Guided Surgery. In: Proc. of the IEEE Ultrasonics Symposium. (2000) 1601–1604
4. Kim, Y., Kim, J., Basoglu, C., Winter, T.C.: Programmable Ultrasound Imaging using Multimedia Technologies: A Next-Generation Ultrasound Machine. IEEE Trans. on Information Technology in Biomedicine, Vol. 1 (1997) 19–29
5. Chiang, A.M., Chang, P.P., Broadstone, S.R.: PC-Based Ultrasound Imaging System in a Probe. In: Proc. of IEEE Ultrasonics Symposium, Vol. 2 (2000) 1255–1260
6. Pesavento, E., Lorentz, A., Ermert, H.: System for Real-Time Elastography. Electronics Letters. Vol. 35(11) (1999) 941–942
7. Jensen, J.L., Jensen, J.A., Stetson, P.F., Antonius, P.: Multi-processor System for Real-Time Deconvolution and Flow Estimation in Medical Ultrasound. In: Proc. of IEEE Ultrasonics Symposium (1996) 1197–1200
8. Scabia, M., Biagi, E., Masotti, L.: Software and Hardware Platform for Real-Time Processing and Visualization of Echographic Radiofrequency Signals. IEEE Trans. on Ultrasonics, Ferroelectrics, and Frequency Control. Vol. 49(10) (2002) 1444–1452
9. Fung, W., Rohling, R.: Design of an Open-Architecture Ultrasound Acquisition System for Real-Time Processing and Control. Technical Report TR002, University of British Columbia, (2003) http://www.ece.ubc.ca/~rohling/publications/tr002.pdf
10. Bilas, A., Zhang, F., Dhanantwari, A., Plataniotis, K.N., Abiprojo, R., Stergiopoulos, S.: Parallelization and Performance of 3D Ultrasound Imaging Beamforming Algorithms on Modern Clusters. Intl. Conf. on Supercomputing (2002) 294–304
11. Holm, H.H., Skjolkbye, B.: Interventional Ultrasound. Ultrasound in Medicine and Biology. Vol. 22(7) (1996) 773–789
12. Fornage, Bruno D.: Sonographically Guided Needle Biopsy of Nonpalpable Breast Lesions. Journal of Clinical Ultrasound, Vol. 27 (1999) 385–398