

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

A61B 8/00 (2006.01)

G09G 5/373 (2006.01)

G06F 19/00 (2006.01)



[12] 发明专利申请公布说明书

[21] 申请号 200710076823.2

[43] 公开日 2009 年 3 月 4 日

[11] 公开号 CN 101375800A

[22] 申请日 2007.8.31

[21] 申请号 200710076823.2

[71] 申请人 深圳市蓝韵实业有限公司

地址 518034 广东省深圳市福田区景田北路
81 号碧景园 E 栋 601

[72] 发明人 程延俊 陈闽峰

[74] 专利代理机构 北京必浩得专利代理事务所

代理人 关松寿

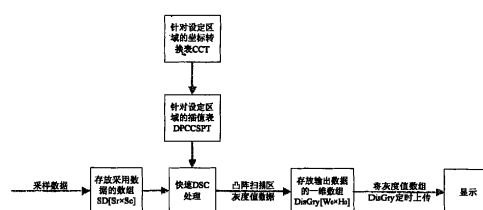
权利要求书 2 页 说明书 15 页 附图 3 页

[54] 发明名称

一种超声图像放大显示的方法

[57] 摘要

本发明公开了一种超声图像放大显示的方法，包括步骤：A1. 设定图像的放大区域，将所述放大区域的位置坐标保存；A2. 根据所述位置坐标建立插值计算表；A3. 根据所述插值计算表和采样数据计算所述放大区域内每个像素的灰度值数据；A4. 将所述灰度值数据输出到显示器进行显示。本发明使用软件进行超声图像放大显示，省去了 DSC 的硬件处理模块，节约了硬件成本，缩小了硬件的体积，因为本发明直接对采样数据进行放大显示，因此放大的图像不会失真。



1、一种超声图像放大显示的方法，其特征在于：包括步骤：

A1、设定图像的放大区域，将所述放大区域的位置坐标保存；

A2、根据所述位置坐标建立插值计算表；

A3、根据所述插值计算表和采样数据计算所述放大区域内每个象素的灰度值数据；

A4、将所述灰度值数据输出到显示器进行显示。

2、根据权利要求1所述的超声图像放大显示的方法，其特征在于：

所述步骤A2包括步骤：预先计算所述放大区域内每个象素的极坐标值，并将所述极坐标值存入坐标转换表中，根据所述坐标转换表建立所述插值计算表。

3、根据权利要求2所述的超声图像放大显示的方法，其特征在于：

所述插值计算表设为一数据结构的赋值集合，所述数据结构包括显示在屏幕上的点的一维坐标变量、矫正插值系数变量、采样点的位置变量。

4、根据权利要求3所述的超声图像放大显示的方法，其特征在于：

在使用所述数据结构前，对其进行初始化，将所有变量赋零值。

5、根据权利要求4所述的超声图像放大显示的方法，其特征在于：

所述坐标转换表设为二维数组。

6、根据权利要求5所述的超声图像放大显示的方法，其特征在于：

所述步骤A3包括步骤：所述灰度值数据首先存入一维数组中。

7、根据权利要求6所述的超声图像放大显示的方法，其特征在于：

设所述数据结构中的矫正插值系数为 CC_1 、 CC_2 、 CC_3 、 CC_4 ，则其根据如下数学式计算：

$$CC_1 = (1 - \alpha) \times (1 - \beta) \times 256 ,$$

$$CC_2 = (1 - \alpha) \times \beta \times 256 ,$$

$$CC_3 = \alpha \times (1 - \beta) \times 256 ,$$

$$CC_4 = \alpha \times \beta \times 256 ,$$

其中：\$(R_{qf}, \theta_{qf})\$ 是所述坐标转换表内的极坐标值，\$R_{qf}\$ 的整数部分为 \$i\$，
\$\theta_{qf}\$ 的整数部分为 \$j\$， \$\alpha = R_{qf} - i\$， \$\beta = \theta_{qf} - j\$。

一种超声图像放大显示的方法

技术领域

本发明涉及超声诊断设备技术领域，具体涉及一种超声图像放大显示的方法。

背景技术

现有技术的数字扫描转换器(DSC)系统由以下部分组成：(1)扫描器 I/O 接口电路；(2)多普勒处理器单元；(3)彩色血流处理器电路对从扫描器 I/O 电路接收到的数字化血流信号进行检测，然后把血流信号和二维信号叠加在一起；(4)M 方式、生理频谱电路把 M 方式生理信号、多普勒等数字信号，根据需要进行处理后，送入视频 I/O 接口板；(5)坐标变换电路利用特定的算法将回波数据的极坐标信号变换成输出的直角坐标信号；(6)图像存储器单元用于存储图像信号，供视频输入输出电路；(7)图像处理单元；(8)视频输入/输出电路将各种图像、字符、信号合成输出到彩色监视器供操作者观察，同时向视频打印机和录像机提供视频信号。

扫描变换为硬件实现，需要由两个帧存储器存储声束扫描格式的图象数据，两个帧存储器采用“乒乓”工作方式，当一个帧存储器按声束扫描时序写入当前的回波数据时，另一个帧存储器则以显示时钟速率读出数据，读出的数据经过二维线性插补运算后得到显示象素之值。每当声束扫描完成一个周期时，便进行两个帧存储器的“乒乓”切换。按照显示

时钟的频率，每隔一定时间输出一个显示象素，而每计算一个显示象素要求从帧存储器中读出四个声束扫描数据。如果这四个数据串行读出的话，要求帧存储器的工作频率高达50MHz以上。为了避免对高速存储器件的要求，帧存储器采用一种特殊的结构。每个帧存储器被分成A1、A2、B1、B2四个子存储器，分别存放不同线上的不同位置的采样点。根据扫描变换器二维线性插补的原理，计算一个显示象素所需的四个声束扫描数据必分散在A1、A2、B1、B2之中，因此可以从帧存储器的四个子存储器中并行读出。

现有技术的数字扫描转换器由硬件实现，首先硬件电路庞杂，PCB板布线复杂，任务量很多，其次硬件电路使用的FPGA（现场可编程单元）处理芯片和存储器都加大了硬件的成本，而且由于DSC使用硬件进行处理，上层应用程序得不到采样数据，对图像的后处理来说，增加了计算的复杂度。

现有技术在对超声图像进行放大显示时，一种处理方法是是对选定的放大区域，利用硬件数字扫描转换器重新进行插值计算，再显示经插值计算的数据，这样就不能避免硬件数字扫描转换器的缺点；一种是利用软件方法对图像数据重新插值计算再显示经插值计算的数据，这种处理方法造成放大的图像失真严重。

发明内容

本发明要解决的技术问题是提供一种超声图像放大显示的方法，克服现有技术使用硬件数字扫描转换器经过重新插值计算对超声图像进行放大显示时，硬件电路庞杂，PCB板布线复杂以及上层应用程序得不到采样数据，图像后处理计算复杂的缺陷；以及利用软件方法对图像数据重新插值计算对超声图像进行放大显示时，放大图像失真严重的缺陷。

本发明为解决上述技术问题所采用的技术方案为：

一种超声图像放大显示的方法，包括步骤：

A1、设定图像的放大区域，将所述放大区域的位置坐标保存；

A2、根据所述位置坐标建立插值计算表；

A3、根据所述插值计算表和采样数据计算所述放大区域内每个像素的灰度值数据；

A4、将所述灰度值数据输出到显示器进行显示。

所述的超声图像放大显示的方法，所述步骤 A2 包括步骤：预先计算所述放大区域内每个像素的极坐标值，并将所述极坐标值存入坐标转换表中，根据所述坐标转换表建立所述插值计算表。

所述的超声图像放大显示的方法，其中所述插值计算表设为一数据结构的赋值集合，所述数据结构包括显示在屏幕上的点的一维坐标变量、矫正插值系数变量、采样点的位置变量。

所述的超声图像放大显示的方法，其中在使用所述数据结构前，对其进行初始化，将所有变量赋零值。

所述的超声图像放大显示的方法：其中所述坐标转换表设为二维数组。

所述的超声图像放大显示的方法，所述步骤 A3 包括步骤：所述灰度值数据首先存入一维数组中。

所述的超声图像放大显示的方法，

设所述数据结构中的矫正插值系数为 CC_1 、 CC_2 、 CC_3 、 CC_4 ，则其根据如下数学式计算：

$$CC_1 = (1 - \alpha) \times (1 - \beta) \times 256,$$

$$CC_2 = (1 - \alpha) \times \beta \times 256,$$

$$CC_3 = \alpha \times (1 - \beta) \times 256,$$

$$CC_4 = \alpha \times \beta \times 256,$$

其中： (R_{qf}, θ_{qf}) 是所述坐标转换表内的极坐标值， R_{qf} 的整数部分为 i ， θ_{qf} 的整数部分为 j ， $\alpha = R_{qf} - i$ ， $\beta = \theta_{qf} - j$ 。

本发明的有益效果为：本发明使用软件进行超声图像放大显示，省去了 DSC 的硬件处理模块，节约了硬件成本，缩小了硬件的体积，因为本发明直接对采样数据进行放大显示，因此放大的图像不会失真。

附图说明

本发明包括如下附图：

图 1 为本发明设定放大区域的线框示意图；

图 2 为本发明通过改变线框大小调节放大区域的示意图；

图 3 为本发明显示数据的象素坐标示意图；

图 4 为本发明采样数据存储格式示意图；

图 5 为本发明对图像进行放大显示的流程示意图；

图 6 为本发明经过放大的图像示意图。

具体实施方式

下面根据附图和实施例对本发明作进一步详细说明：

1、设定放大区域

如图 1 和图 2 所示，在正常显示 B 超采集图像的情况下，判断是否有数字键“4”按下，如果有，则响应 VC 中的 OnKeyDown 消息和 OnMouseMove 消息，并调用绘制由黄色两条弧线和两条黄色直线段构成的扇面边框，在 OnPaint 中进行显示。如图 4 所示，该扇面边框的初始大小设定为采样数据大小的一半，即初始边框选定的区域为采样数据

的第 $\frac{1}{4}Sr$ 行到 $\frac{3}{4}Sr$ 行, 第 $\frac{1}{4}Sc$ 列到 $\frac{3}{4}Sc$ 列。

移动鼠标, 可以改变放大框的位置, 由于第一次显示扇面边框记录的状态为改变扇面边框位置的状态, 因此在此时移动鼠标, 响应 OnMouseMove 消息时, 改变扇面边框所选区域对应采样数据的行数和列数, 但是对应采样数据的行宽和列宽仍然保持不变, 也就是说对应的终止行与起始行的差、对应的终止列与起始列的差保持不变。

比如在没有移动鼠标时对应的采样数据的起始行是第 128 行, 终止行是第 384 行, 起始列是第 128 列, 终止列是第 384 列。那么当鼠标左右移动时, 将改变对应的起始列和终止列; 当鼠标上下移动时, 将改变对应的起始行和终止行。具体说明如下:

当鼠标向左移动一个像素时, 对应的起始列变为第 127 列, 终止列变为第 383 列; 反之, 如果是向右移动一个像素, 对应起始列则变为第 129 列, 终止列变为第 385 列。此时起始行和终止行保持不变。

当鼠标向上移动一个像素时, 对应的起始行变为第 127 行, 终止行变为第 383 行; 反之, 如果是向下移动一个像素, 对应起始行则变为第 129 行, 终止行变为第 385 行。此时起始列和终止列保持不变。

此时, 放大框变为虚线, 移动鼠标就可以改变放大框的大小。鼠标左右移动改变放大框的宽度, 也即是对应的需要放大的采样数据的列数发生改变; 鼠标上下移动改变放大框的高度, 也即是对应的需要放大的采样数据的行数发生改变。当按下鼠标右键, 将响应 VC 的 OnMouseUp 消息, 将改变扇面边框的设定状态为设定扇面边框的大小状态。此时仍然响应 VC 的 OnMouseMove 消息, 但是由于状态值发生了变化, 所以此时移动鼠标将改变扇面边框的大小, 而不改变其中心位置。下面举例来说, 假设扇面边框对应的采样数据的起始行是第 128 行, 终止行是第 384 行, 起始列是第 128 列, 终止列是第 384 列。那么当鼠标左右移动

时，将改变列宽的大小；当鼠标上下移动时，将改变对应的行宽大小。当鼠标向左移动一个像素时，对应的起始列变为第 129 列，终止列变为第 383 列，行宽变小；反之，如果是向右移动一个像素，对应起始列则变为第 127 列，终止列变为第 385 列，行宽变大。此时起始行和终止行保持不变，即列宽保持不变。

当鼠标向上移动一个像素时，对应的起始行变为第 127 行，终止行变为第 385 列，列宽变大；反之，如果是向下移动一个像素，对应起始行则变为第 129 行，终止行变为第 383 列，列宽变小。此时起始列和终止列保持不变，即行宽保持不变。

在完成放大框的大小和位置设定后，再次按下数字键“4”或者按下鼠标左键，OnKeyDown 将此时扇面边框对应的采样数据的起始行 (top)、终止行(bottom)和起始列(left)、终止列(right)的位置数据传送给 4 个变量保存，继而进行放大处理。完成整个放大显示的操作。

2、DSC 预处理

采样数据的格式和大小按照图 4 所示进行存储，共有 S_r 行， S_c 列，大小为 $S_r \times S_c$ Byte。每一行上的采样数据均对应相同的扫描深度，从第一行到第 S_r 行，对应的扫描深度依次等间隔从零增加到设定的探头最大扫描深度。每一列上的采样数据均对应相同的扫描角度，从第一列到第 S_c 列，对应的扫描角度（参考图 2 所示）依次等间隔从 $-\theta_0$ 增加到 θ_0 。

表 1

参数名称	参数的定义	符号	备注
凸阵探头的扫描角度	凸阵探头扫描时，所成扇形的角度	$2\theta_0$	单位为弧度
设定区域的相对扫描角	对应于设定区域 (ROI) 所成	$2roi_ \theta_0$	单位为

度	的扇形张角, 如图 3 所示		弧度
凸阵探头的曲率半径	曲率半径为凸阵探头的物理扇形半径	r_0	单位为物理单位毫米 (mm)
设定区域的相对曲率半径	为凸阵探头的相对物理扇形半径	roi_r_0	单位为物理单位毫米 (mm)
凸阵探头当前的扫描深度	与采样数据对应的凸阵探头的扫描深度	D_0	单位为物理单位毫米 (mm)
针对当前扫描深度 D_0 的设定区域扫描深度	设定区域的扫描深度	roi_D_0	单位为物理单位毫米 (mm)
采样数据的行数	参照图 4 所示	S_r 和	无量纲的参数
采样数据的列数	参照图 4 所示	S_c	无量纲的参数
设定区域的起始行	参照图 4, 该参数小于参数 S_r	top	无量纲的参数
设定区域的终止行	参照图 4, 该参数小于 S_r 且大于参数 top	$bottom$	无量纲的参数
设定区域的起始列	参照图 4, 该参数小于参数 S_c	$left$	无量纲的参数
设定区域的终止列	参照图 4, 该参数小于参数 S_c 且大于参数 $left$	$right$	无量纲的参数
显示在屏幕上的图像区宽度	进行 DSC 之后显示在屏幕上的图像区宽度	W_0	以像素为单位
显示在屏幕上的图像区高度	进行 DSC 之后显示在屏幕上的图像区高度	H_0	同 W_0
垂距	探头圆心到坐标系 $x-y$ 坐标轴 x 的垂直距离	d	单位为物理单位毫米 (mm)
设定区域的相对垂距	探头圆心到设定区域上部边	roi_d	单位为

	缘的距离		物理单位毫米 (mm)
--	------	--	----------------

设定区域的相对曲率半径的计算公式如式(1)所示:

$$roi_r_0 = r_0 + \frac{D_0}{Sr} \times top \quad (1)$$

设定区域的相对扫查角度的计算公式如式(2)所示:

$$roi_ \theta_0 = \theta_0 \times \frac{right - left + 1}{Sc} \quad (2)$$

针对当前扫描深度 D_0 的设定区域扫描深度的计算公式如式(3)所示:

$$roi_D_0 = D_0 \times \frac{bottom - top + 1}{Sr} \quad (3)$$

垂距的计算公式如式(4)所示:

$$d = r_0 \times \cos \theta_0 \quad (4)$$

设定区域的相对垂距的计算公式如式(5)所示:

$$roi_d = roi_r_0 \times \cos(roi_ \theta_0) \quad (5)$$

设定区域的相对坐标平移公式如式(6)所示:

参照图2可知,

$$\begin{cases} u = x + roi_u_0 \\ v = y + roi_v_0 \end{cases} \quad (6)$$

而

$$\begin{cases} u_0 = -W_0/2 \\ v_0 = \frac{roi_d}{roi_r_0 + roi_D_0 - roi_d} \times H_0 \end{cases} \quad (7)$$

由式(6)和式(7)可得

$$\begin{cases} u = x - W_0/2 \\ v = y + \frac{roi_d}{roi_r_0 + roi_D_0 - roi_d} \times H_0 \end{cases} \quad (8)$$

式(8)便是最终用于DSC进行放大显示的针对设定区域的坐标平移公式。

参照图3,在各个参数都确定的情况下,设从 $x-y$ 坐标系下的坐标 (x,y) 经过坐标平移之后对应的在 $u-v$ 坐标系下的坐标为 (u,v) ,而坐标 (u,v) 经过坐标变换后对应 $R-\theta$ 坐标系下的坐标为 (R,θ) ,可得坐标 (u,v) 到坐标 $R-\theta$ 的变换公式如式(9)所示:

$$\begin{cases} R = \sqrt{v^2 + u^2} \\ \theta = \arctan\left(\frac{u}{v}\right) \end{cases} \quad (9)$$

设设定区域的极径量化因子为 roi_q_r ,则 roi_q_r 的计算公式如式(10)所示:

$$roi_q_r = \frac{roi_D_0 + roi_r_0 - roi_d}{roi_D_0} \times \frac{bottom - top + 1}{H_0} \quad (10)$$

设极角量化因子为 roi_q_θ ,则 roi_q_θ 的计算公式如式(11)所示:

$$roi_q_\theta = \frac{right - left + 1}{2 \times roi_D_0} \quad (11)$$

设量化后的极坐标 $(R_{roi_q}, \theta_{roi_q})$,经过坐标平移和极坐标变换后的极坐标为 (R,θ) ,则它们之间的关系如式(12)所示:

$$\begin{cases} R_{roi_q} = R \times roi_q_r \\ \theta_{roi_q} = \theta \times roi_q_\theta \end{cases} \quad (12)$$

为了让极径和采样数据的编号能够统一，需要给极径去偏，设定极径去偏常数为 roi_F_R ，则 roi_F_R 如式 (13) 所示：

$$roi_F_R = -\frac{bottom-top+1}{roi_D_0} \times roi_r_0 \quad (13)$$

为了让极角的 0 位置和采样数据的第一列对齐，需要给极角去偏，设定极角去偏常数为 roi_F_θ ，则 roi_F_θ 如式 (14) 所示：

$$roi_F_\theta = \frac{left+right+1}{2} \quad (14)$$

设量化去偏后的极坐标为 (R_{qf}, θ_{qf}) ，经过坐标平移和极坐标转换后的坐标为 (R, θ) ，则它们之间的关系如式 (15) 所示：

$$\begin{cases} R_{qf} = R \times roi_q_r + roi_F_R \\ \theta_{qf} = \theta \times roi_q_\theta + roi_F_\theta \end{cases} \quad (15)$$

3、建立针对放大显示区域的坐标转换表

坐标转换，其运算量非常之大，但是同时又有一定的规律。如果每次在进行坐标转换的时候都进行一次运算，那么将会影响到 CPU 的耗用，而且会降低帧频，达不到实时性的要求。创建一个二维数组——二维查找表进行坐标的转换，将解决上述问题。

在如图 3 所示的扫描深度 D_0 下，对应的坐标转换表的行数为：

$$H_0 \quad (16)$$

由于显示像素区的宽度为 W_0 ，而且 W_0 在显示屏幕确定的情况下也不会发生变换，所以坐标转换表的列数定为 W_0 。也就是说，如图 3 所示的扫描深度 D_0 下，对应的坐标转换表的列数为：

$$W_0 \quad (17)$$

根据式(16)和式(17)并设坐标转换表 CTT (Coordinate Transform Table) 的行数为 CTT_{row} , 列数为 CTT_{col} , 则

$$\begin{cases} CTT_{row} = H_0 \\ CTT_{col} = W_0 \end{cases} \quad (18)$$

则式(18)为最终确定坐标转换表大小的计算公式。

根据式(5)、式(8)、式(9)、式(15)和式(18), 按照输入坐标 (x, y) 进行循环转换到 (R_{qf}, θ_{qf}) 即可, 其中 $0 \leq x < CTT_{col}$, $0 \leq y < CTT_{row}$ 。

4、建立针对放大显示区域插值计算表

由双线性插值运算公式可知, 要计算极坐标点 (R_{qf}, θ_{qf}) 处的灰度值, 就需要该点周围与之距离最近的四个采样点的值。求得 R_{qf} 的整数部分, 设为 i , 求得 θ_{qf} 的整数部分, 设为 j 。那么, 通过极坐标系下的点 (i, j) 、点 $(i, j+1)$ 、点 $(i+1, j)$ 和点 $(i+1, j+1)$ 四个点的灰度值, 即可双线性插值点 (R_{qf}, θ_{qf}) 的灰度值。

由式双线性插值运算公式可知, 如果设插值系数 α 和 β 且 $\alpha = R_{qf} - i$, $\beta = \theta_{qf} - j$, 并设极坐标下的点 (r, θ) 的灰度值为 $G(r, \theta)$, 则

$$\begin{aligned} G(R_{qf}, \theta_{qf}) = & (1-\alpha) \times (1-\beta) \times G(i, j) + \\ & (1-\alpha) \times \beta \times G(i, j+1) + \\ & \alpha \times (1-\beta) \times G(i+1, j) + \\ & \alpha \times \beta \times G(i+1, j+1) \end{aligned} \quad (19)$$

插值表的内容:

(1) 如图3所示, 将显示像素区内的凸阵扫描区像素点 (x, y) 的坐标位置由二维转化为一维, 并保存下来。设为 DP (Display Position), 则:

$$DP = y \times W_0 + x \quad (20)$$

特别注意：不保存如图 3 所示的凸阵扫描区以外的像素点的坐标位置。

(2) 保存用到的四个矫正插值系数 (Correctional Coefficient), 并将它们从浮点数转化为 WORD (双字节) 整数。四个插值系数分别设为 CC_1 、 CC_2 、 CC_3 、 CC_4 , 则:

$$CC_1 = (1 - \alpha) \times (1 - \beta) \times 256 \quad (21)$$

$$CC_2 = (1 - \alpha) \times \beta \times 256 \quad (22)$$

$$CC_3 = \alpha \times (1 - \beta) \times 256 \quad (23)$$

$$CC_4 = \alpha \times \beta \times 256 \quad (24)$$

(3) 四个采样点的位置

保存插值需要的四个采样点的位置 (Sample Points/Position), 并将其坐标位置由二维转化为一维进行保存, 四个采样点的位置分别设为 SP_1 、 SP_2 、 SP_3 、 SP_4 , 则:

$$SP_1 = i \times Sc + j \quad (25)$$

$$SP_2 = i \times Sc + j + 1 \quad (26)$$

$$SP_3 = (i + 1) \times Sc + j \quad (27)$$

$$SP_4 = (i + 1) \times Sc + j + 1 \quad (28)$$

由于该结构包含有显示在屏幕上的点的一维坐标值 (Display Position), 包含有矫正的插值系数 (Correctional Coefficient), 还包含有采样点的位置 (Sample Points/Position), 所以简称该结构为

DPCCSPT (字母“T”代表 Table)。该结构的定义如下所示:

```
struct DPCCSPT
{
    int DP;
    WORD cc1;
    WORD cc2;
    WORD cc3;
    WORD cc4;
    int P1;
    int P2;
    int P3;
    int P4;
};
```

使用 C++ STL 中的 vector, 向内存申请一个 DPCCSPT 类型的 vector, 申请的变量如下:

```
std::vector<DPCCSPT> FDisplayData
```

向变量 FDisplayData 中存放进行插值需要的数据

(R_{qf}, θ_{qf}) 是经过坐标变换表查找出来的极坐标值, R_{qf} 的整数部分为 i , θ_{qf} 的整数部分为 j , $\alpha = R_{qf} - i$, $\beta = \theta_{qf} - j$ 。

由于并不是显示像素区的所有坐标都有 B 超图像的信息, 所以要将非凸阵扫描区的坐标位置排除掉, 这样做的目的有二: 一是减少运算量, 提高了运算速度; 二是节约了内存存储空间。

存放数据的条件如下:

$$top \leq i \leq bottom \quad (29)$$

并且

$$left \leq j \leq right \quad (30)$$

这样就可以满足以上要求。

遍历 FDisplayData, 将其 DPCCSPT 类型的数据自行计算即可得到屏幕显示位置的灰度值。

5、快速计算进行DSC处理

举例来说, 假设要由 FDisplayData 中的第 k 个 DPCCSPT 类型的数据得到显示屏幕上的一个点的灰度值, 进行如下计算即可。首先设屏幕上第 m 个位置的灰度值为 DisGry (m), 另根据式 (19) 的设定可得:

$$\begin{aligned} \text{DisGry}(\text{FDisplayData}[k].\text{DP}) = & (\text{FDisplayData}[k].\text{cc1} * \text{G}[\text{FDisplayData}[k].\text{P1}] + \\ & \text{FDisplayData}[k].\text{cc2} * \text{G}[\text{FDisplayData}[k].\text{P2}] + \\ & \text{FDisplayData}[k].\text{cc3} * \text{G}[\text{FDisplayData}[k].\text{P3}] + \\ & \text{FDisplayData}[k].\text{cc4} * \text{G}[\text{FDisplayData}[k].\text{P4}]) >> 8 \end{aligned}$$

6、送达显示器进行显示

保存显示坐标的灰度值

由快速计算进行 DSC 处理已经知道, 在 DisGry[] 这个数组中, 已经将所有的凸阵扫描区的像素灰度值进行了保存, 当然, 在显示像素区的而又不在于凸阵扫描区的像素值在初始化的时候已经全部赋了零值而保存在该数组里。那么这个数组已经全部保存了将要显示在屏幕区宽度为 W_0 , 高度为 H_0 的所有像素的灰度值。

将已经保存了所有要显示区域灰度值的数组 DisGry[] 的首地址传递给显示函数即可, 从而完成了整个 DSC 从获取数据、坐标变换、插值处理到最后的送达显示器进行放大显示的全部过程。

图5总结了本发明对设定区域进行放大显示的处理流程:

1、建立坐标转换表CCT和插值表DPCCSPT, (已在上文中详细说明了实现过程)。

2、存放采样数据的数组 SD[Sr×Sc]在预处理中就要定义好, 大小是根据采样数据的大小设定的, 其大小为 Sr×Sc 字节, 数据类型为 BYTE 型, 用 C++ 的定义为: “BYTE SD[Sr*Sc];”该数组存放的是一帧采样数据

的灰度值，灰度值的范围为 0~255。比如该数组第三个 BYTE 型数据的值为 100，则可以认为 SD[2]的值为 100（因为数组的第一个值为 SD[0]，所以第三个为 SD[2]）。由于采样数据每隔一定时间会上传一帧，所以在同一个采样位置（数组中的同一个位置）其灰度值也是在不断发生变化的。

3、存放输出数据的一维数组 DisGry[$W_0 \times H_0$]：该数组存放如图 4 所示的显示像素区各个像素的灰度值，所以其大小为显示像素区的大小 $W_0 \times H_0$ ，数据类型为 BYTE 型，用 C++ 的定义为：“BYTE DisGray[$W_0 * H_0$]”。在进行快速 DSC 处理之前，存放输出数据的一维数组 DisGry[$W_0 \times H_0$]是被初始化过的。除非扫描深度或者显示模式或者探头更换，该数组不再进行初始化。初始化时，该数组的所有元素被赋值为灰度值的零，即为黑色。

4、快速 DSC 处理：在进行快速 DSC 处理时，根据计算插值表的限制条件，可以知道，数组 FDisplayData[]中保存的显示坐标位置 DP 均在设定的放大区域内。

图 6 是经过放大的图像示意图。

本领域技术人员不脱离本发明的实质和精神，可以有多种变形方案实现本发明，以上所述仅为本发明较佳可行的实施例而已，并非因此局限本发明的权利范围，凡运用本发明说明书及附图内容所作的等效结构变化，均包含于本发明的权利范围之内。

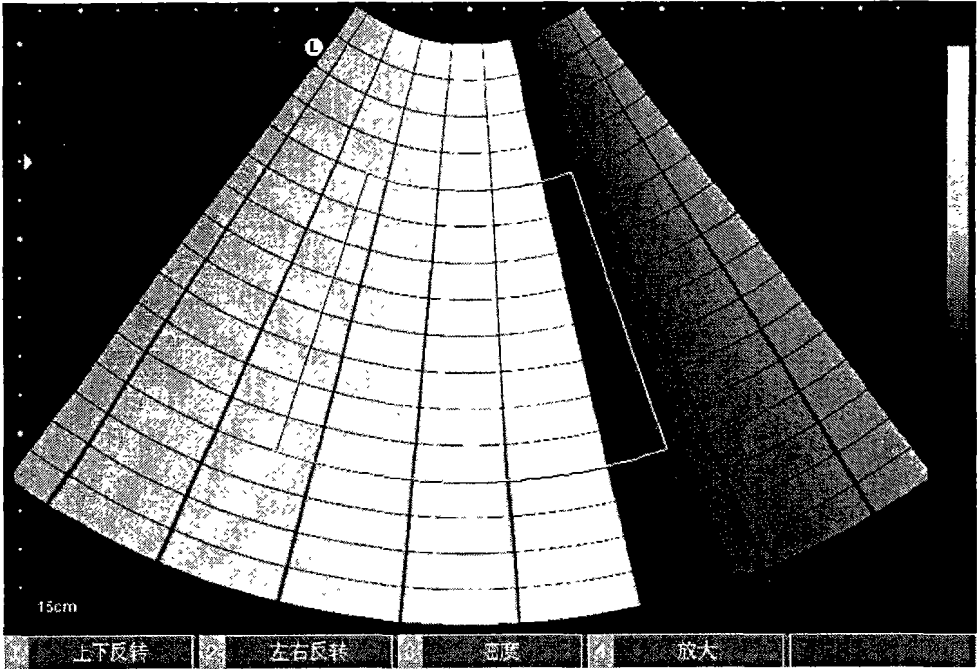


图1

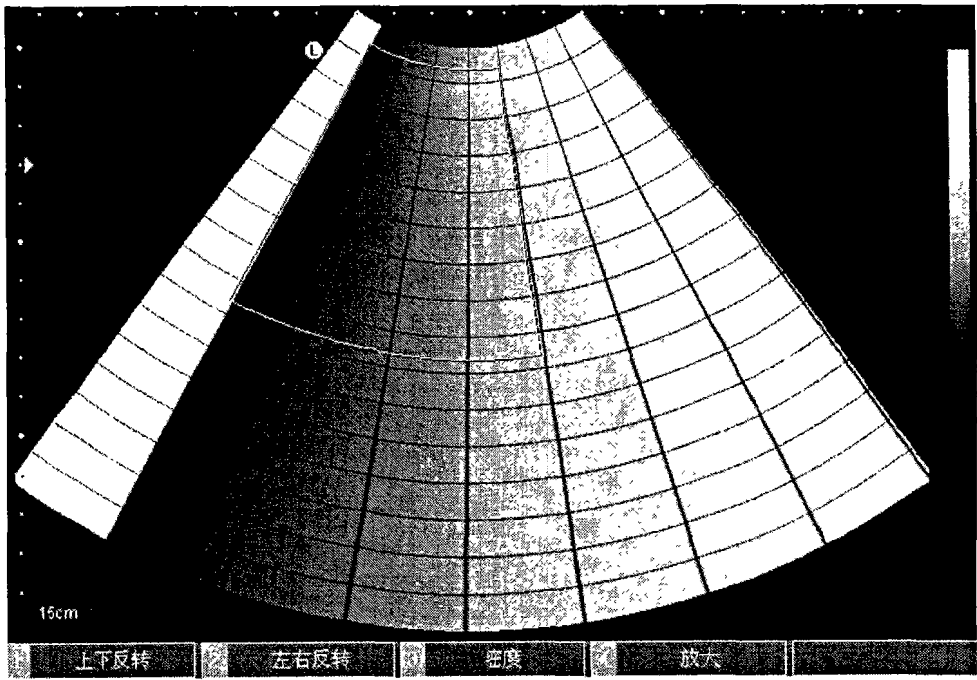


图2

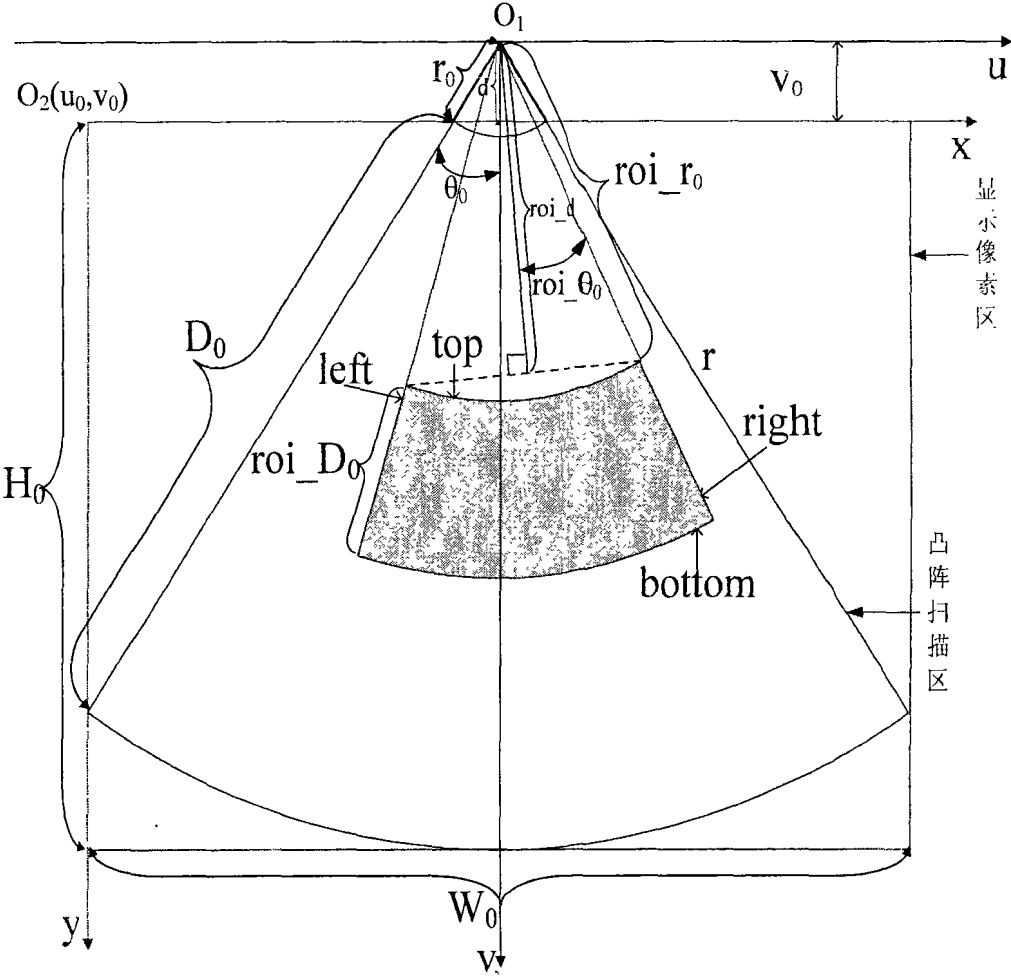


图3

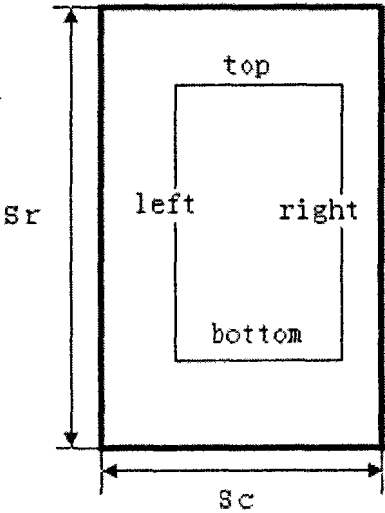


图4

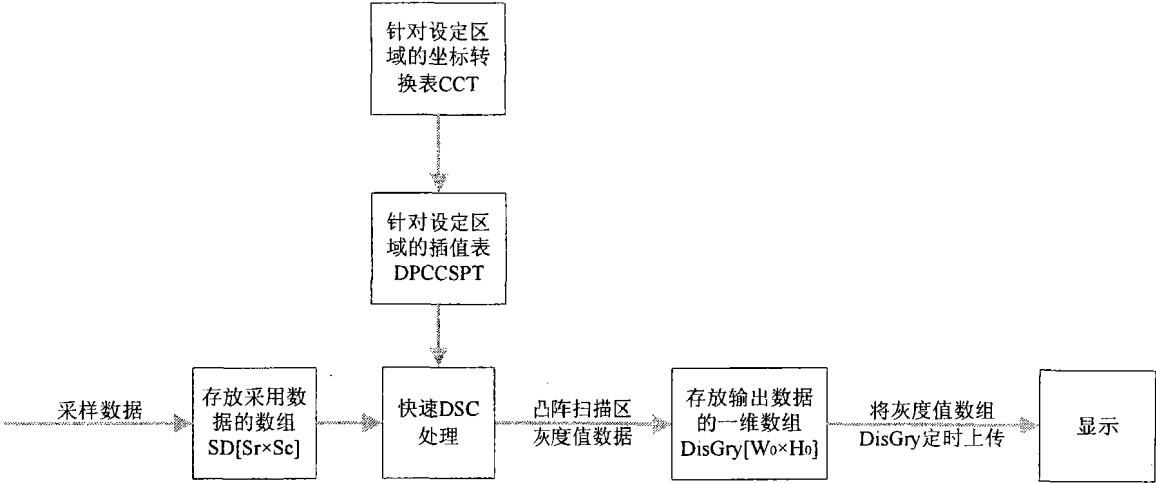


图5

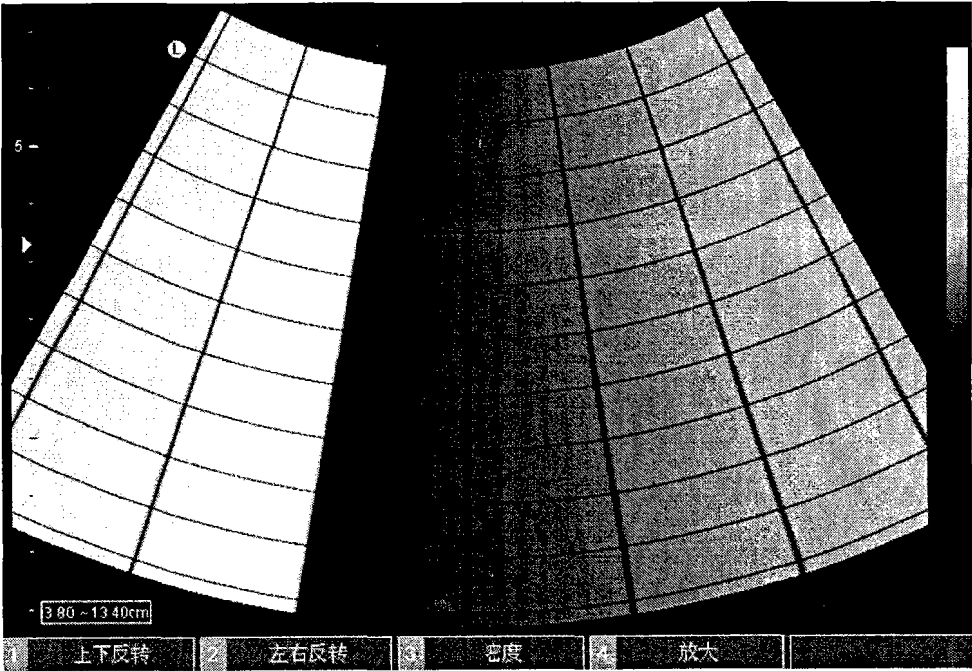


图6

专利名称(译)	一种超声图像放大显示的方法		
公开(公告)号	CN101375800A	公开(公告)日	2009-03-04
申请号	CN200710076823.2	申请日	2007-08-31
[标]申请(专利权)人(译)	深圳市蓝韵实业有限公司		
申请(专利权)人(译)	深圳市蓝韵实业有限公司		
当前申请(专利权)人(译)	深圳市蓝韵实业有限公司		
[标]发明人	程延俊 陈闽峰		
发明人	程延俊 陈闽峰		
IPC分类号	A61B8/00 G09G5/373 G06F19/00		
外部链接	Espacenet SIPO		

摘要(译)

本发明公开了一种超声图像放大显示的方法，包括步骤：A1.设定图像的放大区域，将所述放大区域的位置坐标保存；A2.根据所述位置坐标建立插值计算表；A3.根据所述插值计算表和采样数据计算所述放大区域内每个像素的灰度值数据；A4.将所述灰度值数据输出到显示器进行显示。本发明使用软件进行超声图像放大显示，省去了DSC的硬件处理模块，节约了硬件成本，缩小了硬件的体积，因为本发明直接对采样数据进行放大显示，因此放大的图像不会失真。

