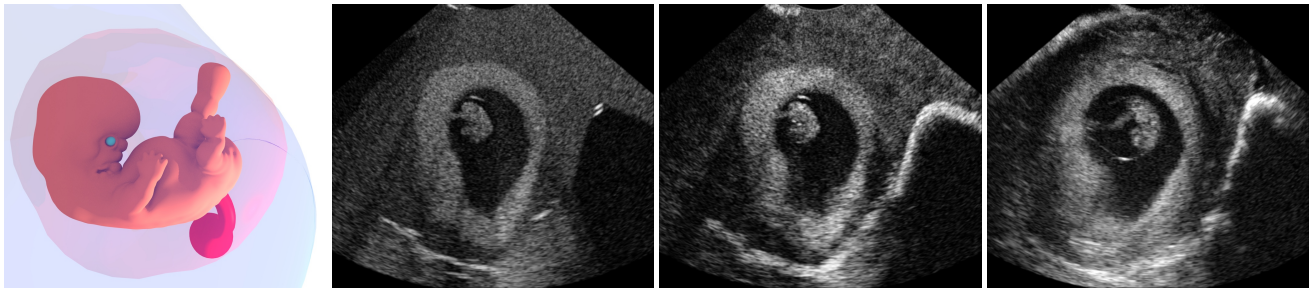
Eurographics Workshop on Visual Computing for Biology and Medicine (2016)
S. Bruckner, B. Preim, and A. Vilanova (Editors)

Monte-Carlo Ray-Tracing for Realistic Interactive Ultrasound Simulation

O. Mattausch and O. Goksel

Computer-assisted Applications in Medicine, Computer Vision Lab, ETH Zurich, Switzerland



(a) Embryo surface model (b) Deterministic ray-tracing (c) Monte-Carlo ray-tracing (d) Actual in-vivo US image

Figure 1: Comparison of deterministic binary ray-tracing (b) with our Monte-Carlo method (c) for ultrasound (US) simulation of pregnancy with a surface-based embryo model of 255K triangles (a). Note that the in-vivo embryo (d) is at slightly different age and position. Our method (c) runs at 1.4 FPS vs. 2.8 FPS for (b) for the given frame, although ours is casting 40× the number of rays per scanline.

Abstract

Ray-based simulations have been shown to generate impressively realistic ultrasound images in interactive frame rates. Recent efforts used GPU-based surface ray-tracing to simulate complex ultrasound interactions such as multiple reflections and refractions. These methods are restricted to perfectly specular reflections (i.e., following only a single reflective/refractive ray), whereas real tissue exhibits roughness of varying degree at tissue interfaces, causing partly diffuse reflections and refractions. Such surface interactions are significantly more complex and can in general not be handled by such deterministic ray-tracing approaches. However, they can be efficiently computed by Monte-Carlo sampling techniques, where many ray paths are generated with respect to a probability distribution. In this paper we introduce Monte-Carlo ray-tracing for ultrasound. This enables the realistic simulation of ultrasound interactions such as soft shadows and fuzzy reflections. We discuss how to properly weight the contribution of each ray path in order to simulate the behavior of a beamformed ultrasound signal. Tracing many individual rays per transducer element is easily parallelizable on modern GPUs, as opposed to previous approaches based on recursive binary ray-tracing.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1. Introduction

Ultrasound is a radiation-free, low-cost, and real-time medical imaging modality, and hence it is commonly used in clinical examination. However, its low signal-to-noise ratio and the existence of various ultrasound-specific artifacts necessitate extensive training of sonographers for performing a particular ultrasound examination technique and/or the imaging of a specific organ. Current standard education is in the form of imaging plastic phantoms and mannequins, as well as supervised examination of real pathology from patients during clinical practice. This approach requires significant time investment of qualified personnel and can be performed only when a supervisor and a patient are both available. A challenge then becomes the training on rare pathologies, e.g., Reis et al. [RLK*08] reports that during one year of standard education the medical students have a chance to learn only 80% of the important pathologies. For example, assuming an approximate 100 pregnancies to be at-

tended by an obstetrics and gynecology (Ob/Gyn) expert a year, heart-failure (a relatively common condition) will be observed on average once a year, whereas skeletal malformation (a rare 1/5000 condition) will take 50 years on average to encounter – longer than the typical career of an Ob/Gyn. These rare pathologies are still of utmost importance to identify, arguably even more so than common cases. More common decisions that are judged from US imaging, such as the assessment of the gestational age of an embryo in the face of a possible abortion, can still be crucial as they lead to life or death. These motivate the need for the computer-based training of ultrasound examination, where arbitrary scenes, pathologies, and embryo instances of different age/complexion can be simulated in a virtual-reality environment, given *realistic, real-time, and flexible* ultrasound simulation techniques.

An ultrasound transducer consists of several (e.g., 128 or 512) piezoelectric elements that can both transmit and receive ultrasound, through conversion between electricity and vibration. The transmitted pressure wave (ultrasound) then interacts with anatomical structures with different acoustic impedances, where any reflected signal is again digitized by the same elements to be used for generating an image of the underneath tissue. US interaction with tissue happens in two different ways: (*i*) On the one hand, structures much smaller than the US wavelength ($\approx 300\,\mu$m) absorb US energy and re-emit (scatter) it omni-directionally as point sources, such as cell nuclei, large proteins, etc. This interference pattern is indeed the source of the typical noisy texture of ultrasound, called the *speckle*. (*ii*) On the other hand, any macroscopic interface of impedance difference causes the US wave to be both reflected and refracted given its incidence angle. Accordingly, US may present both wave-like and ray-like properties, in a way similarly to light. Although the wave properties can be simulated in the entire domain, e.g. using finite difference methods, since the main ultrasound energy is focused in a certain direction, its wave-front can also be modeled as ray propagation in tissue.

2. Related Work

A typical model for simulating US interaction with sub-wavelength particles is to assume that tissue is populated with many (but countable number of) *scatterers* [OTV85]. These can then have varying scattering amplitudes, and act as spherical sources of scattering for incident ultrasound signal. Simulation frameworks like FieldII [Jen04] can accurately model ultrasonic wave propagations from given scatterer distributions, generating high-quality ultrasound B-mode images. Performance-optimized [VT05, Var06] and GPU-accelerated [KWN10] wave simulations for ultrasound exist, but are not fast enought to handle the complexity of clinical ultrasound in real-time. The computational complexity of such wave simulations restricts their use to offline simulation, e.g., for transducer design and image processing algorithm development and validation. In the context of training simulators, generation of US images at *interactive* rates is essential and hence of high interest. A viable approximation of the full wave model is the convolution model for ultrasound speckle [BD80, MBM87, MB95], where the received intensity of the ultrasound is obtained by convolving scatterers with the *point spread function* (PSF) of the incident ultrasound energy at that location. Assuming a PSF separable in 3 axes and a discrete grid approximation, fast separable convolution of scatters was shown to simulate speckles interactively on modern GPUs [GCC*09].

However, scatterer-based methods do not inherently take into account reflection and distortion of the ultrasound beam at the interfaces between large-scale structures, the source of artifacts such as acoustic shadowing, ghosting, and multiple-reflections. These are indeed important cues for differential diagnosis of pathology, and thus should be part of a realistic simulation. In ultrasound beamforming, incoherent (out-of-phase) ultrasound waves cancel out contributions of each other, creating a focused beam of ultrasound. The interaction of beamformed ultrasound with macroscopic structures like bones or organ surfaces can hence be well captured by ray-tracing techniques known from computer graphics. State-of-the art methods thus combine separable convolution for wave-like interference with fast surface-based ray-tracing [RPAS09, WBK*08], and in particular make use of dedicated GPU-based rendering frameworks of mesh representations such as NVidia Optix [BBRH13].

Current surface-based ray-tracing methods [BBRH13, SAP*15] utilize a recursive ray-tracing scheme: Whenever a ray terminates at a surface, a new refraction ray and a new reflection ray are cast according to Snell's law in a binary tree structure, until the contribution of a ray is smaller than a threshold. Such deterministic schemes make the assumption that large-scale structures behave like perfect mirrors with infinitely sharp reflections and refractions. This may be true for numerical phantoms and artificial tissue-mimicking material; however, in actual anatomy perfect reflections and refractions are never the case, hence these methods often produce artificial, toy-like images not comparable to actual patient ultrasound; as demonstrated in Figure 1. For instance, the state-of-the-art methods [BBRH13, SAP*15] evaluate a diffuse reflection term only *locally* (similar to Phong shading in computer graphics) but do not take non-specular indirect contributions into account, hence suffering from "hard" shadows, sharp reflections, and too crisp structure borders – atypical in real US imaging. Salehi et al. [SAP*15] aim to address these issues with a post-processing step to make images visually closer to expected; this, nevertheless, does not deal with the main problem of the wrong initial assumptions, which are treated in our work in a suitable and effective way within the ray-tracing architecture. Nonetheless, this method could in addition be integrated into our rendering pipeline to further improve realism.

For reflections from rough (imperfect) surfaces, contributions from many directions have to be considered. In a framework of deterministic ray-tracing, it is then conceivable to cast multiple rays at each intersection point of the parent ray in order to integrate over their individual contributions. Note that deeper recursion levels require summing over an exponentially-growing number of rays, although these contribute less and less to the final image. Therefore, this algorithm quickly becomes inefficient. Furthermore, such method exhibits poor parallelism, since multiple rays on subsequent levels are dependent on the single parent rays. The problem of computing (diffuse) multiple-reflections corresponds to numerically solving high-dimensional integrals, which are known to be approximated efficiently using Monte-Carlo methods. The idea is to generate many random ray paths that are perturbed according to

a probability distribution, until they converge to the correct solution. With this, utilizing the parallelism of the GPU is then straightforward, since each single ray path can be processed by each thread separately. Accordingly, many more rays can be processed in parallel on the GPU, maximizing throughput and yielding better quality of the resulting ultrasound simulation. The current state-of-the art in ultrasound simulation is analogous to so-called Whitted ray-tracing [App68, Whi80] for global illumination. Introduced in the late seventies, this method has long been shown to be inferior to stochastic techniques for natural scenes [VG97, Laf95].

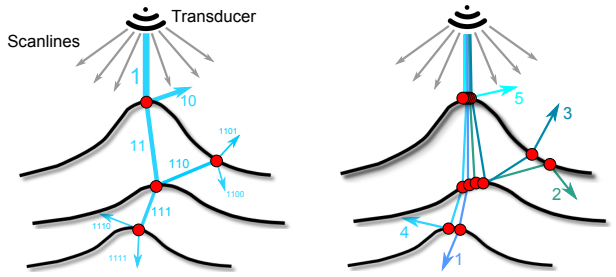
Surprisingly, however, such techniques have not been investigated in the literature for the purpose of ultrasound image simulation. Recently, Monte-Carlo techniques have been utilized for other uses of medical ultrasound, such as focused ultrasound therapy [KVdG*14], due to their excellent performance and inherent parallelism. However, the purpose of these techniques is not to emulate the image formation process of ultrasound machines, and since therapy ultrasound frequencies cannot be used for image generation, these methods cannot be used to simulate radio-frequency or B-mode images. As such, their goal is different than ours in this paper, which aims to present how to leverage stochastic techniques for the purpose of realistic ultrasound image simulation.

3. Contributions

In this paper, we propose a framework for using Monte-Carlo sampling for the purpose of ultrasound simulation. We extend the surface properties of ultrasound tissue by a measure of surface *roughness* (resulting in a random variation of reflected/refracted ray direction) and a measure of surface *thickness* (resulting in a random variation in the ray termination depth). We introduce a stochastic ray-tracing strategy for efficiently evaluating the surface integrals and propose a weighting scheme for a sample contribution that takes into account the beamforming process of US transducers.

For convolution-based US texture generation, we follow the approach of parametric models to instantiate scatterer distributions [BBRH13, MG15]. However, when adopting this approach, we have realized that these current parametric scatterer models are not sufficiently general to model the visual richness and variations in actual patient data. As can be seen in Fig. 1, actual tissue often exhibits a range of small- to large-scale variations of speckle appearance. We thus propose a 2-level parametric model using a multi-texturing approach to represent both the high-frequency local speckle variations and low-frequency large-scale speckle variations. In order to keep track of current tissue characteristics and to choose correct parametrization for tissue that a ray is inside, we propose a novel stack-based approach, based on three simple rules.

Our method is able to achieve vastly improved realism for interactive ultrasound simulation in complex and realistic scenarios, e.g. for simulating an embryo. In the teaser scene, in comparison to the state-of-the-art, our method is seen to successfully simulate soft shadows, fuzzy reflections, and realistic anatomical boundaries; with appearances more similar to the actual US image. Our method is also seen to exhibit more realistic ultrasound speckle (texture), which is less regular and richer in details, as well as to be more successful in simulating complex geometric shapes such as the complex embryo model. The only structure that cannot be replicated by (either) simulation is the bright reflection on top left, due to a complex out-of-plane 3D effect. The input to our method is a mesh-based representation of a scene, therefore the scene is easily customizable (e.g., for simulating embryos of any stage of development and with arbitrary ultrasound parameters) and can potentially be animated.



(a) Deterministic (binary tree) (b) Monte-Carlo (paths)

Figure 2: Deterministic binary ray-tracing vs. Monte-Carlo ray-tracing for a single scanline intersecting 3 tissue layers.

4. Methods

In this section, we first describe our improved stochastic surface ray-tracing model for simulating a beamformed US signal, outputting ray segments between tissue boundaries. Thereafter, we introduce a novel tissue model to characterize the tissue properties along individual ray segments. Finally, we put it all together and describe the actual US image generation process.

4.1. A Novel Ray-Tracing Model for Ultrasound

The so-called radio-frequency (RF) signal is created by recording US echo at several elements and then combining (beamforming) those into a single reading of US signal intensity I as a function of time t. Thus I represents the tissue content along a beamformed RF line, often aligned in front of a transducer element. Following Snell's law, the ultrasound signals undergo distortions (e.g., reflections or refractions) proportional to their impedance ratios at the interfaces between two tissues, which coincides with a ray-surface intersection in a ray-tracing context. In our model, such distortions are not deterministic but subject to random perturbations. If echos following different paths arrive at the transducers at the same time t, their partial contributions create an interference, adding up to the intensity received at that RF-line. Similarly, if the US signal disperses in several directions after a ray-surface intersection, their partial contribution from each adds up to the echo reading from that direction. In particular, when the US signal amplitude $I(t,O)$ received at the RF-line origin O after a surface intersection at point P_T for time $t > T$ is to be computed, we integrate the partial contributions from all directions ω of the surface hemisphere Ω centered at the surface normal:

$$I_{t>T}(t,O) = \int_{\Omega} I(t,P_T,\omega)\cos(\omega)d\Omega \quad . \tag{1}$$

$I(t,P_T,\omega)$ denotes the echo contribution from direction ω originating from P_T. At a further intersection point $P_{T'}$, each ray path may split up again, i.e.,

$$I_{t>T'}(t,P_T,\omega) = \int_\Omega I(t,P_{T'},\omega')\cos(\omega')d\Omega \quad . \tag{2}$$

Computing the echo interference at origin O at time $t > T'$ requires the solution to a high-dimensional recursion of integrals of the form

$$I_{t>T'}(t,O) = \int_\Omega \int_{\Omega'} I(t,P_{T'},\omega')\cos(\omega')d\Omega'\cos(\omega)d\Omega \quad , \tag{3}$$

the dimensionality of the integral being directly proportional to the number of surface intersections encountered. In practice, Eq. 1 can be discretized as

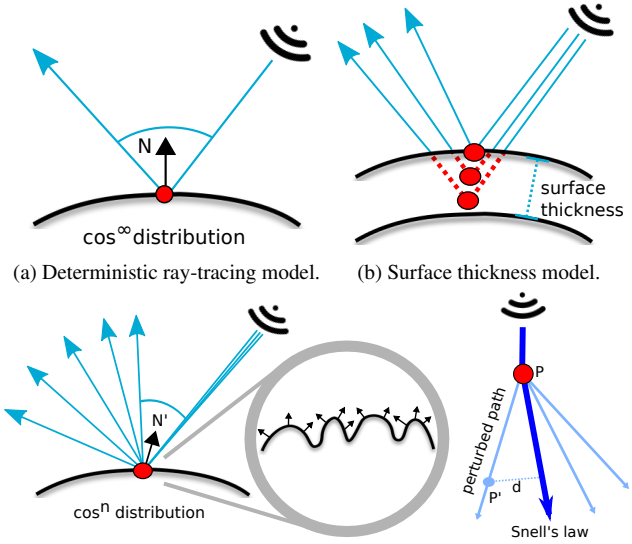
$$I_{t>T}(t,O) = \frac{1}{n}\sum_{i=1..n} I(r(P_T,\omega_i))\cos(\omega_i), \tag{4}$$

where each individual contribution $I(R(P_T,\omega_i))$ is calculated by tracing ray R from P_T in direction ω_i, which is drawn from a distribution over all possible hemispherical directions. Note that deterministic algorithms are not suited well for solving such high-dimensional integrals, while also incurring exponential running time wrt. the number of recursions, as illustrated in 2(a) for 3 surface layers. Instead, these are often tackled using stochastic (e.g., Monte-Carlo) methods to reduce the multi-dimensional integral to a tractable sampling problem (Fig.2 (b)).

Improved surface model. Assuming a cosine-parametrized surface model (see Fig. 3(c)), where the normal is perturbed around the geometric surface normal, we utilize importance sampling for calculating the surface integral at an intersection point: rays are distributed according to a $\cos^n$ distribution; where n=0 corresponds to a completely diffuse surface, while n=∞ to a perfectly specular surface. We decide randomly with a probability of 50% whether a reflective or refractive ray is to be generated at an intersection point.

Stochastic ray tracing allows for varying surface thicknesses to be simulated with our method, through varying the simulated hit point of the surface and the starting point of the reflected/refracted ray path according to a Gaussian distribution along the surface thickness, cf. Fig. 3(b). It is important to correctly attenuate a penetrating ray according to the attenuation and the penetrated distance in the new tissue. The attenuated ultrasound intensity in tissue is calculated as $Ie^{-x\alpha}$ using the Beer-Lambert law, for a distance x traveled in the tissue, the absorption coefficient α being a known characteristic tissue-specific value. For example, bone has a very high attenuation α, hence ultrasound can only penetrate a little amount before a sharp drop in intensity, causing the typical appearance of bone reflections in Fig. 1.

Path weighting. The cancellation of out-of-phase ultrasound reflections due to beamforming is an important ultrasound property, which has the effect that incoherent ultrasound beams lead to destructive interference diminishing the echo intensity. To simulate this effect, we use a weighting scheme for the contribution of each ray path in Eq. 4, as shown in Fig. 3(d). We use a linear factor based on distance from a hypothetical ray path representing the main beam direction, i.e., the direction l given by Snell's law (note that this ray path is not actually generated). This distance is computed as $d = |P - P' - ((P-P')\cdot l)l|$, where P is the origin of the ray, and P' is any point along the ray path.



(a) Deterministic ray-tracing model. (b) Surface thickness model.

(c) Surface roughness model. (d) Path weighting.

Figure 3: (a) Previous work uses infinitely thin, specular surfaces. (b) Our proposed surface model assumes intersection points to be distributed within the thickness of interacted surface, and (c) microscopic surface normals to be cosine-distributed around a main surface normal. (d) Weighting wrt. distance to the main beam is used to simulate the loss of beam coherence.

Incident US energy. At each tissue intersection P, the US echo I_r reflected back to the transducer is computed as

$$I_r = \max(\cos^n(\angle(\vec{V}_r,\vec{D})),0) + \max(\cos^n(\angle(\vec{V}_i,\vec{D})),O), \tag{5}$$

where $\vec{V}_r$ is the reflected and $\vec{V}_i$ is the refracted ray directions, and $\vec{D} = P - O$ for transducer origin O, and n models the specularity of a surface. Note that the refractive term was missing in previous work [BBRH13]. Both terms are important, however, as usually only one of them has non-zero contribution.

4.2. Speckle and Tissue Properties

One contribution to the received ultrasound signal, the directional reflections at large-scale surface interfaces, has been discussed in the previous section. In this section we discuss the other contributing factor coming from microscopic scatterers in the tissue.

Convolution model. In the typical convolution model of ultrasound speckle [BD80, MBM87, MB95], the reflective intensity $r(l,a,e)$ is obtained by convolving the a function representing tissue $g(l,a,e)$ with an ultrasound PSF $h(l,a,e)$ given additive random noise ε, i.e.

$$r(l,a,e) = g(l,a,e) * h(l,a,e) + \varepsilon \tag{6}$$

where l denotes the lateral, a the axial, and e the elevational transducer axes. In recent works [GCC*09, BBRH13, MG15], $h(l,a,e)$ is approximated by a Gaussian envelope modulated at an acquisition center-frequency of f_c (typically between $2-14$Mhz for clinical

use). It is then of the form

$$h(l,a,e) = e^{-\frac{l^2}{2\sigma_l^2} - \frac{a^2}{2\sigma_a^2} - \frac{e^2}{2\sigma_e^2}} \cos(2\pi f_c) \,, \tag{7}$$

which is a function, separable into lateral, axial, and elevational components. This greatly accelerates the convolution computation, by performing separate 1D convolutions instead of a single 3D one.

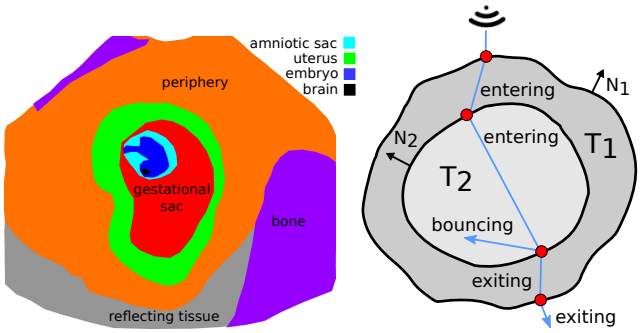
The PSF has been set manually for the purpose of this paper, but alternatively it can either be measured experimentally by imaging sub-wavelength synthetic features, e.g. wires, in degassed water, or estimated automatically from a radio-frequency image [MG16].

Scatterer parametrization. For obtaining the tissue parameters used to create $g(l,a,e)$, we adopted and improved an efficient parametric scatterer model [BBRH13,MG15]. For each tissue type, this model has 3 values to characterize a random distribution of scatterers, which are responsible for a particular speckle appearance after convolution: The pair (μ,σ) describes the normal distribution of scatterer amplitudes, while ρ is the so-called scatterer density, that gives the probability threshold above which a scatterer will provide an image contribution (i.e., the convolution will be performed for that scatterer). Each tissue is characterized by these three values, using which a scatterer value is generated from a random texture variable at each point.

To apply this model with a low memory footprint and high runtime performance, we follow the approach of Burger et al. [BBRH13], where a random Gaussian variable ξ is stored in a 3D texture and used for lookup during the ray marching. For $\xi > \rho$, the scatterer value $c = \mu + \xi\sigma$; whereas $c = 0$ otherwise. To reduce memory requirement while keeping the scatterer resolution high, the 3D texture only covers a small part of the US volume and is replicated for the rest, hence can be set to an easily manageable size, e.g., of 256^3. In order to avoid artifacts revealing the repetitions for a transducer orientation perfectly aligned with the texture, the 3D texture coordinates used for the texture lookup are transformed by a constant global rotation by a random angle.

Multi-frequency scatterer variations. In order to represent a smooth small-to-large scale speckle variation, we extend the basic model with a coarser component using a second low-resolution 3D noise texture $V_l(x,y,z)$. In our experiments, this resolution was empirically chosen as 20^3. Accordingly, two new values are added to the tissue specification; the magnitude A_l and the frequency f_l of low-frequency variation. These variations are implemented by using a lower-frequency 3D noise texture, where f_l acts as a scale factor for the look-up into the noise and A_l controls the influence of random variation on the scatterer values, i.e. the new scatterer value is computed as $c' = c(1 + A_l V_l(xf_l, yf_l, zf_l))$.

Interleaved volumetric model. Anatomical scenes as in Fig. 1 consist of several tissue layers that participate in complex ray interactions. An approximate segmentation of tissue layers in the US plane is depicted in Fig. 4(a). This particular scene consists of seven interleaved layers: uterus, placenta, gestational sac, amniotic sac, embryo body, skeleton, brain and inner organs. Whenever the current ray intersects a surface, one has to keep track of the correct tissue characteristics and scatterer parametrization that is used in the next segment, from the current intersection point until the ray hits another surface boundary. Our model for the bookkeeping of



(a) Pregnancy scene segmentation map. (b) Layered surface model.

Figure 4: Suitable models for in-vivo ultrasound, like the pregnancy scene (a), consist of multiple layers. For transmitting rays entering/exiting a layer (b), tissue properties are pushed on or popped from the stack. Reflective rays just bounce within the same tissue.

current tissue properties is inspired by classical methods for ray-tracing Constructive Solid Geometry (CSG) [Gla89]. In order to find the tissue type within which a particular ray segment is travelling (and thus the corresponding parameters for ultrasound speckle simulation), we propose a stack-based system based on three simple rules, as illustrated in Fig. 4 (b):

Entering The current ray is a transmitting ray, and the ray is entering the current object. In this case, the surface normal is facing into the opposite direction as the ray; e.g., normal N_2 as the ray hits the boundary from T_1 to T_2.

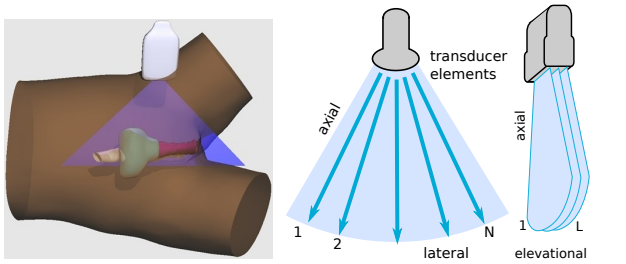
Exiting The current ray is a transmitting ray, and the ray is exiting the current object. In this case, the surface normal is facing into the same direction as the ray; e.g., normal N_2 as the ray hits the boundary from T_2 to T_1.

Bouncing If the current ray is a reflecting ray, it simple bounces within a tissue, so the current tissue region and hence properties stay unchanged.

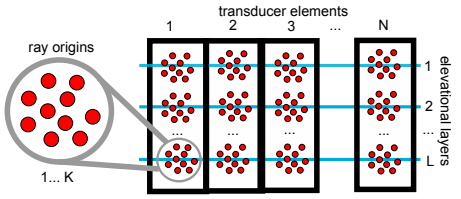
To avoid tissue classification errors due to numerical errors, a constant surface offset δ=$1e^{-4}$ is used (as is common in conventional ray-tracing applications to avoid light leaking or shadow artifacts). Furthermore, the interleaved surface meshes should not contain holes, have a consistent normal orientation, and should not intersect or touch each other. Note that the proposed layered model, although introduced in this paper, is used in both deterministic and Monte-Carlo simulations in all shown experiments.

4.3. Ultrasound image generation.

A typical 2D ultrasound acquisition scheme is depicted in Fig. 5 (a) and (b). The ultrasound is acquired in the so-called transducer plane (a). A ray is started from each of the K transducer element and continuously travels within a tissue yielding the echo signal along that scanline (b). To account for finite transducer thickness which leads to a non-negligible out-of-plane effect, we instantiate rays at several acquisition layers along the elevational (out-of-plane) axis of the transducer, as depicted in Fig. 5 (c), a strategy which was also used in previous work [BBRH13, SAP*15]. To initialize our



(a) Transducer plane. (b) Acquisition scheme. (c) Layers



(d) Perturbed ray origins.

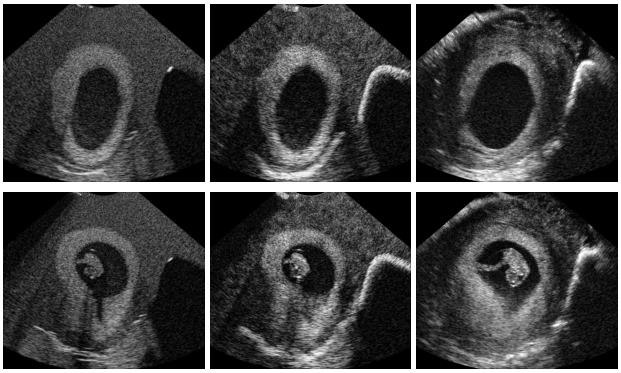
Figure 5: US B-mode is a 2D image in the transducer plane (a) generated by recording echo amplitudes for each transducer element (b). Out-of-plane effects are simulated using a small number of elevational acquisition layers (c). Stratified sampling is used to generate origins for K ray paths per element/layer (d).

Monte-Carlo ray-tracing scheme, K ray paths are originated around the center point of each scanline/elevation layer following a normal distribution, as shown in Fig. 5 (d). Their intensity contributions for the transducer element are accumulated according to Eq. 4. This stratified sampling scheme also avoids aliasing effects due to regular sampling of an elevation layer.

To find the speckle contributions along all participating rays and to form an ultrasound B-mode image, we use the following steps. First, our surface ray-tracing model is used to find the ray path segments between tissue boundaries, corresponding to the current tissue being traveled in. Then we march stepwise along the ray segment, averaging the scatterer contributions of our improved tissue model in a small volume around each sample point. The result is stored in a 3D grid $g(l,a,e,)$, consisting of N lateral scanlines, M axial sample points, and L elevation layers. Each axial sample point in $g(l,a,e,)$ corresponds to the distance traveled at time $t_{i=1..M}$ along the ray segment. The time step is given by the desired RF sampling frequency to simulate, which is typically around $20-100$MHz, and the distance traveled per timestep is given by the speed of sound in the current tissue. The regular structure of $g(l,a,e)$ enables us to run fast separable PSF convolution with PSF $h(l,a,e)$ in parallel, using Eq. 6. The result is $r(l,a)$, the simulated beamformed US radio-frequency (RF) image in the transducer plane. Emulating typical clinical imaging systems, we then input this RF image to a post-processing pipeline (envelope computation, time-gain compensation, log compression, and scan-conversion into radial coordinate, if a convex transducer is being simulated), which yields the final grayscale B-mode image displayed on US machines.

Model	Spherical	Embryo	Heart
Triangles	1.5K	255K	325K
Full depth	6.0cm	8.1cm	6.0cm
Transducer freq.	6.MHz	7.MHz	6.MHz
Transducer FOV	99°	75°	90°
Scanlines	128	192	128
Axial res.	1024	2048	1024
Elevation layers	5	5	5
Axial Sampling	.06 mm	.04 mm	.06 mm
Scatterer res.	.03 mm	.03 mm	.03 mm
Animation	No	No	Yes
Frame times			
Deterministic	107ms	357ms	201ms
MC 5 rays	106ms	303ms	108ms
MC 15 rays	139ms	384ms	128ms
MC 25 rays	169ms	476ms	164ms
MC 40 rays	232ms	714ms	190ms
Throughput (rays/sec)			
MC 5 rays	45K	15K	44K
MC 15 rays	104K	37K	113K
MC 25 rays	142K	50K	146K
MC 40 rays	166K	54K	202K

Table 1: Statistics and timings for the used datasets.



(a) Deterministic (b) Monte-Carlo (c) In-vivo image

Figure 6: Two additional slices of the embryo scene. Comparison of deterministic ray-tracing (a), our Monte-Carlo method (b), and a corresponding in-vivo image (c).

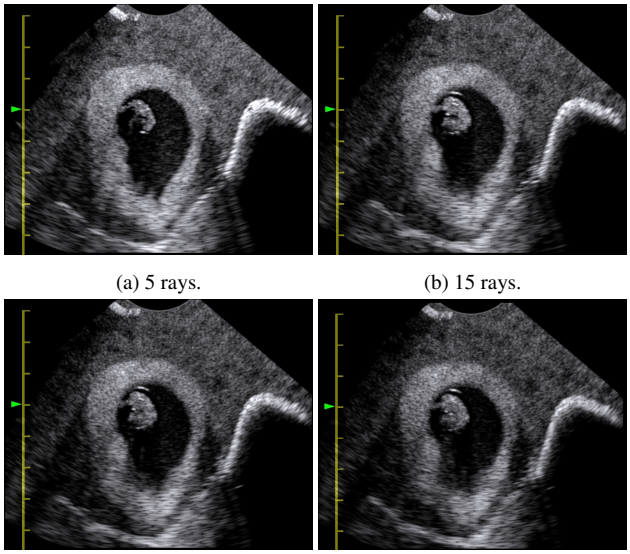
5. Results

Our method was developed in C++, using NVidia Optix as framework for the mesh-based ray-tracing, and NVidia Cuda v7.0 for the ultrasound post-processing (image formation) pipeline. An NVIDIA GTX 780Ti GPU with 3GB was used for all experiments. Detailed statistics of the presented scenes and timing comparisons are shown in Tab. 1. The scales of individual ultrasound images are marked in centimeters on the left of the images, together with a green arrow indicating the chosen transmit beamforming focus.

The embryo scene was shown in Fig. 1. The embryo was designed as an anatomically correct 3D surface model, whereas the gestational sac, uterus, and nearby bones were annotated from the

(a) Hypoechoic spherical object. (b) Simple heart model.

Figure 7: Additional datasets used for experiment and their in-plane segmentation layout.



(a) 5 rays. (b) 15 rays.

(c) 25 rays. (d) 40 rays.

Figure 8: Monte-Carlo ray-tracing using a varying number of rays per transducer element/elevation level.

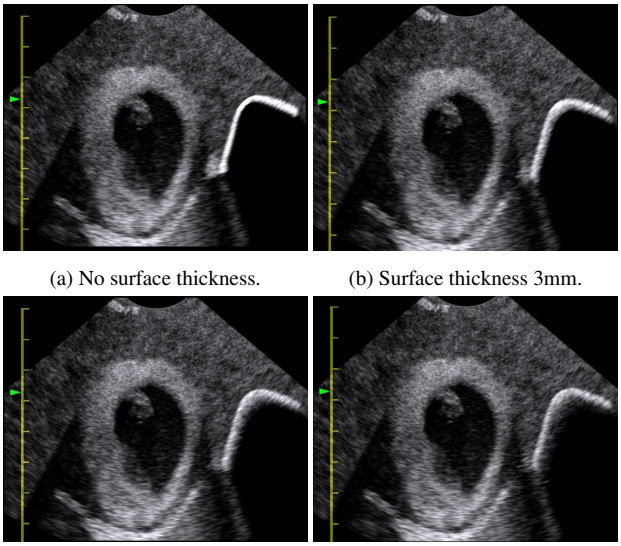
scanned 3D image volume and exported as surface meshes. The image shown in Fig. 1 (d) is a single slice of this fan-shaped scan volume, with a total of 215 slices spanning 120 degrees in out-of-plane direction. Other sample slices, at 95 and 110, can be seen in Fig. 6. The embryo model has detailed anatomy such as the brain, stomach, eyes, and a skeleton. This turns out to be necessary to simulate the visually rich content of the actual embryo, which is, as expected, at the focus of attention for a trainee during examination. Note that the hand-crafted embryo model in Fig. 1 (a) is at a slightly earlier gestational age than the embryo shown in the in-vivo image of Fig. 1 (d), therefore a 100% visual match is not possible.

The additional models used in our experiments are shown in Fig. 7. The first scene is an approximately spherical shape filled with a hypoechoic fluid. It represents a class of structures often found in in-vivo ultrasound images, like for example fluid-filled cysts or a full bladder. The heart model is a composition of interleaved spherical objects representing the heart shape and the 4 heart chambers, and was animated to simulate a periodic motion resembling a heart beat. Starting from perfect spheres, the objects were deformed individually to reduce regularity

The timings in Tab. 1 were recorded for a maximum of 12 recursion levels for both deterministic ray-tracing and Monte-Carlo ray-tracing (MC). The number of rays per transducer element/elevation levels for Monte-Carlo ray-tracing varies from 5 to 40. For 5 elevation levels per elements, this means 25-200 rays cast per element, resulting in a total of 480-38400 ray paths for 192 scanlines per frame. Note that the throughput of rays/second increases for a higher number of generated rays, exhibiting an increased level of parallelism. The slower render times of the embryo model compared to the heart model, despite the former having less triangles, can be attributed to the particularly complex in-plane and out-of-plane ultrasound interactions.

Level	Det	MC5	MC15	MC25	MC40
7	233 ms	300 ms	392 ms	472 ms	705 ms
10	299 ms	298 ms	384 ms	471 ms	709 ms
12	357 ms	303 ms	384 ms	476 ms	714 ms

Table 2: Comparison of rendering times for the embryo image using different recursion levels.



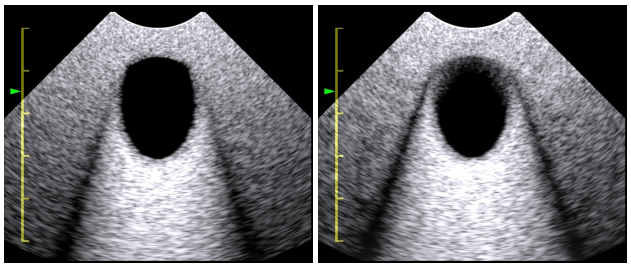
(a) No surface thickness. (b) Surface thickness 3mm.

(c) Surface thickness 6mm. (d) Surface thickness 9mm.

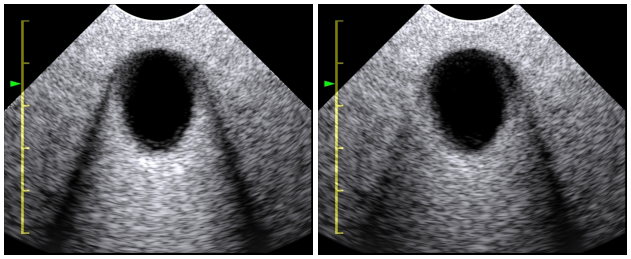
Figure 9: The effect of finite surface thickness on the realism of the bone.

Tab. 2 describes the relation between the number of recursion levels and the render times. Binary ray-tracing is greatly affected by this factor, which is unsurprising due to the exponential dependence of the algorithm complexity on the number of recursions. On the other hand, Monte-Carlo ray-tracing remains almost unaffected. The importance of the number of rays for the image quality can be observed in Figure 8 for $5-40$ rays. The sampling quality is high for ≥ 25 rays and in areas of less complex ray interactions, but decreases slightly for a lower number of rays in areas of complex ray interactions. For example, entire interactions are missed in the simulation of the embryo.

Fig. 9 and Fig. 10 demonstrate the effect of our surface model with roughness and finite thickness. In Fig. 9, ignoring the thickness parameter results in a flat and unrealistic bone appearance (a).

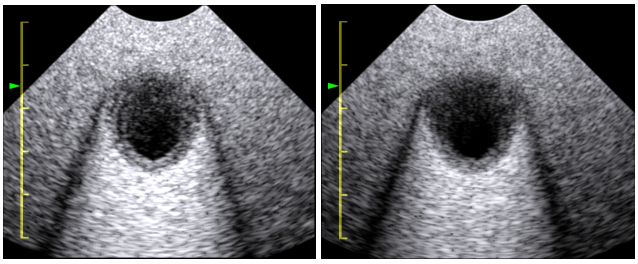


(a) Deterministic. (b) $\cos^{inf}$ distribution.



(c) $\cos^{100}$ distribution (d) $\cos^{10}$ distribution

Figure 10: The effect of the surface roughness on a spherical objects and on the brightness of the hyperechoic posterior region.



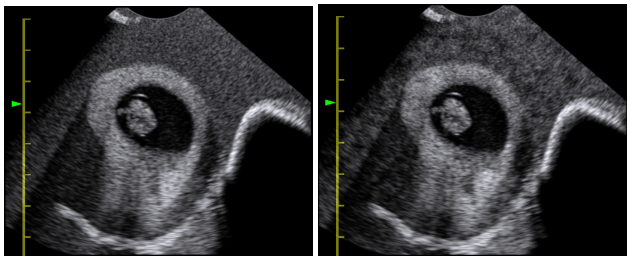
(a) No perturbation. (b) With perturbation.

Figure 11: The effect of perturbing the ray origin in this simple scene with a spherical object.

Increasing the finite thickness parameter influences the appearance of the bone, which becomes more plastic and organic for larger thickness values (b-d). The realistic attenuation of the thick bone segment is achieved because our model takes into account the correct attenuation with respect to the penetration distance that a ray travel inside of the bone.

In Fig. 10, (a) and (b) correspond to perfectly specular surfaces, with the difference that (a) corresponds to deterministic ray-tracing where perfectly specular surfaces are the default, while (b) still uses the other extensions of the proposed method like finite thickness. For images (c) and (d), the surface of the scanned object is set to become increasingly rougher. This results in softer refraction shadows and fuzzier object boundaries. For simulating a particular tissue, the proper value of surface roughness has to be set according to experiment or empirically by manual trial-and-error.

Fig. 11 depicts the importance of the stratified sampling scheme for the ray origins. In Fig. 11 (a), 5 elevation layers of ray origins have been used to approximate finite transducer thickness, as proposed by previous work [BBRH13]. The discrete layers can be clearly seen in the data, while these aliasing effects are not visible in Fig. 11(b) using our stochastic scheme.



(a) Without speckle variation. (b) With speckle variation.

Figure 12: The influence of large-scale speckle variation on realism in a frame of the pregnancy scene.

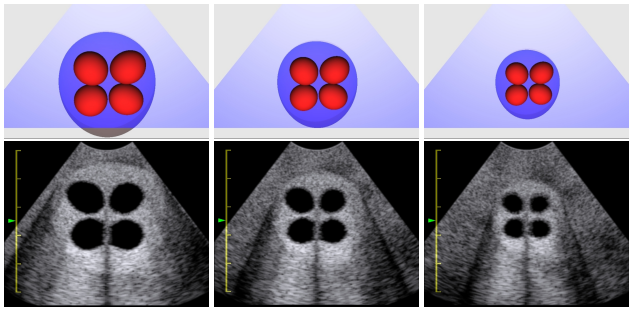


Figure 13: Toy model of an animated heart (top) and simulation (bottom) for 3 contraction phases of a heartbeat. Changing surface boundaries requires animation of both geometry and speckles.

In Fig. 12 we demonstrate the influence of our improved parametrization model. The large-scale variation gives the illusion of complexity in parts of the image where actual geometric detail is missing. The increased visual complexity can best be observed in the upper right part of the image, in the absence of any real geometry. Furthermore, it gives a more detailed appearance to parts of the image where geometry exists but is not sufficiently detailed. This can be seen, for example, when comparing the appearances of the uterus (the bright band in the center of the image) in both images. While speckle variations cannot fully replace the complexity of actual geometry, it nevertheless greatly simplifies the task of a modeler in the face of the overwhelming complexity of an actual patient's interior.

Fig. 13 shows an example of 3 frames of an animation of the heart model which is contracted according to a sine motion. The contraction of the heart also causes the surrounding tissue to move. Hence, to correctly account for a changing ultrasound volume, it is necessary to also animate the speckle appearance, using the same transformation for the coordinates of the 3D noise texture lookups.

6. Discussions and Limitations

Our method is able to generate plausible ultrasound images from a *good* surface model and *suitable* tissue parametrization. However, creating sufficiently complex surface models and finding these tissue parameters are still tedious manual tasks that takes considerable time and substantial fine-tuning. This is a potential bottleneck in a wider application of the proposed method. Possible future work in this direction can leverage image-based techniques for automatically parametrizing individual tissues [MG15], or for automatically improving similarity to a target image in a post-process [SAP*15].

A pending problem in ultrasound image simulation is an *objective* evaluation of our method, in terms of an objective metric that can distinguish realistic from insufficient ultrasound simulations. We focused on visual comparisons of the images, and left an objective evaluation for future work. Nevertheless, we asked an expert sonographer to rate the images in Fig 1 from 1 to 7 in terms of realism (7 is very realistic), and in terms of perceived similarity to the in-vivo image (ignoring the different age of the embryo). The expert rated deterministic ray-tracing as 2 on both accounts, and rated the image generated with our method as 6.5 on both accounts; confirming the high realism and clinical feasibility of our presented image simulation approach. The expert stated that both pregnancy and surroundings are not very realistic for Fig. 1 (b), while in Fig. 1 (c) the pregnancy is very realistic and the surroundings close to realism. This indicates that our method can reproduce the physics of ultrasound imaging better than related work, but an an improved speckle model and/or more detailed geometry are still needed for a truly realistic training simulation.

7. Conclusions

We have presented a novel ultrasound Monte-Carlo ray-tracing method, which significantly improves realism in ultrasound simulation. The method enables the simulation of realistic tissue as present in in-vivo phantoms and as such is essential for plausible ultrasound simulation for training medical students with interactive virtual-reality scenes. In future work, we plan to study more sophisticated Monte-Carlo techniques such as Metropolis sampling [VG97] or bi-directional path tracing.

Acknowledgements. This work was supported by the Swiss Commission for Technology and Innovation (CTI). We thank Maxim Makhynia for proofreading and valuable suggestions.

References

[App68] APPEL A.: Some techniques for shading machine renderings of solids. In *Proceedings of the Spring Joint Computer Conference* (1968), pp. 37–45. 3

[BBRH13] BÜRGER B., BETTINGHAUSEN S., RADLE M., HESSER J.: Real-time GPU-based ultrasound simulation using deformable mesh models. *IEEE Transactions on Medical Imaging 32*, 3 (2013), 609–618. 2, 3, 4, 5, 8

[BD80] BAMBER J. C., DICKINSON R. J.: Ultrasonic B-scanning: a computer simulation. *Physics in Medicine and Biology 25*, 3 (1980), 463. 2, 4

[GCC*09] GAO H., CHOI H., CLAUS P., BOONEN S., JAECQUES S., VAN LENTHE G., VAN DER PERRE G., LAURIKS W., D'HOOGE J.: A fast convolution-based methodology to simulate 2-D/3-D cardiac ultrasound images. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control 56*, 2 (feb 2009), 404–409. 2, 4

[Gla89] GLASSNER A. S. (Ed.): *An Introduction to Ray Tracing*. Academic Press Ltd., London, UK, 1989. 5

[Jen04] JENSEN J.: Simulation of advanced ultrasound systems using field ii. In *Proceedings of ISBI* (2004), IEEE, pp. 636–639. 2

[KVdG*14] KOSKELA J., VAHALA E., DE GREEF M., LAFITTE L. P., RIES M.: Stochastic ray tracing for simulation of high intensity focal ultrasound therapy). *The Journal of the Acoustical Society of America 136*, 3 (2014), 1430–1440. 3

[KWN10] KARAMALIS A., WEIN W., NAVAB N.: Fast ultrasound image simulation using the westervelt equation. In *Proceedings of MICCAI* (2010), pp. 243–250. 2

[Laf95] LAFORTUNE E. P.: *Mathematical Models and Monte Carlo Algorithms for Physically Based Rendering*. PhD thesis, Department of Computer Science, KU Leuven, Celestijnenlaan 200A, 3001 Heverlee, Belgium, February 1995. 3

[MB95] MEUNIER J., BERTRAND M.: Ultrasonic texture motion analysis: theory and simulation. *IEEE Transactions on Medical Imaging 14*, 2 (Jun 1995), 293–300. 2, 4

[MBM87] MEUNIER J., BERTRAND M., MAILLOUX G.: A model for dynamic texture analysis in two-dimensional echocardiograms of the myocardium. *Proceedings of SPIE 0768* (1987), 193–200. 2, 4

[MG15] MATTAUSCH O., GOKSEL O.: Scatterer reconstruction and parametrization of homogeneous tissue for ultrasound image simulation. In *IEEE EMBC* (August 2015). 3, 4, 5, 9

[MG16] MATTAUSCH O., GOKSEL O.: Image-based PSF estimation for ultrasound training simulation. In *Workshop on Simulation and Synthesis in Medical Imaging (SAHSIMI)* (October 2016), Springer. 5

[OTV85] OOSTERVELD B., THIJSSEN J., VERHOEF W.: Texture of B-mode echograms: 3-D simulations and experiments of the effects of diffraction and scatterer density. *Ultrasonic Imaging 7*, 2 (1985), 142–160. 2

[RLK*08] REIS G., LAPPE B., KOHN S., C.WEBER, BERTRAM M., HAGEN H.: Towards a virtual echocardiographic tutoring system. In *Visualization in Medicine and Life Sciences*. 2008, pp. 99–119. 1

[RPAS09] REICHL T., PASSENGER J., ACOSTA O., SALVADO O.: Ultrasound goes GPU: real-time simulation using CUDA. *Proceedings of SPIE 7261* (2009), 726116–726116–10. 2

[SAP*15] SALEHI M., AHMADI S.-A., PREVOST R., NAVAB N., WEIN W.: Patient-specific 3D ultrasound simulation based on convolutional ray-tracing and appearance optimization. In *Proceedings of MICCAI* (2015), pp. 510–518. 2, 5, 9

[Var06] VARSLOT T.: Forward propagation of acoustic pressure pulses in 3d soft biological tissue. *Modeling, Identification and Control* (2006). 2

[VG97] VEACH E., GUIBAS L. J.: Metropolis light transport. In *Proceedings of Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1997), SIGGRAPH '97, ACM Press/Addison-Wesley Publishing Co., pp. 65–76. 3, 9

[VT05] VARSLOT T., TARALDSEN G.: Computer simulation of forward wave propagation in soft tissue. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control 52*, 9 (Sept 2005), 1473–1482. 2

[WBK*08] WEIN W., BRUNKE S., KHAMENE A., CALLSTROM M. R., NAVAB N.: Automatic CT-ultrasound registration for diagnostic imaging and image-guided intervention. *Medical Image Analysis 12*, 5 (2008), 577–585. 2

[Whi80] WHITTED T.: An improved illumination model for shaded display. *Communications ACM 23*, 6 (June 1980), 343–349. 3