3D freehand ultrasound reconstruction using a piecewise smooth Markov random field

Hyungil Moon[a], Geonhwan Ju[a], Seyoun Park[b], Hayong Shin[a,*]

[a] Department of Industrial and Systems Engineering, KAIST, Daejeon, Republic of Korea
[b] Department of Radiation Oncology and Molecular Radiation Sciences, Johns Hopkins School of Medicine, Baltimore, USA

ARTICLE INFO

ABSTRACT

In this paper, we introduce a novel three-dimensional (3D) reconstruction framework for ultrasound images using a piecewise smooth Markov random field (MRF) model from irregularly spaced B-scan images obtained by freehand scanning. Freehand 3D ultrasound imaging is a useful system for various clinical applications, including image-guided surgeries and interventions, as well as diagnoses, due to the variety of its scan ranges and relatively low cost. The reconstruction process performs a key role in this system because its sampling irregularities may cause undesired artifacts, and ultrasound images generally suffer from noise and distortions. However, traditional approaches are based on simple geometric interpolations, such as pixel-based or distance-weighted methods, which are sensitive to sampling density and speckle noise. These approaches generally have an additional limitation of smoothing objects boundaries. To reduce speckle noise and preserve boundaries, we devised a piecewise smooth (PS) MRF model and developed its optimization algorithm. In our framework, we can easily apply an individual noise level for each image pixel, which is specified by the characteristics of an ultrasound probe, and possibly, the lateral and axial positions of an image. As a result, the reconstructed volume has sharp object boundaries with reduced speckle noise and artifacts. Our PS-MRF model provides simple segmentation results within a reconstruction framework that is useful for various purposes, such as clear visualization. The corresponding optimization methods have also been developed, and we tested a virtual phantom and a physical phantom model. Experimental results show that our method outperforms existing methods in terms of interpolation and segmentation accuracy. With this method, all computations can be performed with practical time consumption and with an appropriate resolution, via parallel computing using graphic processing units.

1. Introduction

Ultrasound 3D imaging has received significant attention in many diagnostic areas, particularly obstetrics [1] and cardiology [2]. Not only 3D visualizations, such as in volume and surface rendering, but also 2D sectional images at various orientations, may provide helpful clues for diagnoses. In addition, many studies have demonstrated the applicability of 3D ultrasound imaging to image-guided surgery and interventions, e.g., neurosurgery [3], biopsy [4], and radiation therapy [5].

There are two types of probes that can generate 3D ultrasound data: dedicated 3D probes and conventional 2D probes with mechanical or freehand scanning. Dedicated 3D probes can scan the 3D range rapidly and can generate volumetric images quickly.

However, systems with dedicated 3D probes are more expensive, and the large contact surface of the probe makes it difficult to obtain clean images of hidden structures under bones or gas. These probes are larger and heavier than 2D probes, and their scanning ranges are limited by their size.

A conventional probe generates 2D images initially, but 3D data can be obtained through sweeping the probe through the target volume while acquiring the position and orientation information from a position sensor. Therefore, the scanning ranges have a greater variety than those of dedicated 3D probes. Sweeping is conducted using a mechanical device or by hand. Mechanical sweeping generates regularly spaced B-scan images along a predefined path, while freehand sweeping generates irregularly spaced B-scan images along an arbitrary path. Mechanical devices can sweep or rotate the volume with a uniform speed, but it is impossible to scan larger volumes than the mechanics allows. This limitation cannot be overcome through simply increasing the size of the system [6]. Compared with other scanning methods,

* Corresponding author. Fax: +82 42 350 3110.
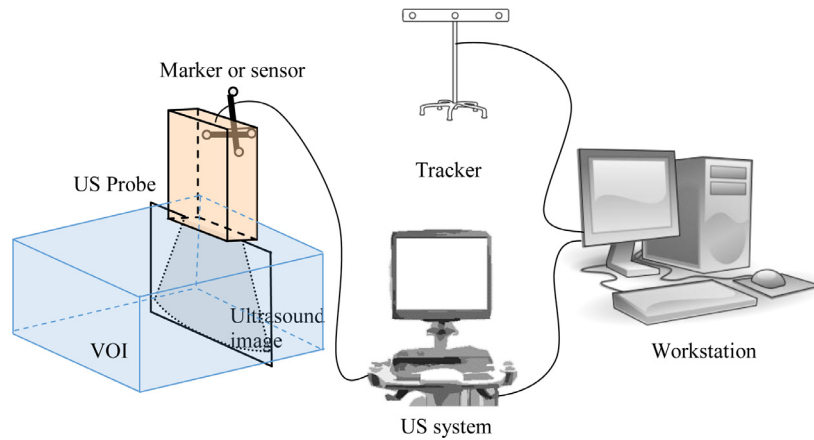 E-mail address: hyshin@kaist.ac.kr (H. Shin).

Fig. 1. Conceptual diagram of the freehand 3D ultrasound imaging setup [5].

freehand ultrasound imaging has more freedom in terms of scanning range, and various normal 2D probes can be used directly. These advantages are useful for various applications. Fig. 1 shows a conceptual diagram of the freehand scanning system.

In freehand scanning, because B-scan images are captured at arbitrary locations and orientations, it is not guaranteed that the physical image pixel locations will match the voxel positions. Therefore, it is necessary to fill in missing voxel values in the volume of interest (VOI) from the scanned images. To obtain a better reconstruction result, noise and artifacts should be considered while interpolating the voxel values.

However, many previous studies have focused on the calibration of the position sensors to accurately convert the image coordinates to global coordinates [7]. Compared with the acquisition process, the reconstruction process has been considered simply in most works despite its importance for the quality of the final images of a volume.

There are numerous considerations in the reconstruction step to improve the volume quality. First, ultrasound images have various types of noise and artifacts, such as speckle noise, refraction, shadowing, reverberation, and so on. Most artifacts originate from the interaction between the ultrasound signal and inside materials, i.e. they represent different patterns according to the inner materials. High-level processing is usually required in order to reduce them. Meanwhile, speckle noise is typically distributed throughout B-scan images. Speckle noise also shows unique patterns according to the materials, but it is distributed all over images with high frequency compared to other artifacts (see Fig. 2(a)). This results in

degradation of the image quality and makes it difficult for viewers to interpret these images and make diagnoses. Therefore, speckle noise reduction is an important issue in ultrasound image analysis, and it has been investigated in previous works as post-processing [8]. However, it is not practically efficient to filter every scanned image in case of this study.

Another important characteristic of ultrasound images that should be considered is that the spatial resolution is not uniform within an image due to the transducer and signal characteristics. Fig. 2(c) presents scan result of a phantom for quality assurance, and the image quality varies with the penetration depth as well as the lateral direction. Therefore, B-scan image pixel data cannot be considered to have the same confidence level over all pixels, and the data confidence should be considered according to pixel location.

However, most existing reconstruction methods involve simple averaging or interpolation, i.e. pixel nearest neighbor (PNN), voxel nearest neighbor (VNN), distance weighted (DW), and radial basis function (RBF) methods [10,11]. These approaches interpolate the voxel intensities from the sampled data by simply taking the nearest neighbors (PNN and VNN), which are sensitive to the sampling density, or weighted averages (DW and RBF). To improve the quality, various median filters are introduced [12]. In addition, Wen et al. introduced a fast marching method to fill holes in order to overcome the limitations of the nearest-neighbor selection [13]. Instead of spatial interpolation of the voxels or pixels, another approach for the 3D reconstruction is to interpolate probe trajectories to create intermediate virtual scanning planes [14].
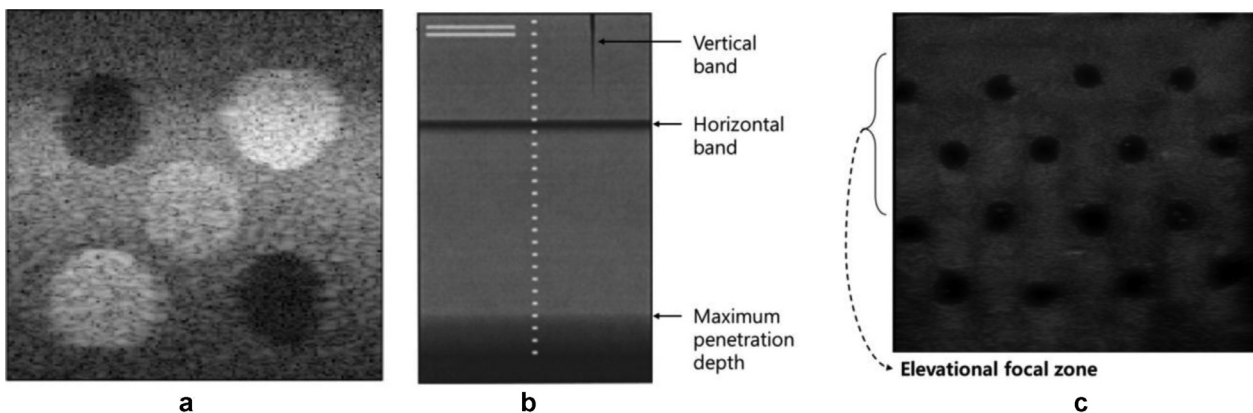

Fig. 2. Noise and artifacts: (a) speckle noise, (b) transducer malfunction [9], and (c) elevational focal zone.

Another important approach uses a Bayesian framework to infer the voxel values in a grid [15,16]. This approach assumes a three-dimensional parametric function that has basis functions centered at every voxel. Each basis function is determined by a corresponding coefficient, and all coefficients of the voxel grid are modeled as a 3D grid Markov random field (MRF) with the typical *6-connected neighborhood* system [15]. In this approach, the observations are assumed to be Rayleigh distributed random variables to represent speckle noises.

However, one significant drawback of these methods is that they may not preserve object boundaries. Boundaries are easily smoothed out using these models because they assume uniform smoothness. To resolve this problem, we introduce a novel framework for reconstructing 3D ultrasound images from freehand scanning. The primary goals of this research are the following:

(1) *Speckle noise reduction*: Speckle noise results in images that are noisy in ways such that the interpolated values could be severely affected. Our first goal in this research is to reduce the influence of speckle noise.

(2) *Boundary conservation*: It can be helpful in many clinical applications if surface boundaries are clearly observable in ultrasound images. However, most state-of-the-art reconstruction algorithms smoothly interpolate the sampled data points.

(3) *Noise level* (*data confidence*) *consideration*: Within one B-scan image, the spatial resolution and image quality vary with the location. If several data are observed in close physical locations, e.g., the same voxel, their confidence level should be given significant consideration, rather than simply considering distances.

(4) *Computation time efficiency*: In many clinical applications, fast computation is required for practical use. In particular, the algorithm needs to be insensitive to the amount of data sampled from a wide scan range for the freehand system.

To achieve the above goals, we adopt a Bayesian framework to infer the voxel intensities from the observed data as in [15]. Through modeling the voxel grids as an MRF and applying an observation model that considers the noise distribution, we can alleviate the effects of speckle noise in the 3D reconstruction process. In addition, we use a piecewise smooth MRF for boundary conservation through considering the ultrasound imaging characteristics. MRF reconstruction models can be categorized into three types: smooth models, piecewise constant (PC) models, and piecewise smooth (PS) models [17]. The smooth models do not consider signal discontinuities, so object boundaries are easily blurred. The PC models can preserve the discontinuities, but they separate the gradual changes of signals into several different regions. One of the primary characteristics of ultrasound imaging is that the signal intensities for the same tissue may not be uniform due to attenuations or shadow effects. Therefore, the PC models also are not appropriate for ultrasound reconstruction. The PS models can represent regions with gradual changes and separate different regions, only with large jumps. For this reason, we propose a PS model that conserves the boundary information of different regions. Within our framework, the noise levels can be simply applied without changing or adding terms. In addition, the computation to get the optimal solution is accelerated by parallelization on graphic processing units (GPUs).

This paper is organized as follows. In Section 2, an overview of the Bayesian formulation of our problem is introduced, and several types of prior models and corresponding optimization methods are described. In Section 3, experiments with synthetic US images and real images are described. In Section 4, we discuss the advantages and limitations of the proposed method and suggest directions for future work.

2. Methods

2.1. Overview of MRF models

The objective of reconstruction in this paper is to determine the most representative intensity value at each voxel of regularly spaced grids in a volume of interest (VOI) from an observed dataset. Let us denote a vector of the voxel values of the VOI by $\mathbf{f} = (f_1, \ldots, f_m)^T$, where m is the number of voxels. We propose some probabilistic inferences of $\mathbf{f}$ in the Bayesian framework. In this paper, the likelihood and prior distributions are defined, and the corresponding optimization methods that maximize the posterior probability distribution functions (PDFs) are discussed. According to the Bayes rule, a posterior PDF $p(\mathbf{f}|\mathcal{D})$ is proportional to a prior PDF $p(\mathbf{f})$ and the likelihood function $p(\mathcal{D}|\mathbf{f})$, i.e. $p(\mathbf{f}|\mathcal{D}) \propto p(\mathbf{f}) p(\mathcal{D}|\mathbf{f})$.

A dataset ($\mathcal{D}$) contains triplets of an observed pixel value (y_p), a world coordinate position vector ($\mathbf{x}_p$), and a noise variance (σ_p^2) for all pixels in all B-scan images: $\mathcal{D} = \{(y_p, \mathbf{x}_p, \sigma_p^2)|p = 1, \ldots, n\}$, where n is the number of pixels in the B-scan images. The most common method to obtain the world coordinate is to use an optical tracker. The optical tracking method used in this research is explained in Section 3.2.1. The variance σ_p^2 can be determined from the user input or by pre-processing the images, or any other characteristics of a probe. As depicted in Fig. 2(c), a general ultrasound image has a focal zone determined by the characteristics of the ultrasound signal and probes. As a result, the spatial resolution varies depending on the location of a pixel in an image, and the data accuracy also varies according to pixel location. If there are two pixel points from two different image planes and they have similar distances from a voxel, then the point in the focal region is more likely to represent the original intensity. To enhance the quality of 3D reconstruction, this type of data precision information as well as the geometric distance must be applied. In our framework, this can easily be applied to the model by changing the observation variance σ_p^2 for each p.

The observation model defines the likelihood of the observed dataset $\mathcal{D}$ with given $\mathbf{f}$. All observed pixel values are assumed to be independent normal random variables; therefore, the likelihood function is

$$p(\mathcal{D}|\mathbf{f}) = \prod_{p=1}^{n} \phi\left(y_p; \mu_p, \sigma_p^2\right), \tag{1}$$

where $\phi(y_p; \mu_p, \sigma_p^2)$ is a normal PDF with a mean μ_p and a variance σ_p^2. In this paper, the mean μ_p is assumed to be $\mu_p = f_{v(\mathbf{x}_p)}$, where $v(\mathbf{x}_p)$ is the nearest voxel index from $\mathbf{x}_p$. This mapping, also referred to as *pixel nearest neighbor mapping* [11], simplifies the model and calculations. This observation model implies that y_p only depends on $v(\mathbf{x}_p)$. That is, a conditional independence is assumed such that $p(y_p|\mathbf{f}) = p(y_p|f_{v(\mathbf{x}_p)})$ for any pixel index p.

The prior distribution is from a priori knowledge about $\mathbf{f}$. The prior distribution contains information about what values of $\mathbf{f}$ are preferred and how much they are preferred. This paper constructs the prior distribution using an MRF framework. An MRF is a set of random variables that has a certain conditional independence structure that is graphically described using nodes and links [18]. Throughout this paper, a node represents a voxel value or an observed pixel value, and a link represents a similarity relationship between two nodes.

A basic example of an MRF model of a voxel grid is the *6-connected neighborhood model*, which is referred to as the *uniformly smooth model* in this paper, or the smooth model. The model has all possible links along the x, y, and z directions in the voxel grid. Furthermore, all observations are connected to the corresponding
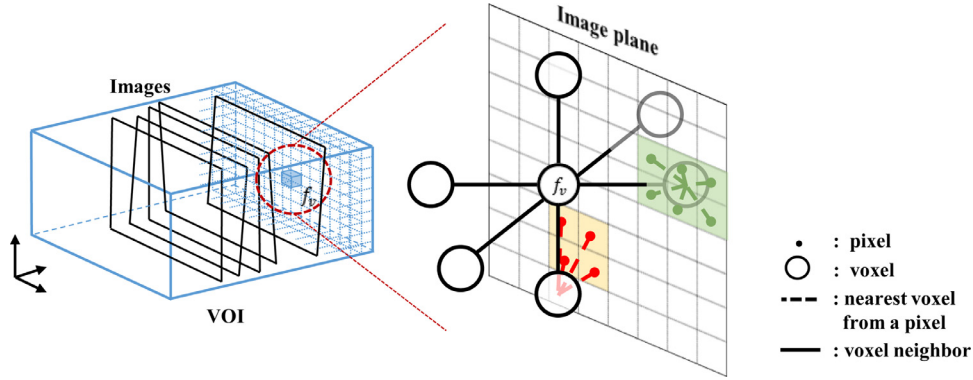
Fig. 3. Graphic representation of the smooth MRF model.

nearest voxels due to the assumed observation model. A graphic representation of the smooth model is presented in Fig. 3.

To model the discontinuities at some locations in the voxel grid, it must be determined whether each similarity relationship is turned on or off. Because it is difficult to optimize an energy function of a discontinuity preserving model, various methods are proposed for each discontinuity preserving model. Sections 2.2 and 2.3 describe the mathematical formulations and optimization methods of the smooth model and of the discontinuity preserving models, respectively.

2.2. Uniformly smooth model

A prior probability distribution function of the smooth model is

$$p(\mathbf{f}) \propto \exp\left(-\frac{\psi}{2}\left(\sum_{(v,v')\in\mathcal{E}}(f_v - f_{v'})^2\right)\right), \tag{2}$$

where ψ is the smoothness strength coefficient, $\mathcal{E}$ is the set of all links between voxels in the *6-connected neighborhood model*, and (v, v') is the link between two voxels, v and v'. Note that the MRF is an undirected graph, i.e. (v, v') and (v', v) indicate the same link. The posterior PDF of the smooth model is

$$p(\mathbf{f}|\mathcal{D}) \propto p(\mathcal{D}|\mathbf{f})p(\mathbf{f})$$

$$\propto \exp\left(-\frac{1}{2}\sum_{p=1}^{n}\tau_p\big(f_{v(p)} - y_p\big)^2\right)$$

$$\exp\left(-\frac{\psi}{2}\left(\sum_{(v,v')\in\mathcal{E}}(f_v - f_{v'})^2\right)\right)$$

$$\propto \exp\left(-\frac{1}{2}\left(\sum_{p=1}^{n}\tau_p\big(f_{v(p)} - y_p\big)^2 + \psi\sum_{(v,v')\in\mathcal{E}}(f_v - f_{v'})^2\right)\right), \tag{3}$$

where $\tau_p = 1/\sigma_p^2$, which indicates the precision of an observation. This notation is used in the remaining sections.

The voxel-based representation of the likelihood terms is more convenient for later calculations and result expressions. For any voxel index v, a set $\mathcal{P}_v$ is a collection of pixel indices, in which the nearest voxel is v: $\mathcal{P}_v = \{p|v(p) = v, \ p = 1, \ldots, n\}$; then, all pixel indices from 1 to n are classified into mutually disjoint sets $\mathcal{P}_1$ to $\mathcal{P}_m$. Thus, Eq. (3) can be rewritten as

$$p(\mathbf{f}|\mathcal{D}) \propto \exp\left(-\frac{1}{2}\left(\sum_{v=1}^{m}\sum_{p\in\mathcal{P}_v}\tau_p(f_v - y_p)^2 + \psi\sum_{(v,v')\in\mathcal{E}}(f_v - f_{v'})^2\right)\right). \tag{4}$$

By the Hammersley-Clifford theorem [19], the joint distribution of an MRF is a Gibbs distribution, which is represented as $p(\mathbf{f}|\mathcal{D}) = \frac{1}{Z}\exp\{-E(\mathbf{f},\mathcal{D})\}$, where Z is a normalizing constant and $E(\mathbf{f},\mathcal{D})$ is an energy function. Therefore, maximizing a posteriori of an MRF can be handled more easily by minimizing the corresponding energy function because there is no exponential term, and the normalizing constant Z can be ignored. In this paper, the Gibbs distribution is expressed as $p(\mathbf{f}|\mathcal{D}) = \frac{1}{Z}\exp\{-\frac{1}{2}E(\mathbf{f},\mathcal{D})\}$ in order to eliminate the common term $1/2$ in the energy function. Therefore, the energy function of the smooth model is

$$E(\mathbf{f},\mathcal{D}) = \sum_{v=1}^{m}\sum_{p\in\mathcal{P}_v}\tau_p(f_v - y_p)^2 + \psi\sum_{(v,v')\in\mathcal{E}}(f_v - f_{v'})^2. \tag{5}$$

Then, $E(\mathbf{f},\mathcal{D})$ becomes the function in the quadratic form of the decision variables f_v and measured data y_p, which are from the data likelihood and the prior distribution function. Smaller differences of neighboring variables and measured data induce a smaller energy, and this results in a higher posterior density. The optimal solution can be obtained by solving $\nabla E = 0$:

$$\left\{ \text{diag}\left(\begin{bmatrix} \sum_{p\in\mathcal{P}_1}\tau_p \\ \vdots \\ \sum_{p\in\mathcal{P}_m}\tau_p \end{bmatrix}\right) + \psi \mathsf{A} \right\} \mathbf{f} = \begin{bmatrix} \sum_{p\in\mathcal{P}_1}\tau_p y_p \\ \vdots \\ \sum_{p\in\mathcal{P}_m}\tau_p y_p \end{bmatrix}, \tag{6}$$

where A is the $m \times m$ matrix in which $\mathsf{A}(i, i)$ is the number of links connected to the voxel i, and $\mathsf{A}(i, j) = -1$ if $(i, j) \in \mathcal{E}$ or $(j, i) \in \mathcal{E}$. All other elements of A are zeroes. The conjugate gradient method is a common method for solving this type of system of linear equations [20].

With the solution of Eq. (6), the vth row of (6) is

$$\left(\sum_{p\in\mathcal{P}_v}\tau_p + \psi|\mathcal{N}_v|\right)f_v^* - \psi\sum_{v'\in\mathcal{N}_v}f_{v'}^* = \sum_{p\in\mathcal{P}_v}\tau_p y_p, \tag{7}$$

where $\mathcal{N}_v$ is a set of voxels connected to the vth voxel, $|\mathcal{N}_v|$ is the cardinality of the set, and f_v^* is the vth element of the solution vector $\mathbf{f}^*$. Then, Eq. (7) can be rearranged as

$$f_v^* = \frac{\sum_{p\in\mathcal{P}_v}\tau_p y_p + \psi\sum_{v'\in\mathcal{N}_v}f_{v'}^*}{\sum_{p\in\mathcal{P}_v}\tau_p + \psi|\mathcal{N}_v|}. \tag{8}$$

Eq. (8) describes that each optimal value f_v^* is the weighted average of the observation pixels and neighbor voxel values with weights τ_p and ψ for a pixel p in $\mathcal{P}_v$ and neighbor voxels, respectively.
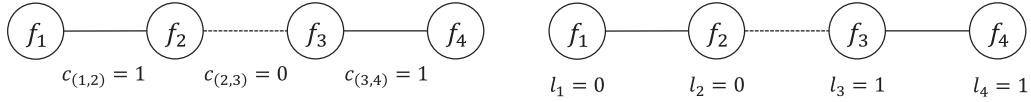
Fig. 4. Conceptual differences between the edge-based model (left) and region-based model (right).

2.3. Piecewise smooth models

Basically, there are two possible formulations of the PS models: edge-based and region-based labeling formulations. In the edge-based model, the formulation has an additional term $c_{(v,v')} \in \{0, 1\}$ for each edge (v, v'), which determines whether the link is connected or not. In the region-based formulation, every voxel has a label $l_v \in \mathbb{L}$, where $\mathbb{L}$ is the set of possible labels. A link is only connected if two neighbor voxels have the same label. Graphic representations of the two formulations are presented in Fig. 4. There are some variants that combine multiple models or methods for the 2D image segmentation including a combined method of the edge-based and region-based models [21] and an MRF of distinct regions with an edge detector considering speckles [22]. Those methods have more considerations about the boundaries. However, it is hard to apply directly to 3D reconstruction with sparse data. More context-aware methods are discussed in Section 4.

Of the two approaches, we adopted the region-based model for three reasons. First, the edge-based model easily produces undesirable short and separated boundaries [23]. It requires additional complex energy terms to prevent undesirable results, and the corresponding optimization becomes much more difficult. If the pixel-by-pixel signal jumps in ultrasound images are considered, it is difficult to determine whether the signal jumps result from speckle noise or underlying materials. Thus, it is more natural to consider the regional properties than the edge properties. Another advantage of the region-based model is that the label values of the region-based method provide a simple segmentation result without additional processing, which is useful for a variety of purposes [24]. Finally, a significantly more robust and efficient optimization method exists for the region-based model than the edge-based model.

From a technical perspective, it is not trivial to compute the optimal solution of most piecewise smooth models because their energy functions have many local optima in general. However, the label-based formulation can be converted to a graph-cut problem in a binary case, whose global solution can be found efficiently using the max-flow/min-cut algorithm [25]. Therefore, we formulated the proposed MRF model as a binary piecewise smooth model to represent the foreground and background in this research. The multi-label cases are discussed further in Section 4.

The energy function of the basic region-based formulation is

$$E(\mathbf{f}, \mathcal{L}, \mathcal{D}) = \sum_{v=1}^{m} \sum_{p \in \mathcal{P}_v} \tau_p (f_v - y_p)^2 + \psi \sum_{(v,v') \in \mathcal{E}} \left\{ \delta_{l_v, l_{v'}} (f_v - f_{v'})^2 \right.$$
$$\left. + \left(1 - \delta_{l_v, l_{v'}}\right)\alpha^2 \right\} .$$

(9)

where $\mathcal{L} = \{l_v | v = 1, \ldots, m\}, \delta_{l_v, l_{v'}}$ is the Kronecker delta function which returns 1 when $l_v = l_{v'}$ or 0 if $l_v \neq l_{v'}$, and α is a parameter related to the level of signal jump. If the difference of two neighboring values f_v and $f_{v'}$ is more than α, then the energy between the two voxels is lowered with $\delta_{l_v, l_{v'}} = 0$, which turns off the smoothness penalty. As explained before, we restricted our attention to the binary labels case $l_v \in \mathbb{L} = \{0, 1\}$ in this study. As seen in Eq. (9), the smoothing term between neighbor voxels is only applied when they belong to the same region. For abdominal imaging, soft tissue might belong to the background, and organs or other structures might be foreground objects. The initial separation can be performed via pre-processing using an optimal global thresholding method, such as Otsu's thresholding [26].

2.4. Optimization

To obtain an optimal solution for Eq. (9) efficiently, we can convert it to an alternative two-layered formulation, as discussed in [27]. The alternative formulation assumes that two separate layers exist: $\mathbf{g}^0 = (g_1^0, \ldots, g_m^0)^T$ and $\mathbf{g}^1 = (g_1^1, \ldots, g_m^1)^T$. A label l_v is used to select one of the two signal values as the reconstructed value of the vth voxel: $f_v = (1 - l_v)g_v^0 + l_v g_v^1$, and a set of pixels $\mathcal{P}_v$ is connected to the selected layer. Fig. 5 is a graphical representation of the alternative formulation.

The alternative formulation is

$$E\left(\mathbf{g}^0, \mathbf{g}^1, \mathcal{L}, \mathcal{D}\right)$$
$$= \sum_{h=0}^{1} \sum_{v=1}^{m} \delta_{l_v, h} \left(\sum_{p \in \mathcal{P}_v} \tau_p \left(g_v^h - y_p\right)^2 + \frac{\psi}{2} \sum_{v' \in \mathcal{N}_v} \left(g_v^h - g_{v'}^h\right)^2 \right)$$
$$+ \psi \alpha^2 \sum_{(v,v') \in \mathcal{E}} \left(1 - \delta_{l_v, l_{v'}}\right) .$$

(10)

The energy function consists of the data fitness, smoothness, and the discontinuity terms. Note that the data fitness and
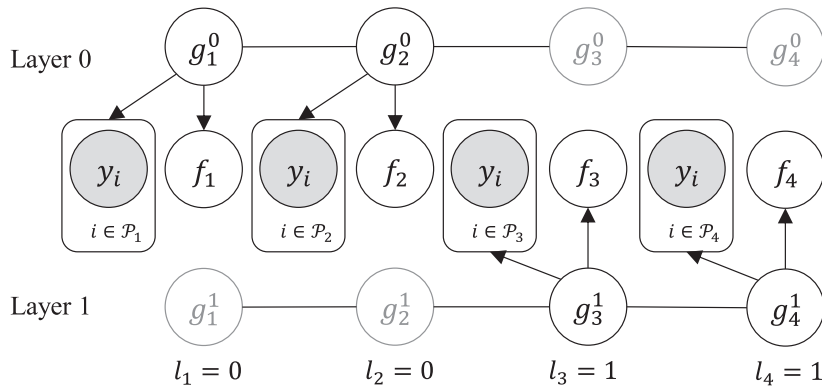

Fig. 5. Graphical representation of the alternative formulation.

smoothness terms of the layers are turned on or off by the labels. In addition, energy terms exist if any two neighbor voxels have different labels. As explained in Section 2.2, $\mathcal{E}$ is a set of all links in the *6-connected neighborhood model*. However, an MRF with the neighborhood model does not have any preference whether boundaries are blocky or not when data is sparse [28]. In freehand ultrasound scanning, there may be a significant number of voxels that are unobserved, i.e. $\mathcal{P}_v = \emptyset$, and it may cause undesired "*blockiness*" artifacts during the reconstruction process. Therefore, additional energy terms for boundary regularization are required in order to avoid this artifact. One simple solution in our framework is to expand the neighborhood system to include more neighbors in various directions [28]. For example, in 2D MRFs, the *8-connected neighborhood model*, which connects all combinations of x and y axis displacements, produces more smooth boundaries than the *4-connected neighborhood model*. Likewise, the *26-connected neighborhood model* in 3D MRFs, which connects -1, 0, $+1$ combinations for all x, y, and z directions except itself, produces more smooth boundaries than the *6-connected neighborhood model*. Note that the expanded neighborhood model is useful for regularizing the boundary and not for interpolation properties. Therefore, we simply expand the neighborhood model only for the label difference terms and scale the terms to the original *6-connected neighborhood model* to preserve the meaning of parameters. The final energy function is

$$
E\big(\mathbf{g}^0, \mathbf{g}^1, \mathcal{L}, \mathcal{D}\big)
$$

$$
= \sum_{h=0}^{1} \sum_{v=1}^{m} \delta_{l_v,h} \left(\sum_{p \in \mathcal{P}_v} \tau_p \big(g_v^h - y_p\big)^2 + \frac{\psi}{2} \sum_{v' \in \mathcal{N}_v} \big(g_v^h - g_{v'}^h\big)^2 \right)
$$

$$
+ \frac{6}{26} \psi \alpha^2 \sum_{(v,v') \in \mathcal{E}^{26}} \big(1 - \delta_{l_v, l_{v'}}\big), \tag{11}
$$

where $\mathcal{E}^{26}$ represents a set of all links in the 26-*connected neighborhood model*.

Algorithm 1 is a simple optimization algorithm for the energy function in Eq. (10), which was proposed by Grady and Alvino [27], and it can be applied to minimize the extended energy function in Eq. (11). We also introduce an improved algorithm after explaining Algorithm 1. Algorithm 1 iterates some interpolation steps and segmentation steps. In practice, a few iterations are sufficient to converge, and the result of the method is robust regarding variations in the initial label setting [27]. These properties contrast with the properties of traditional contour evolution methods to optimize the Mumford-Shah functional while maintaining closed boundaries. Those benefits result from a graph-cut method of segmentation steps.

With the given $\mathcal{L}'$, the energy function in Eq. (11) becomes a quadratic form, so each of $\mathbf{g}^0$ and $\mathbf{g}^1$ can be minimized as in the smooth model using the conjugate gradient method. However, some elements of the layers do not affect the energy, so they cannot be determined; for a voxel v, if the voxel and the neighbors of the voxel do not belong to a label h, i.e. $\delta_{l_v,h} = 0$ and $\delta_{l_{v'},h} = 0$

for all $v' \in \mathcal{N}_v$, then the intensity value g_v^h does not incur any of the data-fitness penalty or smoothness penalty. Hence, the intensity value g_v^h is independent from the energy. However, all of the values in $\mathbf{g}^0$ and $\mathbf{g}^1$ should be determined, because the values are used in the next label optimization, which will be solved by the graph-cut algorithm. To determine the energy-independent values, the values can be interpolated from the already determined values, which affect the energy. For each layer, a uniformly smooth model can be constructed with the fixed determined values of the layer [27]. Then, solving the uniformly smooth model produces interpolated values for the undetermined values. This interpolation is the same as solving a discrete Laplace equation with fixed boundaries. See [27] for more details.

After calculating $\mathbf{g}^0$ and $\mathbf{g}^1$, each layer functions as preferred values of the corresponding label. A label set $\mathcal{L}$ should be selected that minimizes the energy function in Eq. (11), which is affected by the data-fitness terms and discontinuity penalty terms. The graph-cut method, which has been used successfully for many computer vision problems, can find the exact minimum label set. The graph-cut method has two stages. First, it transforms an energy minimization problem into a max-flow/min-cut problem on a graph [29], and then the transformed problem can be solved efficiently using graph algorithms [30,31].

The most significant advantages of Algorithm 1 are that the method is not prone to become stuck in local minima, and the time consumption of the algorithm is not sensitive to the initial label set $\mathcal{L}^{initial}$ [27]. In our experiments, we set $\mathcal{L}^{initial}$ for the outer voxels of the voxel grid to 0 and the other voxels to 1. This type of $\mathcal{L}^{initial}$ presumes that there might be some objects surrounded by background materials. The initial boundaries between label-0 voxels and label-1 voxels will be squeezed or expanded by the algorithm to reduce the boundary penalties in accordance with the observed data. If it can be certain that the outer voxels are background voxels, then the outer labels can be fixed in the segmentation steps. After this, the boundaries cannot be expanded from the initial label set, and only squeezing is possible.

We improved the algorithm in two ways. Experiments showed that the values of the label set $\mathcal{L}$ do not change significantly from $\mathcal{L}'$ in most iterations. This indicates that the max-flow/min-cut problem for $\mathcal{L}$ is also similar to that of $\mathcal{L}'$. Therefore, the flow information at the max-flow/min-cut problem could be reused in the next problem to reduce computation time [32]. Next, we applied a hierarchical approach using multi-level grids. Algorithm 1 sometimes requires a large number of iterations to converge depending on the problems, and each iteration on a larger grid consumes a considerable amount of time. In contrast, the iterations on a small grid are much faster, because the information can propagate over a longer distance per iteration. Therefore, we can calculate an approximate result on a coarse grid and reuse the approximate result as the initial data on a fine grid. This hierarchical approach can reduce the computation time significantly, particularly for large grid problems. In addition, the reconstruction of the coarse grid is robust to speckle noise because the noise is averaged out in the coarse grid. In this work, we used two levels, but more levels could be applied when the grid size is large. The procedure of the two-level grid method is described in Algorithm 2.

Algorithm 2 has an additional parameter ψ_c, but it can be calculated from ψ and the grid configurations. Suppose that the voxel interval of the coarse grid is t-times larger than that of the fine

Algorithm 1
Optimizing procedure of the Grady and Alvino [27] method.

Function OptimizeEnergy($\mathcal{L}^{initial}$, $\mathcal{D}$, ψ, α, *voxel grid configuration*)
Output: $(\mathcal{L}, \mathbf{g}^0, \mathbf{g}^1)$
exit ← false
$\mathcal{L}' \leftarrow \mathcal{L}^{initial}$
repeat
    Determine $\mathbf{g}^0$ and $\mathbf{g}^1$ by solving systems of linear equations with fixed $\mathcal{L}'$
    Find $\mathcal{L}$ using a graph-cut algorithm with fixed $\mathbf{g}^0$ and $\mathbf{g}^1$
    if $\mathcal{L} = \mathcal{L}'$, **then** *exit ← true*
    $\mathcal{L}' \leftarrow \mathcal{L}$
until *exit = true*

Algorithm 2
Two-level grid optimizing procedure.

$(\mathcal{L}_c, \mathbf{g}_c^0, \mathbf{g}_c^1) \leftarrow$ OptimizeEnergy($\mathcal{L}_c^{initial}$, $\mathcal{D}$, ψ_c, α, *coarse-grid*)
$\mathcal{L}^{initial} \leftarrow$ nearest-neighbor interpolate $\mathcal{L}_c$ and sample at *fine-grid*
$(\mathcal{L}, \mathbf{g}^0, \mathbf{g}^1) \leftarrow$ OptimizeEnergy($\mathcal{L}^{initial}$, $\mathcal{D}$, ψ, α, *fine-grid*)

---

grid, then $t \times t \times t$ voxels in the fine grid become one voxel in the coarse grid. A set of $t \times t \times t$ voxels is in contact with other sets along the x, y, and z directions. Two neighbor sets have $t \times t$ smoothness energy of ψ, so ψ_c should be $t^2\psi$.

To implement the algorithms, we used the *GridCut* software library as a graph-cut solver [30,33–35]. In addition, we applied a push-relabel algorithm in the Compute Unified Device Architecture (CUDA) of NVidia [36] to accelerate the computing times using GPUs [37]. The push-relabel algorithm traverses all voxels while pushing excess flow or relabeling for each voxel, and all operations can be parallelized. Our implementation allocates one thread per voxel logically and conducts pushing or relabeling.

3. Results

This section compares the results obtained by the MRF reconstruction methods and other conventional methods. Herein, we denote the smooth model as US-MRF and the modified formulation in Eq. (11) of the PS model as PS-MRF. For comparison, the PNN and DW methods were selected because they exhibit good reconstruction quality among the existing methods within a reasonable computation time [10,11]. The PNN and DW methods are implemented based on pseudo-codes [11]. The PNN that was implemented here locates the nearest neighbor voxel of each pixel for an initial bin-filling step, and the remaining empty voxels are filled with a Gaussian weighted average value of local-neighbor bin-filled voxels within a radius R. The DW method traverses all pixels of the input images while adding a weighted pixel value to voxels within the radius R, where the weight is the inverse of the distance between the pixel and voxel [38].The radius R was set to 1.5 mm for all experiments. Unfilled voxels were set to the average value of all observation pixel values.

3.1. Simulated data

Simulated ultrasound images of a virtual phantom were generated using the *FieldII* software library [39,40]. There were two ball-shaped volumes, one of which had a higher density volume, and the other had a lower density volume than the background. The phantom size was approximately $40 \times 26 \times 60$ mm^3, and 50 images were generated from the virtual probe directions sweeping y-direction with randomness, while the images were kept parallel to the x-axis. A voxel grid was generated for the virtual phantom with $161 \times 104 \times 232$ voxels with 0.25 mm spacing. Fig. 6 presents the virtual model, a simulated image, and a yz-sectional plane of the voxel grid where the image planes are represented by thick lines.

To evaluate the 3D reconstruction performance of various methods, the reconstruction errors, interpolation errors, segmentation errors, and computation times were measured. As mentioned in Sections 1 and 2, the objective of this paper is to reconstruct the representative voxel values while preserving object boundaries. Hence, the reconstruction methods should minimize the *reconstruction error*,

$$reconstruction\ error = \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \left| f_v - f_v^{true} \right|, \tag{12}$$

where f_v is the value restored by each method, $|\mathcal{V}|$ is the cardinality of the set $\mathcal{V}$, and f_v^{true} is the true representative value of the voxel v. To get the ground truth of f_v^{true}, we simulated 5000 images with random probe positions and directions, as shown in Fig. 6(c), and conducted the nearest-voxel mapping and averaging of all pixels.

Interpolation error measures the restoration performance by removing a scanned image from the input dataset proposed in [10]. Although our reconstruction purpose is not to generate exactly the same data with the original ultrasound image including noises and artifacts, this evaluation assessed the capability of the reconstruction algorithm to fill in missing data regions. Let $\mathcal{V}_{removed}$ denote a set of the voxel indices that lose data. The *interpolation error* is the average difference between the restored voxel values and the bin-filled voxel values of the removed image:

$$interpolation\ error = \frac{1}{|\mathcal{V}_{removed}|} \sum_{v \in \mathcal{V}_{removed}} \left| f_v - \frac{1}{|\mathcal{P}_v|} \sum_{p \in \mathcal{P}_v} y_p \right|, \tag{12a}$$

where $|\mathcal{V}_{removed}|$ and $|\mathcal{P}_v|$ are the cardinalities of the corresponding sets. In our experiments, we removed each image plane in the dataset and calculated the interpolation error for the image.

The *segmentation error* is the proportion of misclassified voxels. The true label of a voxel is defined as *foreground* if the voxel is in the higher or lower density area; otherwise, it is defined as *background*. The true label set can be calculated because we know the exact shape of the virtual phantom. The PS-MRF model generates an estimated label set, so we can compare this label set with the true label set. For other reconstruction methods that do not segment the voxel grid, we classified the reconstructed voxels using certain thresholds: *background* if $k_1 \leq f_v \leq k_2$; otherwise, *foreground*. The threshold parameters k_1 and k_2 were selected by minimization of the *segmentation error*. This is the ideal threshold segmentation for comparisons, and this type of segmentation is not possible in reality.

The PS-MRF requires some pre-processed inputs and parameters. The variance of observation noise was set to $\sigma_p^2 = 8^2$ for all p. The smoothness parameter was set to $\psi = 1$. The discontinuity penalty was set to $\alpha = 8$, which resulted in a rapid and stable jump over '8' likely to be a boundary.
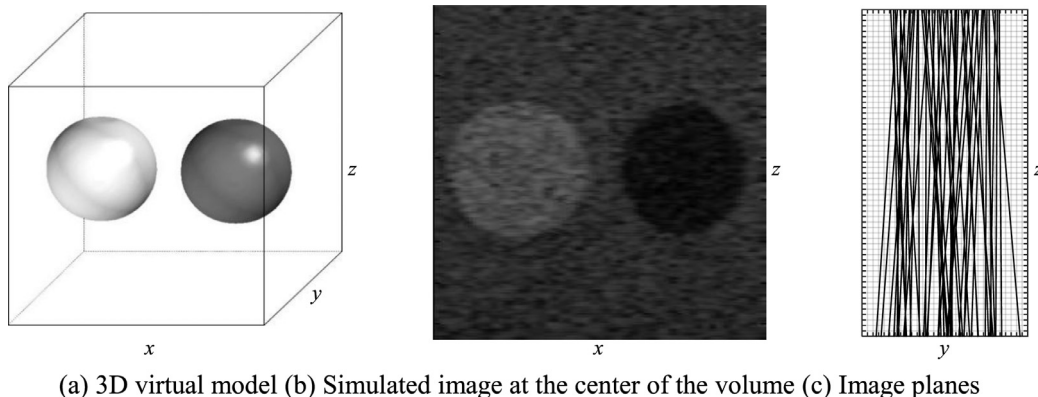

(a) 3D virtual model (b) Simulated image at the center of the volume (c) Image planes

Fig. 6. Virtual phantom; the grid in (c) has 1 mm spacing.

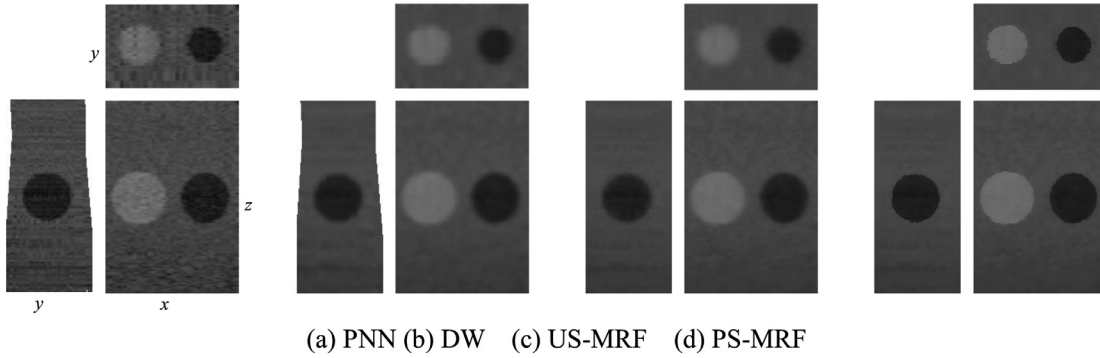(a) PNN (b) DW   (c) US-MRF   (d) PS-MRF

Fig. 7. Comparison of sectional images generated using the four reconstruction methods.

The noise variance parameters σ_p^2 were determined by measurement of the speckle noise variance from the sampled images. This measurement might not be very accurate. However, obtaining very accurate σ_p^2 values is not important because σ_p^2 does not directly affect the smoothness of the results. Rather than σ_p^2 itself, the ratio between the aggregate precision of a voxel $\sum_{p \in \mathcal{P}_v} 1/\sigma_p^2$ and the smoothness parameter ψ determines the smoothness of the voxel v. Eq. (8) implies that a neighbor voxel can be considered as another observation with a noise variance of $1/\psi$. Therefore, a neighbor voxel has a weight of $\sigma_p^2 \times \psi$ observations within the constant σ_p^2 settings, because a neighbor has the precision ψ, and an observation has the precision $1/\sigma_p^2$. In this experiment, the number of observations per voxel was approximately 25, on average, over the voxels that had at least one observation, and a neighbor had a weight of 8^2 observations. The ratio between the two numbers of observations enabled an almost speckle-free reconstruction result in the experiment. The smaller ψ, the closer to the original ultrasound image reconstruction result that can be generated, but it is noisier. In this case, speckles might be assigned to different labels, and this is undesirable. Therefore, a higher discontinuity penalty α is recommended for a smaller ψ in order to prevent the undesired separation of speckle noise in our experiments. For example, some speckles were separated using a different label with $\psi = 0.125$ and $\alpha = 10$, but the parameters of $\psi = 0.125$ and $\alpha = 15$ produced clear object boundaries.

We measured the computation times using a desktop computer with an Intel i7-3770 CPU and NVidia GTX 780 graphics card including 2304 CUDA cores. All CPU computations were single-threaded for fair time comparisons. For the PS-MRF, the CPU and GPU implementations were based on Algorithm 1 without reuse of flows and Algorithm 2 with reuse of flows, respectively. The coarse grid of Algorithm 2 is constructed with 2 *mm* voxel spacing. The quality performance results for each method are summarized in Table 1. The computation time in Table 1 is the average time for a reconstruction. As shown in Table 1, the PS-MRF method outperformed all other methods. From the computation perspective, both the CPU and GPU implementations took 3 iterations on the 0.25 mm spaced grid. Hence, the speed up from 436.49 s to 9.41 s is due to the reuse of flows and the computation power of the GPU.

Table 1
Quality measures of the *Two spheres* results.

	Reconstruction error	Interpolation error	Segmentation error	Computation time (s)
PNN	7.68	8.63	2.06%	26.97
DW	6.05	8.25	1.43%	391.87
US-MRF	6.11	8.24	1.34%	14.02
PS-MRF	**6.03**	**8.18**	**1.33**%	CPU: 436.49 GPU: 9.41

Sectional images of the reconstructed 3D voxel grid with full data are presented in Fig. 7 with 0.25 mm voxel spacing.

Fig. 8 presents the intensity profiles of the reconstructed volumes along the *x*-axis at $y = 63$ and $z = 140$. The image plane that crossed $y = 63$ and $z = 140$ was removed for the reconstruction, so we could compare the reconstructed data and the original data that was removed at the reconstruction. As seen in Fig. 8, the PNN and DW generated false speckle noises, and the DW and US-MRF blurred the boundary information. In contrast, the PS-MRF captured speckle-free signal values while preserving the boundary information.

3.2. Scanned data

3.2.1. Data acquisition

To reconstruct a 3D model from a real ultrasound image, the global position and orientation data of each image was measured simultaneously by the optical tracking system depicted in Fig. 9.

The pixel coordinates of the ultrasound image $\mathbf{u} = (u, v, 0, 1)^T$ can be converted to the world coordinate $\mathbf{x} = (x, y, z, 1)^T$ by a simple transformation $\mathbf{x} = T_W^T T_T^M T_M^P T_P^I \mathbf{u}$. Here, T_W^T is the world-to-tracker coordinate transformation matrix, T_T^M represents the tracker-to-marker coordinate transformation, and T_M^P is the
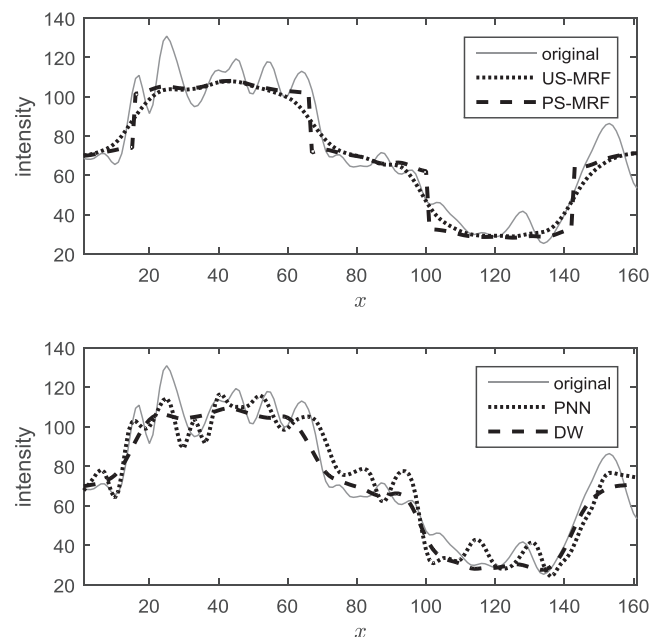

Fig. 8. Intensity profiles of the reconstructed volumes along the *x*-axis at the center of the data-removed section.
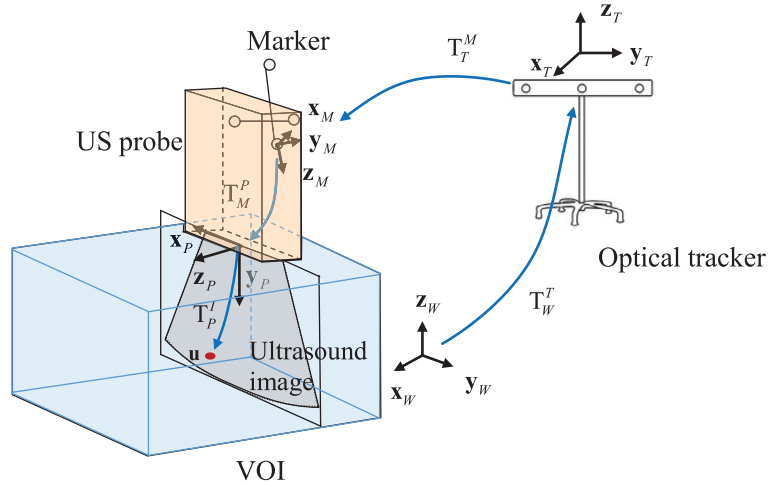
Fig. 9. Conceptual model of the scanning system [5].

transformation matrix of the marker-to-ultrasound probe. The 4 × 4 scale matrix has diagonal elements $(s_x, s_y, 1, 1)$, which represent the pixel-spacing values of the ultrasound image. For the probe calibration, the traditional crosswire phantom was used [7], and the T_M^P matrix and s_x, s_y values were obtained using the Levenberg-Marquardt method [5]. While scanning a VOI using a calibrated ultrasound probe, our system automatically saved the ultrasound images with their probe position at a frequency of 10 Hz.

A liver phantom model was obtained via casting, using a frozen gelatin solution based on a silicone mold made of MoldMax 30 (SmoothOn Inc., Easton, PA, USA). The initial shape of the liver was obtained from a 3D virtual model and created using a 3D printer to make a mold. Finally, we put the phantom liver into a gelatin solution with a different concentration level to be scanned using the ultrasound device (SonixTouch, Analogic Co., Peabody, MA, USA). All images were scanned with an L14-6 linear probe, and each B-scan image had 450 × 450 pixels.

3.2.2. Reconstruction results

The 3D model was reconstructed using 149 ultrasound images for 57 × 54 × 44 mm³ area (230 × 216 × 177 voxel grid with 0.25 mm voxel spacing). The parameter values of $\psi = 0.5$ and $\alpha = 15$ were used in the PS-MRF model. In this experiment, the noise variance parameter σ_p^2 was applied differently according to the position of the pixel in the image for the PS-MRF model. We can observe that the images generated using our machine exhibited a consistent focal zone. Approximately 30 pixel widths to left and right of the images and 100 pixel widths at the bottom of images were considered far from the focal zone; data in the non-focal zone exhibited degraded quality due to signal attenuation and lower resolution. Therefore, we set σ_p^2 in the non-focal zone to 32^2 and σ_p^2 in the focal zone to 4^2. Varying the coefficient σ_p^2 is

particularly useful when accurate data and inaccurate data overlap. In our experiments, the probe moved back and forth creating a "V" shape; thus, the focal zone and non-focal zone overlapped. For the US-MRF model, σ_p^2 was set to 4^2 for all pixels.

In terms of smoothing strength, the number of observations per voxel was approximately 55, on average, for the voxels that had at least one observation, and a neighbor voxel had a weight of $4^2 \times 0.5$ observations, if we assumed that all pixels had the same variance coefficient $\sigma_p^2 = 4$. Hence, the smoothing strength was much weaker than in previous experiments. We intended the parameter settings to preserve small features in the volume, and α was increased to be higher than in previous experiments to compensate the low smoothing strength, as discussed in Section 3.1.

To evaluate the reconstruction quality, the interpolation errors at the focal zone were measured. Due to the poor quality of the original signals at the non-focal zone, we only compared intensity values of the focal zone in the removed scan image with the intensity values of the reconstructed volume. In addition, we measured surface differences because the ground truth of the surface model (Fig. 10(a)) was known in this case. For this purpose, we measured surface feature points from the phantom liver using the optical tracker and converted them to global coordinates. We then found the rigid transformation from the initial (virtual) surface model to the phantom liver by the iterative closest points (ICP) method after initial feature point matching. The transformed virtual surface was used as a ground truth. Iso-surfaces were extracted from the reconstructed volumes, and we measured the Euclidean distance from each vertex of the transformed virtual surface model (ground truth) to the iso-surfaces. We computed root mean squares of error (RMSE) as well as the mean and standard deviations of the distances. For all other methods except PS-MRF, which does not
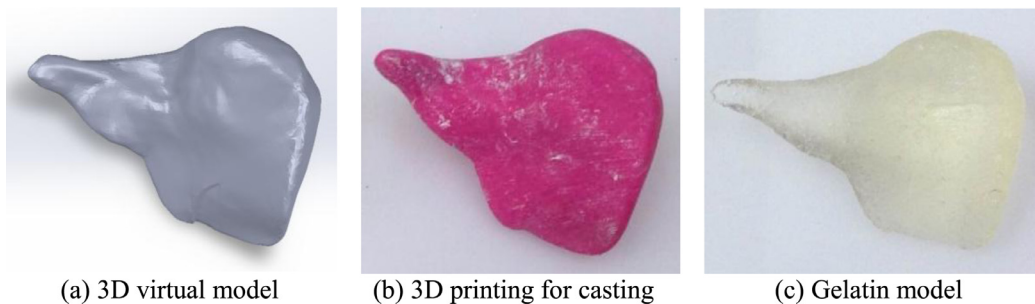

(a) 3D virtual model   (b) 3D printing for casting   (c) Gelatin model

Fig. 10. Physical phantom models.

Table 2
Quality measures of the *Scanned data* results.

	Interpolation error	Surface error (mm)		Computation time (s)
		Mean ± standard deviation	RMSE	
PNN	6.38	1.02 ± 0.67	1.22	55.65
DW	6.64	1.03 ± 0.69	1.24	328.79
US-MRF	6.24	1.00 ± 0.71	1.22	47.93
PS-MRF	**5.89**	**0.80 ± 0.68**	**1.05**	CPU: 25297.79 GPU: 22.06

produce surface information, iso-surfaces were used for comparison. Specifically, the iso-surface level was set to 20, which provides accurate surfaces of object boundaries. When computing the surface errors, we only measured one-side distances from the ground truth to the iso-surfaces to avoid the dark regions by signal attenuation in the deep scan area being included in the computation. Quality measures are presented in Table 2.

The PS-MRF method produced significant lower interpolation error, because the reconstructed volumes of other methods except the PS-MRF method were corrupted by the degraded signals of the non-focal zone data, which is due to the fact that all input pixels are considered as having the same precision. Moreover, the surface error was minimized with the PS-MRF method. In terms of computation time, Algorithm 1 consumed 28 iterations, and Algorithm 2

consumed just two iterations on the fine-grid because the energy was almost minimized at the coarse-grid stage with a negligible computation time. As a result, the computation time gap between CPU with Algorithm 1 and GPU with Algorithm 2 was substantial.

Figs. 11 and 12 show the sectional images and 3D surfaces of the reconstructed models, respectively. For results that do not contain boundary information (i.e. the PNN, DW, and US-MRF methods), the iso-surfaces at two different levels were drawn.

The sectional images demonstrate that the PNN, DW, and US-MRF methods blurred the object boundaries and that the PNN and DW methods did not fill all voxels in the grid. These characteristics increase the difficulty of further data analyses, such as segmentation and boundary detection. In contrast, the PS-MRF method provided clear boundaries, and all voxels were filled. As seen from the iso-surfaces, inaccurate boundaries were created under the object due to the gradual decay of the ultrasound signal.

Fig. 13 presents the reconstruction result with (a) a constant σ_p^2 and (b) a varying σ_p^2. The vertical dark lines, which arise from the axial border of the scan images and are indicated by red arrows, are recovered by the data from the focal zone in Fig. 13(b). We also note that the liver boundaries, which are indicated by blue arrows in Fig. 13(b), exhibited different reconstruction results. Some boundaries exhibited less aliasing because lowering the precision of the non-focal area causes the data fitness penalties of the energy function (11) to be lower. Consequently, the boundary penalties affect the energy more than in case (a).
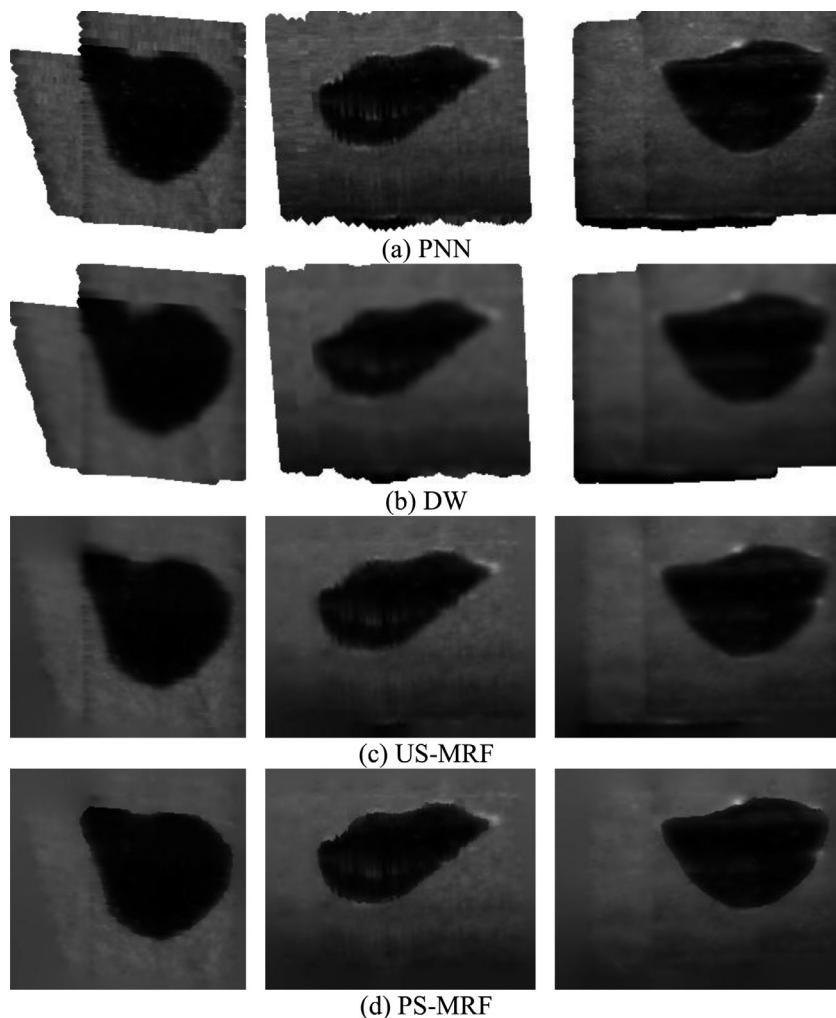

(a) PNN

(b) DW

(c) US-MRF

(d) PS-MRF

Fig. 11. 2D sectional images of the reconstructed volume (*xy*, *yz*, and *xz* planes from left to right).

(a) Iso-surfaces of PNN, DW, and US-MRF (left to right) at level 15.

(b) Iso-surfaces of PNN, DW, and US-MRF (left to right) at level 20.

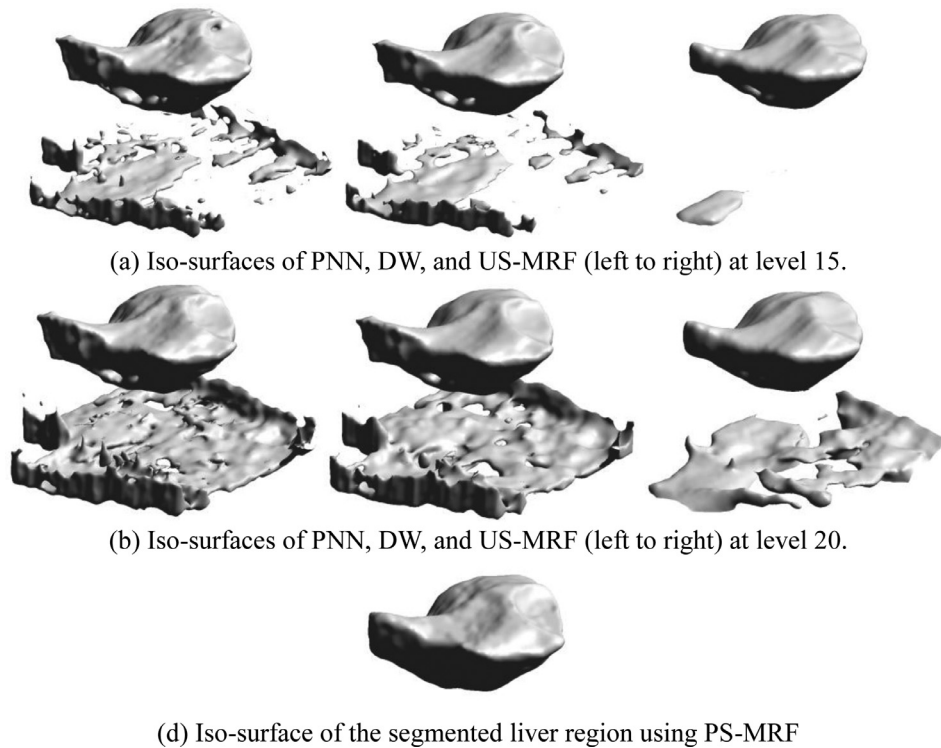(d) Iso-surface of the segmented liver region using PS-MRF

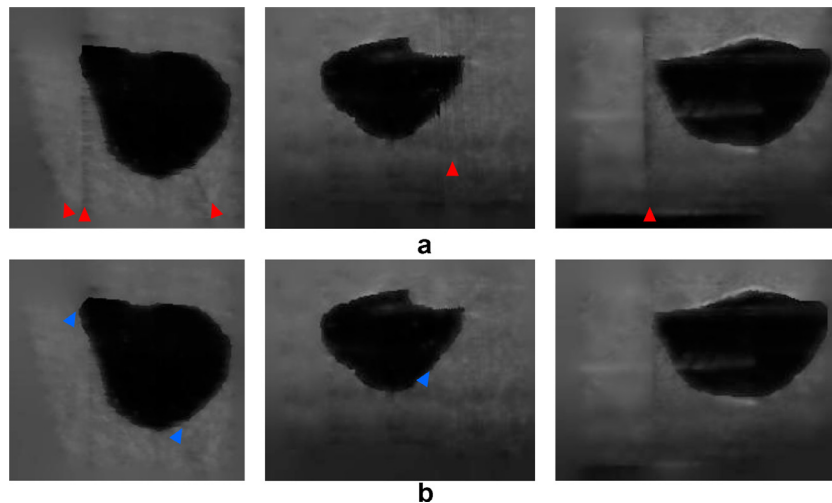**Fig. 12.** 3D comparison via extraction of iso-surfaces.

Fig. 13. PS-MRF reconstruction with consideration of (a) uniform and (b) varying noise models.

4. Conclusion and discussion

We proposed and developed an MRF model to reconstruct a 3D volume from freehand ultrasound images. Our model can reduce speckle noise and can easily be expanded for a piecewise smooth model for boundary conservation. The region-based PS-MRF model provides a simple and intuitive formulation and generates high-fidelity results from irregular and noisy input. This model is also easily optimized for binary labels. For computational efficiency, which is critical for clinical applications such as image-guided interventions, we proposed an improved algorithm and parallelized the optimization step. In our experiments, the computation time of Algorithm 2 using GPUs was at least 46.39 times faster than Algorithm 1 using a CPU.

Our PS-MRF method was compared with traditional reconstruction methods using synthetic ultrasound images and real ultra-

sound images by assessing the methods in terms of the proposed quality measures. The results demonstrated that the object boundaries generated by the PS-MRF model were significantly clearer and that the image noise was reduced significantly. The interpolation and segmentation error values were low, while the computation was performed in a reasonable time using GPUs. Because a simple classification can be performed simultaneously with the reconstruction step, clear 2D and 3D visualizations are possible without additional computation for various clinical and research purposes (e.g. embryology). We also expect that our noise-reduced reconstruction will be useful for image-guided surgery and interventions without radiation exposure.

In our experiments, we simply determined the confidence level of each pixel by dividing B-scan images into focal/non-focal zones. However, several studies have measured the confidence levels of each pixel in more systematic ways. In particular, [41] focused on

creating an acoustic shadow map, and [42] proposed a method of generating a confidence map including the shadow region. The reconstruction quality of our method can be further improved by combining these approaches, although there are tradeoffs to incorporate the material/content-dependent processing in the reconstruction step. Because the internal materials are not known in the reconstruction step, content-dependent processing requires additional computations for each B-scan image. Depending on clinical applications, there can be trade-offs between the computation time and quality in practical use.

In many anatomical images, the use of multiple labels might provide better reconstruction results than binary labels. The appropriate cardinality of labels $|\mathbb{L}|$ can be determined using simple intensity analyses, such as counting the peaks in a smoothed histogram, or any other classification, as a preprocessing step. Another possibility of determining the number of labels is Bayesian model comparison methods within the probabilistic inference framework. If the number of labels is determined, the formulation of multi-label cases can be extended from Eq. (11). The energy function consists of three types of penalty: data fitness, smoothness, and boundary length. The data fitness and smoothness penalties at voxel v occur only in the selected layer l_v, and the boundary length penalties occur when any two neighbor voxels have different labels. Therefore, the energy function of the multi-label case can be easily extended from Eq. (11) by setting the outer summation index h from 0 to $k-1$ for the k-label formulation.

From the algorithmic aspect, multi-label extension can be achieved by modifying Algorithm 1 in two aspects: the interpolation step and the segmentation step. The interpolation step is straightforward for the multi-label case. If the number of labels is L, there are L layers to interpolate. Each layer is interpolated in the same way as the binary label case. In contrast, optimizing multi-label segmentation is a NP-hard problem in contrast with the binary label case [43]. However, there are efficient optimization methods called $\alpha - \beta$ swap and α expansion, which are known to perform well on a variety of computer vision problems [43].

As additional further work, the machine-dependent parameters in the observation model, such as the variance coefficients σ_p^2, should be estimated for various machines and probes as well as for various materials. We have tested and used the value estimated under fixed settings. However, an ultrasound image exhibits different intensities and qualities according to the frequency and other scanning parameters. Sensitivity analyses using such parameters determined by the characteristics of the machines would be necessary in order to optimize the user-defined parameters. The probe characteristics, speckle strengths, and any other knowledge about noise and artifacts could be applied.

Acknowledgments

References

[1] D.H. Pretorius, N.N. Borok, M.S. Coffler, T.R. Nelson, Three-dimensional ultrasound in obstetrics and gynecology, Radiol. Clin. North Am. 39 (2001) 499–521, doi:10.1016/S0033-8389(05)70294-3.

[2] L. Gindes, J. Hegesh, B. Weisz, Y. Gilboa, R. Achiron, Three and four dimensional ultrasound: a novel method for evaluating fetal cardiac anomalies, Prenat. Diagn. 29 (2009) 645–653, doi:10.1016/S0033-8389(05)70294-3.

[3] G. Unsgaard, S. Ommedal, T. Muller, A. Gronningsaeter, T.A. Nagelhus Hernes, Neuronavigation by intraoperative three-dimensional ultrasound: initial experience during brain tumor resection, Neurosurgery 50 (2002) 804–812 discussion 812.

[4] A. Fenster, K. Surry, W. Smith, J. Gill, D.B. Downey, 3D ultrasound imaging: applications in image-guided therapy and biopsy, Comput. Graph. 26 (2002) 557–568, doi:10.1016/S0097-8493(02)00101-2.

[5] O. Minin (Ed.), Ultrasound Imaging – Medical Applications, InTech, 2011, doi:10.5772/689.

[6] G. Treece, Volume measurement and surface visualisation in sequential freehand 3d ultrasound (Doctoral dissertation), University of Cambridge, 2000.

[7] L. Mercier, T. Langø, F. Lindseth, D.L. Collins, A review of calibration techniques for freehand 3-D ultrasound systems, Ultrasound Med. Biol. 31 (2005) 143–165, doi:10.1016/j.ultrasmedbio.2004.11.001.

[8] J. Huang, X. Yang, Fast reduction of speckle noise in real ultrasound images, Signal Process. 93 (2013) 684–694, doi:10.1016/j.sigpro.2012.09.005.

[9] J.T. Bushberg, A.J. Seibert, E.M. Leidholdt, J.M. Boone, The Essential Physics of Medical Imaging, 3rd ed., Lippincott Williams & Wilkins, Philadelphia, PA, 2012.

[10] R. Rohling, A. Gee, L. Berman, A comparison of freehand three-dimensional ultrasound reconstruction techniques, Med. Image Anal 3 (1999) 339–359, doi:10.1016/S1361-8415(99)80028-0.

[11] O.V. Solberg, F. Lindseth, H. Torp, R.E. Blake, T. a Nagelhus Hernes, Freehand 3D ultrasound reconstruction algorithms—a review, Ultrasound Med. Biol. 33 (2007) 991–1009, doi:10.1016/j.ultrasmedbio.2007.02.015.

[12] Q.-H. Huang, Y.-P. Zheng, Volume reconstruction of freehand three-dimensional ultrasound using median filters, Ultrasonics 48 (2008) 182–192, doi:10.1016/j.ultras.2007.11.005.

[13] T. Wen, Q. Zhu, W. Qin, L. Li, F. Yang, Y. Xie, et al., An accurate and effective FMM-based approach for freehand 3D ultrasound reconstruction, Biomed. Signal Process. Control 8 (2013) 645–656, doi:10.1016/j.bspc.2013.05.009.

[14] P. Coupé, P. Hellier, X. Morandi, C. Barillot, Probe trajectory interpolation for 3D reconstruction of freehand ultrasound., Med. Image Anal. 11 (2007) 604–615, doi:10.1016/j.media.2007.05.002.

[15] J.M. Sanches, J.S. Marques, A Rayleigh reconstruction/interpolation algorithm for 3D ultrasound, Pattern Recognit. Lett. 21 (2000) 917–926, doi:10.1016/S0167-8655(00)00053-2.

[16] J. Sanches, J. Marques, A fast MAP algorithm for 3D ultrasound, in: Proceedings of EMMCVPR 2001, Springer, Berlin, Heidelberg, 2001, pp. 63–74, doi:10.1007/3-540-44745-8_5.

[17] S.Z. Li, Markov Random Field Modeling in Image Analysis, Springer, London, 2009, doi:10.1007/978-1-84800-279-1.

[18] R. Kindermann, J.L. Snell, Markov Random Fields and Their Applications, American Mathematical Society, Providence, Rhode Island, 1980, doi:10.1090/conm/001.

[19] J.M. Hammersley, P. Clifford, Markov fields on finite graphs and lattices, 1971 (Unplublished manuscript). http://www.statslab.cam.ac.uk/~grg/books/hammfest/hamm-cliff.pdf.

[20] G. Strang, Computational Science and Engineering, Wellesley-Cambridge Press, 2007.

[21] J. Wu, A.C.S. Chung, A segmentation model using compound Markov random fields based on a boundary model, IEEE Trans. Image Process. 16 (2007) 241–252, doi:10.1109/TIP.2006.884933.

[22] Xuezhi Yang, D.A. Clausi, Evaluating SAR sea ice image segmentation using edge-preserving region-based MRFs, IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 5 (2012) 1383–1393, doi:10.1109/JSTARS.2012.2217940.

[23] J. Marroquin, S. Mitter, T. Poggio, Probabilistic solution of ill-posed problems in computational vision, J. Am. Stat. Assoc. 82 (1987) 76–89, doi:10.1080/01621459.1987.10478393.

[24] J.A. Noble, D. Boukerroui, Ultrasound image segmentation: a survey, IEEE Tansactions Med. Imaging 25 (2006) 987–1010, doi:10.1109/TMI.2006.877092.

[25] Y. Boykov, G. Funka-Lea, Graph cuts and efficient N-D image segmentation, Int. J. Comput. Vis. 70 (2006) 109–131, doi:10.1007/s11263-006-7934-5.

[26] N. Otsu, A threshold selection method from Gray-level histograms, IEEE Trans. Syst. Man. Cybern. 9 (1979) 62–66, doi:10.1109/TSMC.1979.4310076.

[27] L. Grady, C.V Alvino, The piecewise smooth Mumford–Shah functional on an arbitrary graph, IEEE Trans. Image Process. 18 (2009) 2547–2561, doi:10.1109/TIP.2009.2028258.

[28] Y. Boykov, V. Kolmogorov, Computing geodesics and minimal surfaces via graph cuts, in: Proceedings of the Ninth IEEE International Conference on Computer Vision, vol. 1, IEEE, 2003, 2003, pp. 26–33, doi:10.1109/ICCV.2003.1238310.

[29] V. Kolmogorov, R. Zabih, What energy functions can be minimized via graph cuts? IEEE Tans. Pattern Anal. Mach. Intell. 26 (2004) 147–159, doi:10.1109/TPAMI.2004.1262177.

[30] Y. Boykov, V. Kolmogorov, An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision, IEEE Trans. Pattern Anal. Mach. Intell. 26 (2004) 1124–1137, doi:10.1109/TPAMI.2004.60.

[31] A.V Goldberg, R.E. Tarjan, A new approach to the maximum flow problem, in: Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing, ACM, New York, NY, USA, 1986, pp. 136–146, doi:10.1145/12130.12144.

[32] Y. Boykov, M.-P. Jolly, Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images, in: Proceedings of the Eighth IEEE International Conference on Computer Vision, ICCV 2001, IEEE Computer Society, 2001, pp. 105–112, doi:10.1109/ICCV.2001.937505.

[33] GridCut, 2015. http://gridcut.com/ (accessed 17.08.15).

[34] J. Liu, J. Sun, Parallel graph-cuts by adaptive bottom-up merging, in: Proceedings of the 2010 IEEE Conference on Computer Vision Pattern Recognition, IEEE, 2010, pp. 2181–2188, doi:10.1109/CVPR.2010.5539898.

[35] O. Jamriska, D. Sykora, A. Hornung, Cache-efficient graph cuts on structured grids, in: Proceedings of the 2012 IEEE Conference on Computer Vision Pattern Recognition, IEEE, 2012, pp. 3673–3680, doi:10.1109/CVPR.2012.6248113.

[36] CUDA. https://developer.nvidia.com/cuda-zone (accessed 06.01.16).

[37] V. Vineet, P.J. Narayanan, CUDA cuts: Fast graph cuts on the GPU, in: Proceedings of the IEEE Computer Society Conference on Computer Vision Pattern Recognition Workshop, IEEE, 2008, pp. 1–8, doi:10.1109/CVPRW.2008.4563095.

[38] C.D. Barry, C.P. Allott, N.W. John, P.M. Mellor, P.A. Arundel, D.S. Thomson, et al., Three-dimensional freehand ultrasound: Image reconstruction and volume analysis, Ultrasound Med. Biol. 23 (1997) 1209–1224, doi:10.1016/S0301-5629(97)00123-3.

[39] J.A. Jensen, N.B. Svendsen, Calculation of pressure fields from arbitrarily shaped, apodized, and excited ultrasound transducers, IEEE Trans. Ultrason. Ferroelectr. Freq. Control 39 (1992) 262–267, doi:10.1109/58.139123.

[40] J.A. Jensen, FIELD: A program for simulating ultrasound systems, in: Proceedings of the 10th Nordic-Baltic Conference on Biomedical imaging, vol. 4, Suppl. 1, part 1, 1996, pp. 351–353.

[41] P. Hellier, P. Coupé, X. Morandi, D.L. Collins, An automatic geometrical and statistical method to detect acoustic shadows in intraoperative ultrasound brain images, Med. Image Anal. 14 (2010) 195–204, doi:10.1016/j.media.2009.10.007.

[42] A. Karamalis, W. Wein, T. Klein, N. Navab, Ultrasound confidence maps using random walks, Med. Image Anal. 16 (2012) 1101–1112, doi:10.1016/j.media.2012.07.005.

[43] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, IEEE Trans. Pattern Anal. Mach. Intell. 23 (2001) 1222–1239, doi:10.1109/34.969114.