Fast Hybrid Freehand Ultrasound Volume Reconstruction

Athanasios Karamalis[a,b], Wolfgang Wein[b], Oliver Kutter[a] and Nassir Navab[a]
[a]Chair for Computer Aided Medical Procedures (CAMP) Technische Universität München, Germany
[b]Imaging and Visualization Dept., Siemens Corporate Research, Princeton, NJ, USA

ABSTRACT

The volumetric reconstruction of a freehand ultrasound sweep, also called compounding, introduces additional diagnostic value to the ultrasound acquisition by allowing 3D visualization and fast generation of arbitrary MPR(Multi-Planar-Reformatting) slices. Furthermore reconstructing a sweep adds to the general availability of the ultrasound data since volumes are more common to a variety of clinical applications/systems like PACS. Generally there are two reconstruction approaches, namely forward and backward with their respective advantages and disadvantages. In this paper we present a hybrid reconstruction method partially implemented on the GPU that combines the forward and backward approaches to efficiently reconstruct a continuous freehand ultrasound sweep, while ensuring at the same time a high reconstruction quality. The main goal of this work was to significantly decrease the waiting time from sweep acquisition to volume reconstruction in order to make an ultrasound examination more comfortable for both the patient and the sonographer. Testing our algorithm demonstrated a significant performance gain by an average factor of 197 for simple interpolation and 84 for advanced interpolation schemes, reconstructing a 256^3 volume in 0.35 seconds and 0.82 seconds respectively.

Keywords: Visualization, Ultrasound, Volume Reconstruction, GPU

1. DESCRIPTION OF PURPOSE

Today's most widely used ultrasound imaging technique is freehand ultrasound, that when tracked (optic or magnetic) allows volume reconstruction from freehand sweeps. A number of reconstruction methods have been introduced [4] which can generally be categorized into forward and backward compounding methods.

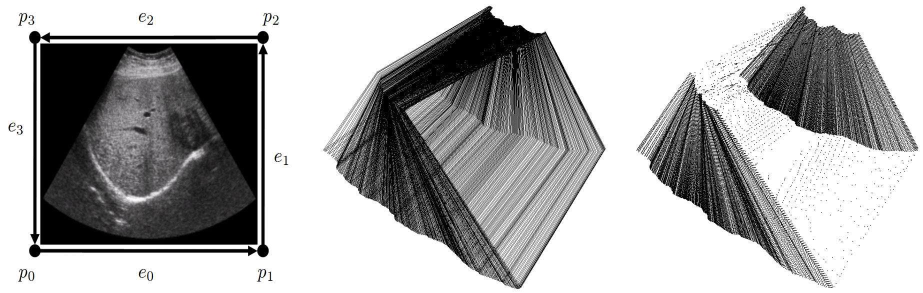
Forward compounding is qualitatively the inferior approach, projecting the acquired ultrasound slices directly into a volume. The lack of any interpolation results in gaps between the slices in the volume. This can be avoided by increasing the slice thickness for the projection, however the reconstruction quality remains poor. On the other hand, backward compounding is qualitatively the superior approach for which each voxel of the volume is calculated by taking into account adjacent slices. Different interpolation schemes have been introduced for the weighting of the adjacent slices [8], reconstruction based on probe trajectory delivering the best qualitative results so far [1].

As stated, the main concern of our work is to reduce the waiting time from acquisition to volume reconstruction, which especially becomes irritating when the acquisition has to be repeated multiple times, e.g. in the context of whole organ imaging for guiding interventional oncology procedures [7], or registration to a pre-operative plan for prostate radiotherapeutic delivery.

In the next sections we will present a high-performance reconstruction method that combines the forward and the backward approach, using graphics hardware to accelerate some of the steps required.

Further author information:(Send correspondence to Athanasios Karamalis)
Athanasios Karamalis: E-mail: akaramal@cs.tum.edu
Wolfgang Wein: E-mail: wolfgang.wein@siemens.com
Oliver Kutter: E-mail: kutter@cs.tum.edu
Nassir Navab: E-mail: navab@cs.tum.edu

Submitted to SPIE Medical Imaging 2009



(a)Image edge lines (b)Sweep edge lines (c)Intersection points

Figure 1. Visualizing algorithm steps

2. METHODS

In this section we will briefly present our algorithm starting with the required modeling of the input. Our algorithm gets a geometrical representation for each ultrasound image (slice) by defining a line equation for each edge as demonstrated in figure(1 a). Figure (1 b) shows a visualization of the image edge lines for a given ultrasound sweep.

Algorithm 1 Fast Hybrid Freehand Ultrasound Volume Reconstruction

```
Require: Sweep image edge lines
 1: Find optimal sampling direction
 2: Define sampling layers
 3: for all sampling layers do
 4:    for all slices do
 5:       if edge line not parallel to sampling layer then
 6:          find valid line-plane intersections
 7:          calculate texture coordinates
 8:       end if
 9:    end for
10:    render interpolated quadrilaterals
11: end for
```

As for the algorithm, the sampling layers compose the layers of a compounded volume and are defined using $\vec{n} \cdot (x - x_0) = 0$ where the plane normal $\vec{n}$ becomes the optimal sampling direction. Furthermore valid line-plane intersections occur when an intersection between an image edge line and a sampling plane result in a point that lies on the line segment defining the image edge. Figure (1 c) visualizes the intersection points for a given sweep. Only step 10 (render interpolated quadrilaterals) of the algorithm is performed on the GPU. More on calculating texture coordinates and rendering quadrilaterals follows.

2.1 MPR generation

For each sampling layer we render quadrilaterals (quads) utilizing the intersection points and texture coordinates of adjacent ultrasound slices by using common graphic APIs like OpenGL (in our case) or DirectX. Texture coordinate are simply calculated for each slice by dividing the Euclidean distance of an intersection point to an image edge point, by the length of the edge it lies on. Forming $n - 1$ quadrilaterals for n slices produces a MPR of the sweep for a given sampling layer. Furthermore by using quadrilaterals to connect adjacent ultrasound slices we avoid gaps that might arise, as in the case of forward compounding.

If the acquisition images the same anatomy twice, the most recent sweep imaging data will be reconstructed due to the successive drawing of slices. Repeatedly imaging the same anatomy with freehand ultrasound usually does not yield the exact same data, due to the highly dynamic imaging process (speckle noise, tissue deformation etc.); hence this behavior is desirable, unless explicit averaging is needed (similar to aperture compounding on ultrasound transducers).

In case of discontinues sweeps we cannot use quadrilaterals and have to fall back to rendering lines for each intersection between a slice and a sampling plane. This would result in a forward reconstruction with decreased quality.

2.2 Quadrilateral interpolation

Each pixel on the quadrilaterals is the result of a linear interpolation between the bilinear interpolated values of the adjacent ultrasound slices. However, even modern GPU hardware does not support direct rendering of quadrilaterals and decomposes them into two triangles which are processed separately. This results in a discontinuous attribute interpolation inside the quadrilateral [2], which is highly visible since we use irregular convex quadrilaterals for rendering. In order to deal with this problem we used the following two approaches. The first approach approximates an improved interpolation by splitting the quadrilateral into four triangles using the center point inside the quadrilateral. The final interpolation is performend in a simple fragment shader program that runs on the GPU [5]. The second approach is more sophisicated and defines a continuous attribute interpolation inside the quadrilateral by implementing general barycentric coordinates for irregular polygons [3] in a fragment shader program, making this approach the computationally more demanding one.

2.3 Volume generation

Nowadays most graphic cards provide the ability to 'render-to-volume' which allows redirecting the rendering from the display to an internal volume on the graphics card, without much effort. This volume can directly be used to generate arbitrary MPRs or for 3D rendering of the entire volume, whereas trilinear interpolation inside the volume is supported by graphic cards without any performance penalty. Moreover the volume can be read back into the RAM for further processing.

3. RESULTS

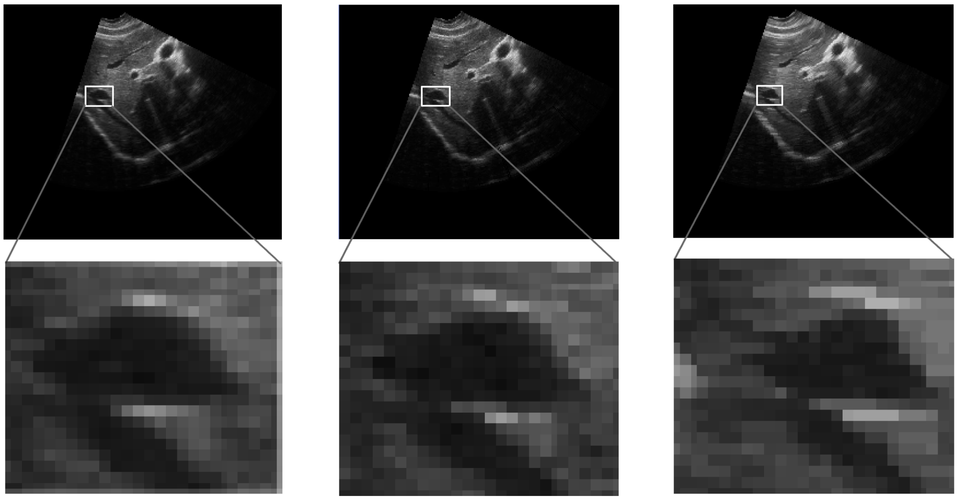
Our test system consisted of an Intel Xeon 3.2 GHz CPU with 2GB RAM and a GeForce 8800GTX 768MB GPU. We tested our method against a variety of established backward compounding approaches that have been implemented in a separate framework for the CPU and which enforce various acceleration techniques including efficient pre-sorting of the slices [8]. We also tested our method against a non-accelerated forward compounding implementation (not included in the mentioned average factors). The performance measurements for a dataset containing 293 slices and a target reconstruction volume size of 256^3 can be found in table 1. Additionally figure (2) presents a comparison between our method, the backward trajectory method and the backward maximum method. For a readback from GPU to host memory 0.275 seconds would be required for a 256^3 volume.

Table 1. Performance measurements with execution times for each method and the performance gain factor compared to the quadrilateral split and barycentric coordinate interpolation methods discussed in section 2.2

	Hybrid Quad Split	Hybrid Barycentric	Backward Trajectory	Backward Maximum	Backward Gaussian	Backward Selection	Backward Inverse	Forward
(sec)	0.35s	0.82s	72.72s	60.80s	69.78s	75.60s	66.52s	471.88s
	Gain	-	208x	174x	199x	216x	190x	-
	-	Gain	88x	74x	85x	92x	81x	-

4. CONCLUSION

In this paper we presented an efficient method for ultrasound volume reconstruction using graphics hardware that demonstrated a significant performance gain compared to other methods. The hybrid use of the forward and backward method not only ensures high-performance but also high reconstruction quality. The general simplicity of our approach leads to a fast re-implementation time and further encourages its use. The proposed technique yields on-the fly volumetric reconstruction, which can be of use in a great variety of clinical settings for visualization, registration and interventional navigation. In theory, it can be applied to any imaging modality that produces slices along a continuous trajectory in 3D space. Our method can be extended to accommodate for more sophistic geometric modeling of a sweep. In particular, taking the probe trajectory into account [1] can be implemented in the GPU fragment shader. Likewise, a two-dimensional curvilinear scan-conversion can be implicitly computed, allowing for high-fidelity reconstruction from raw US data, or efficient simulation of ultrasonic volumes [6].



(a)Hybrid (b)Trajectory (c)Maximum

Figure 2. MPR slices from reconstructed volume using different methods. Sweep shows a liver with a tumor on the left side.

REFERENCES

[1] Coup, P., Hellier, P., Azzabou, N., and Barillot, C. 3d freehand ultrasound reconstruction based on probe trajectory. In *Medical Image Computing and Computer-Assisted Intervention MICCAI 2005* (Sept. 2005), vol. 3749/2005, pp. 597–604.

[2] Hormann, K., and Tarini, M. A quadrilateral rendering primitive. In *HWWS '04: Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware* (New York, NY, USA, 2004), ACM, pp. 7–14.

[3] Meyer, M., Lee, H., Barr, A., and Desbrun, M. Generalized barycentric coordinates on irregular polygons. *journal of graphics tools 7*, 1 (2002), 13–22.

[4] Rohling, R., Gee, A., and Berman, L. A comparison of freehand three-dimensional ultrasound reconstruction techniques. *Medical Image Analysis 3*, 4 (1999), 339–359.

[5] Rost, R. J. *OpenGL(R) Shading Language (2nd Edition)*. Addison-Wesley Professional, 2006.

[6] Shams, R., Hartley, R., and Navab, N. Real-time simulation of medical ultrasound from ct images. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI) (Submitted)* (2008).

[7] Wein, W., Brunke, S., Khamene, A., Callstrom, M., and Navab, N. Automatic ct-ultrasound registration for diagnostic imaging and image-guided intervention. *Medical Image Analysis 12* (Sept. 2008).

[8] Wein, W., Pache, F., Roeper, B., and Navab, N. Backward-warping ultrasound reconstruction for improving diagnostic value and registration. In *Medical Image Computing and Computer-Assisted Intervention MICCAI 2006* (Sept. 2006), vol. 4191/2006, pp. 750–757.