Automatic 3D foetal face model extraction from ultrasonography through histogram processing

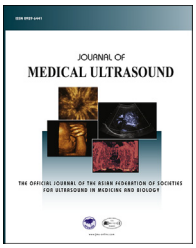
Journal of Medical Ultrasound (2016) **24**, 142—149

Chinese Taipei Society of Ultrasound in Medicine

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.jmu-online.com



ORIGINAL ARTICLE

Automatic 3D foetal face model extraction from ultrasonography through histogram processing

Luca Bonacina [1], Antonio Froio [2], Daniele Conti [3], Federica Marcolin [4], Enrico Vezzetti [4*]

[1] *Department of Electronics, Information and Bioengineering, Polytechnic University of Milan, Milan,* [2] *Department of Energy,* [3] *Department of Applied Science and Technology, and* [4] *Department of Management Engineering and Production, Polytechnic University of Turin, Turin, Italy*

Received 13 May 2016; accepted 28 July 2016
Available online 10 September 2016

KEYWORDS
3D histogram processing,
3D ultrasound,
DICOM,
Image segmentation

Abstract Ultrasound is by far the most adopted method for safe screening and diagnosis in the prenatal phase, thanks to its non-harmful nature with respect to radiation-based imaging techniques. The main drawback of ultrasound imaging is its sensitivity to scattering noise, which makes automatic tissues segmentation a tricky task, limiting the possible range of applications.

An algorithm for automatically extracting the facial surface is presented here. The method provides a comprehensive segmentation process and does not require any human intervention or training procedures, leading from the output of the scanner directly to the 3D mesh describing the face. The proposed segmentation technique is based on a two-step statistical process that relies on both volumetric histogram processing and 2D segmentation. The completely unattended nature of such a procedure makes it possible to rapidly populate a large database of 3D point clouds describing healthy and unhealthy faces, enhancing the diagnosis of rare syndromes through statistical analyses.

Conflict of interest: none.

* Correspondence to: Professor Enrico Vezzetti, Department of Management Engineering and Production, Polytechnic University of Turin, c.so Duca degli Abruzzi 24, Turin 10129, Italy.
E-mail address: enrico.vezzetti@polito.it (E. Vezzetti).

http://dx.doi.org/10.1016/j.jmu.2016.08.003

Introduction

The popularity of 3D ultrasound (US) scans (Figure 1) has rapidly increased during recent years [1]. However, even though it could provide information to physicians, its importance as a diagnostic tool is somewhat limited in prenatal imaging, as nowadays it is mainly used as a visualization tool. On the other hand, it is self-evident that a detailed 3D model of the fetus could play a crucial role in prenatal diagnosis. For instance, the detection of dysmorphic features could be tackled by exploiting biometrics and landmark identification on a 3D model of the fetus [2–9]. Algorithms and software exist for 3D US segmentation and volume extraction, mainly based on region growing [10, 11] or deformable models [12, 13], but human intervention is requested, e.g., to choose the seeds or for pre/post-processing. The algorithm by Feng et al. [14] is automatic, but it requires a training stage before it can be applied; moreover, its output is a coarse mesh.

The proposed method automatically extracts relevant 3D information of the fetuses face from a set of planar 2D US slices, outputting a high resolution mesh of the facial surface; this does not require any training step as it is not based on machine learning techniques. To achieve the maximum interoperability, the whole procedure has been designed and tested with DICOM [16] files as input and a triangular mesh in PLY [17] files as output. Although the algorithm has been implemented using MATLAB®, it is easily portable to any other language.

The main phases of the procedure are shown in Figure 2.

Methods

Preprocessing

The DICOM is initially imported as a stack of 2D planar slices. This can be done in any of the three coordinate directions, as the algorithm does not depend on the position of the fetus. Input stack I is a now an N-dimensional matrix (3D in ultrasonography case) such that:

$$I(\mathbf{v}) = g, \tag{1}$$

where $\mathbf{v} = (r,c,s)$ is a voxel, r,c,s are respectively row, column, and slice indexes, and g is an integer value ($0 \leq g \leq (2^N - 1)$, where N is the image bit depth) representing gray-scale.

US wave scattering damages the images by introducing a type of noise called *speckle* [18]. In order to enhance segmentation performance, noise is removed through filtering in the first phase. Current state-of-the-art algorithms performing this task are based on statistical analysis, inverse PSF filtering, and log separation [19]. In our algorithm, a standard 3D low-pass Gaussian filtering (or, alternatively, a Laplacian of Gaussian to better preserve the edges) was accurate enough for a preliminary speckles removal. Nonetheless, the most effective image cleaning is done *after* the extraction of the surface. After this preprocessing step, the image improves in terms of quality and helps to enhance the performance, although it is not yet clean enough for achieving complete surface extraction.

Surface reconstruction through 3D histogram processing

The most widely adopted method to extract a volume from a point cloud relies on the *marching cube* algorithm [20]. To extrapolate the facial *surface*, the detection (per each DICOM slice) of the edges of the face itself would be desirable.

Both volume and surface reconstruction are demanding tasks, due to the coexistence of different tissues in the same image (from both mom and fetus). Edge extraction is even trickier when the fetuses head lays directly on the mother's uterus; in this scenario, the two tissues are virtually merged from the US imaging standpoint. To face this situation, a novel algorithm has been developed, based on a statistical analysis of local and 3D histograms that combine slice and volumetric processing.

Volumetric processing

As a first step, the algorithm processes the entire stack as a whole, applying a star-shaped cubic kernel to each voxel of the 3D array higher than a threshold gray-scale value t.

The kernel support is a subset of the stack. In particular, for each voxel $\mathbf{v} = (r,c,s)$, considering a side $2l + 1$ (with l chosen by the user, in order to finely tune the result) and the cube $\mathcal{C}^{\mathbf{v}}(l) = \{[r-l, r+l] \times [c-l, c+l] \times [s-l, s+l]\}$ centered in $\mathbf{v}$, the kernel $\mathcal{K}_l^{\mathbf{v}}$ will be:

Figure 1 Three examples of 3D ultrasound images exported from 4DView by General Electric [15]. Given that these are rendered 2D images, although image quality is among the best achievable today, they cannot be 3D-processed.

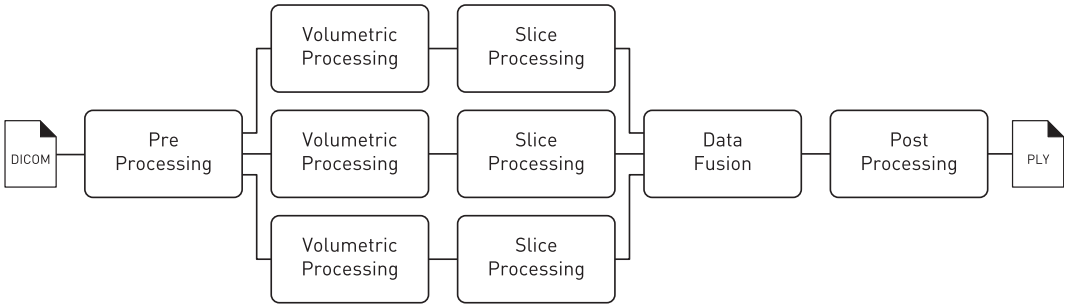


Figure 2 Scheme of the algorithm. The "Volumetric Processing" and "Slice Processing" blocks are repeated three times, extracting the slices from the 3D volume along three perpendicular directions.

$$\mathcal{K}_l^{\mathbf{v}}(i,j,k)=\begin{cases} I(i,j,k) & (i,j,k)\in\mathcal{S}^{\mathbf{v}}(l)\subset\mathcal{C}^{\mathbf{v}}(l) \\ 0 & (i,j,k)\in\mathcal{C}^{\mathbf{v}}(l)\setminus\mathcal{S}^{\mathbf{v}}(l) \end{cases} \tag{2}$$

with:

$$\begin{gathered}\mathcal{S}^{\mathbf{v}}(l)=\{(i,j,k)\in\mathcal{C}^{\mathbf{v}}(l):\\ (|i-r|=|j-c|\wedge k=s)\vee(|i-r|=|k-s|\wedge j=c)\vee\\ \vee(|j-c|=|k-s|\wedge i=r)\vee(|i-r|=|j-c|=|k-s|)\vee\\ \vee[|i-r|+|j-c|+|k-s|=max(|i-r|,|j-c|,|k-s|)]\}\end{gathered} \tag{3}$$

In other words, this is equivalent to extracting the voxels belonging to the main diagonals of $\mathcal{C}^{\mathbf{v}}(l)$, as well as those on the diagonals of the three mid-planes of the cube (i.e., the three coordinate planes passing through **v**) and on the segments passing through **v** on the three coordinate axes. The **0**-centered kernel, with the default value of $l = 10$, is plotted in Figure 3.

The gray-scale histogram $\mathbf{H}^{\mathbf{v}}$ of the kernel $\mathcal{K}_l^{\mathbf{v}}$ will be computed using the standard iterative formula:

$$\mathbf{H}^{\mathbf{v}}\left(\mathcal{K}_l^{\mathbf{v}}(i,j,k)\right)=\mathbf{H}^{\mathbf{v}}\left(\mathcal{K}_l^{\mathbf{v}}(i,j,k)\right)+1,\quad (i,j,k)\in\mathcal{S}^{\mathbf{v}}(l) \tag{4}$$

starting from $\mathbf{H}^{\mathbf{v}} = \mathbf{0}$. A local (per voxel) threshold value L (**v**) is set to the $(100 - p)$th percentile of the histogram, with p being an input parameter to be set by the final user.

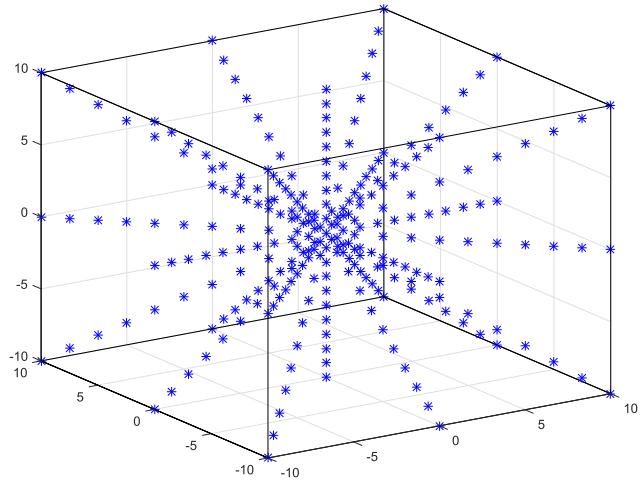


Figure 3 The star-shaped kernel $\mathcal{K}_{10}^{0}$ used in the 3D histogram processing (section 2.2.1), centered in **0**. The blue stars represent the points that are actually considered for the histogram computation; the black line is the side of the cube (equal here to $2l + 1 = 21$ voxels).

Thus, as an intermediate output, a new 3D array is produced, containing in each cell the local threshold value.

Slice processing

A second statistical analysis is then performed on a per-slice basis, i.e., the volume segmentation is obtained via individual segmentation of each 2D slice by means of a new global thresholding technique [21]. All the local threshold values lower than t are set to 0 and the slice histogram is computed. The global slice threshold is set to $\mathcal{T}(s)=M-\lfloor\sigma(2^N-1)\rceil$, where M is the brightest pixel appearing at least once in the slice, σ is the standard deviation of the computed histogram, N is the image bit depth, and $\lfloor\cdot\rceil$ represents the nearest integer function. The volume to be extracted is then mapped into a new output binary slice B (**v**), defined as

$$B(\mathbf{v})=\begin{cases} \text{true} & I(\mathbf{v})\geq\mathcal{T}(\mathbf{s}) \\ \text{false} & \text{otherwise} \end{cases}\quad \forall v=(r,c,s). \tag{5}$$

Each slice of the obtained binary volume represents a "filled" level set of the facial surface (Figures 6c to 10c). The algorithm is reported as pseudo-code in Appendix A.

Data fusion

Potentially, the model extracted after the second step is robust and complete enough to recognize the shape of the

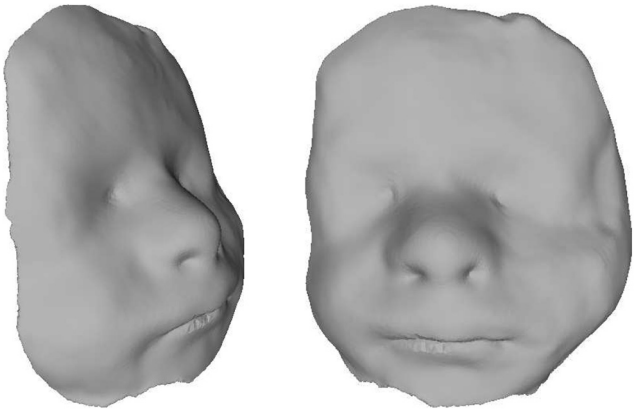


Figure 4 Rendering (produced with MeshLab [26]) of the triangular mesh obtained by the algorithm. The input US is the same as Figures 5 and 6 below.

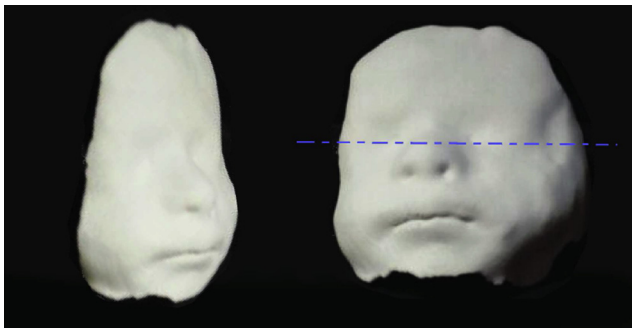


Figure 5 3D printed version of the model shown in Figure 4. The dash-dotted line represents the plane where the slices shown in the next figures are extracted.

fetuses face, but to enhance detection and improve the accurateness we have added a *data fusion* [22] step to the workflow. We repeat the previous 3D shape extraction from all three directions by permuting the dimensions of the DICOM stack, and combine the extracted surfaces evaluating the logical AND of the three binary arrays. This operation increases the robustness of the algorithm, as it is not likely that the artifacts generated by the processing from a viewpoint would be the same as those generated by changing the point of view. This is why the data fusion step generates further noise removal and increased veracity. Moreover, this kind of procedure heavily reduces the number of false positives, i.e., edges that do not actually belong to the desired surface.

Post-processing

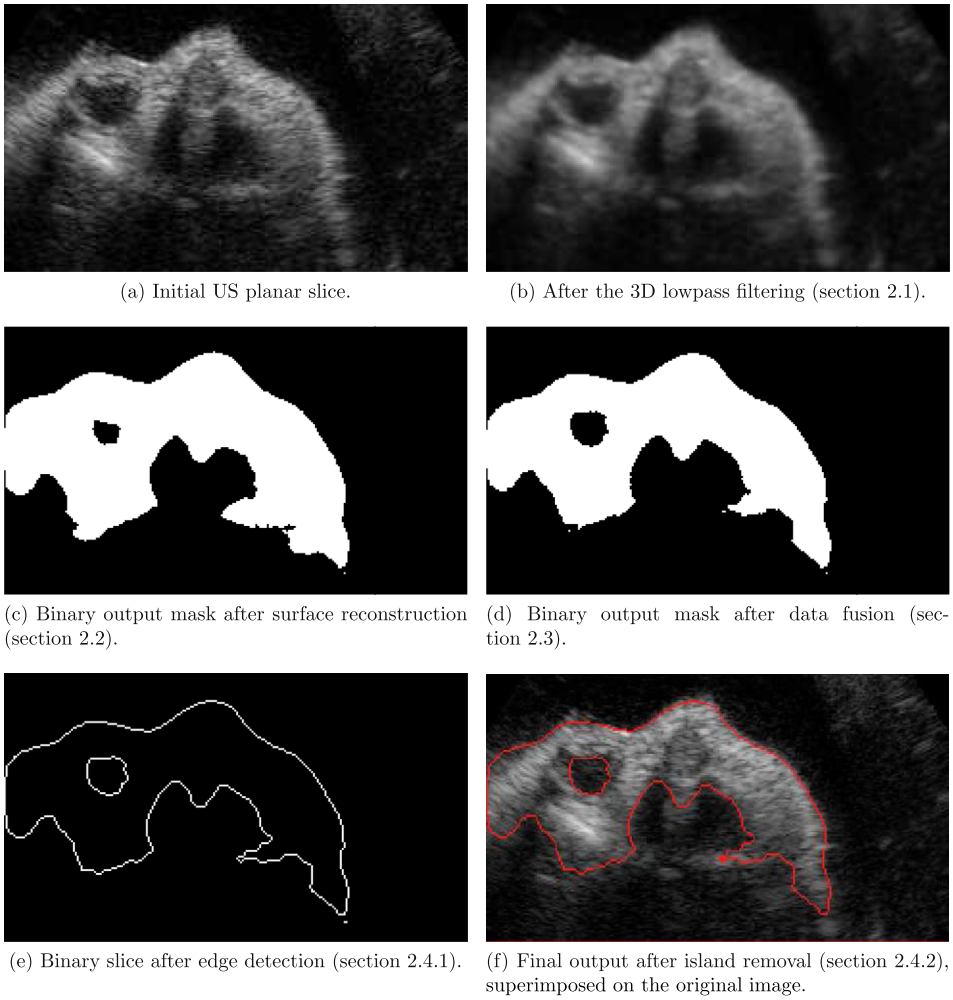
After the data fusion, the output is fairly "clean", but can still be refined via post-processing. The three main operations that take place in this phase are described below.

Edge detection

As a first stage, the Marr—Hildreth *Laplacian of Gaussian* 2D edge detection algorithm [23] is applied to each slice of the fused stack to obtain another binary stack with a more precise definition of the facial surface.

Island removal

The whole 3D image is further "cleaned" by removing the isolated edges that are not connected to the main surface.



(a) Initial US planar slice.

(b) After the 3D lowpass filtering (section 2.1).

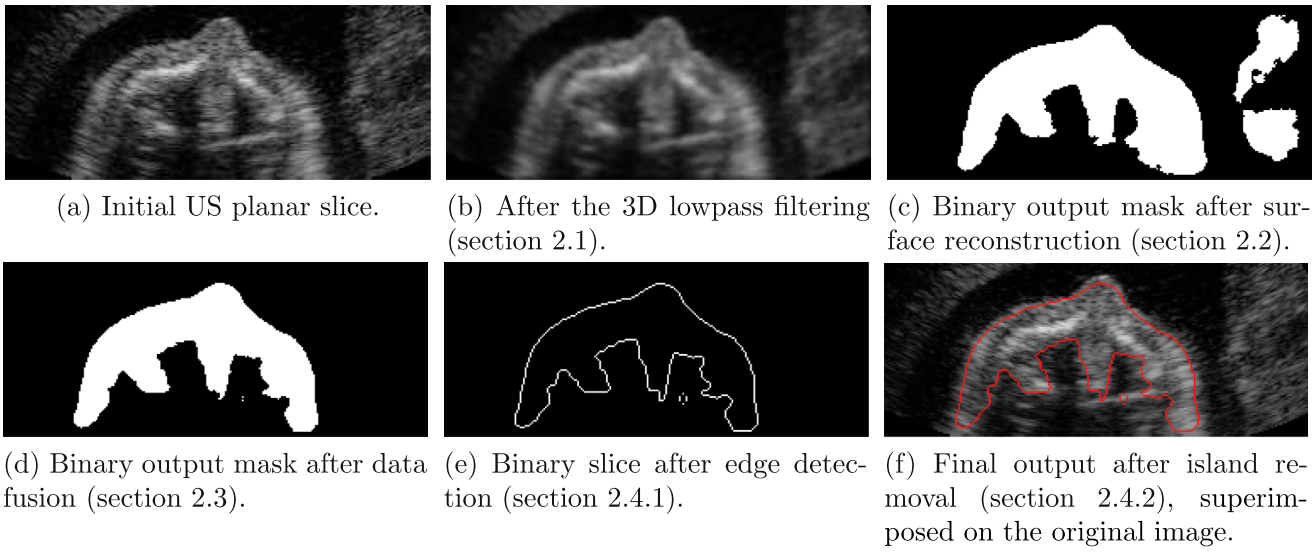
(c) Binary output mask after surface reconstruction (section 2.2).

(d) Binary output mask after data fusion (section 2.3).

(e) Binary slice after edge detection (section 2.4.1).

(f) Final output after island removal (section 2.4.2), superimposed on the original image.

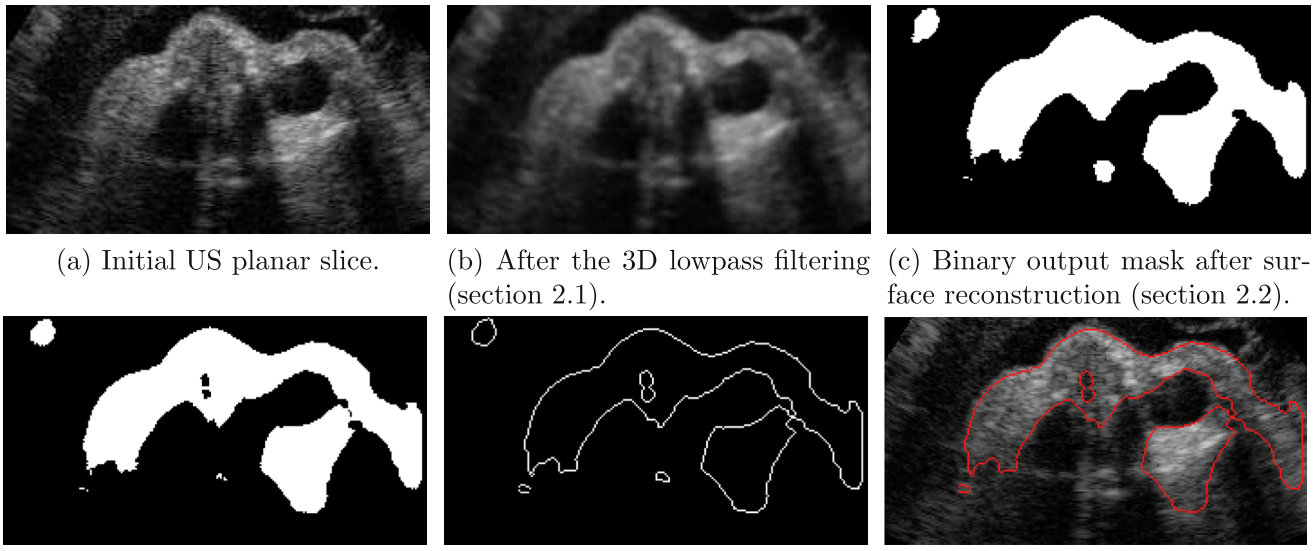
Figure 6 Example of the output of the different steps. The segmentation of the surface of interest is excellent in this case.



(a) Initial US planar slice. (b) After the 3D lowpass filtering (section 2.1). (c) Binary output mask after surface reconstruction (section 2.2).

(d) Binary output mask after data fusion (section 2.3). (e) Binary slice after edge detection (section 2.4.1). (f) Final output after island removal (section 2.4.2), superimposed on the original image.

Figure 7 Example of successful application of the algorithm. The efficacy of the data fusion step is self-evident.



(a) Initial US planar slice. (b) After the 3D lowpass filtering (section 2.1). (c) Binary output mask after surface reconstruction (section 2.2).

(d) Binary output mask after data fusion (section 2.3). (e) Binary slice after edge detection (section 2.4.1). (f) Final output after island removal (section 2.4.2), superimposed on the original image.

Figure 8 Example of the successful application of the algorithm. The efficacy of the island removal is largely evident.

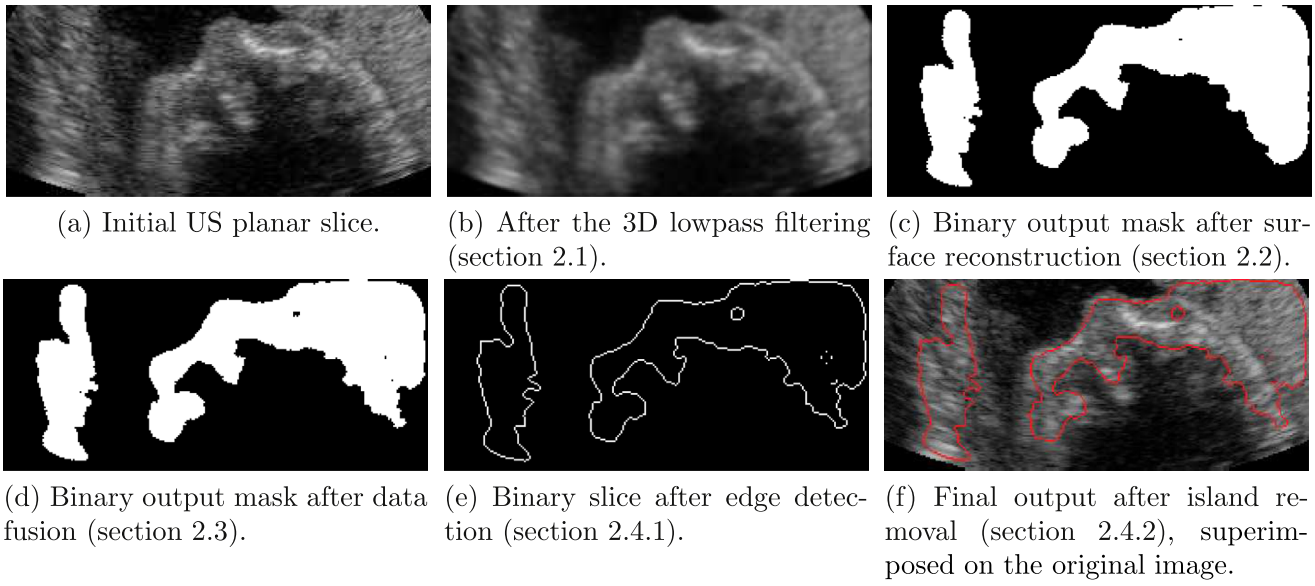
To perform this task, the algorithm searches for all the 3D c-connected components, together with their bounding boxes. c is an input parameter chosen by the user, with a default value of 26. Finally, only the connected components belonging to the r bounding boxes with the largest volumes are kept. The algorithm for island removal is reported as pseudo-code in Appendix B.

Point cloud and mesh generation

In this phase, each true value in the 3D matrix represents a point of the surface. Thus, the final 3D point cloud can be obtained by simply looping over the 3D structure and assigning the Cartesian coordinates to each non-zero element, starting from an arbitrary origin. Finally, the surface is reconstructed via the *Poisson surface reconstruction* [24] algorithm, exploiting the publicly-available pointCloud2rawMesh function [25]. The resulting vertices and faces are written in the PLY file.

Results and discussion

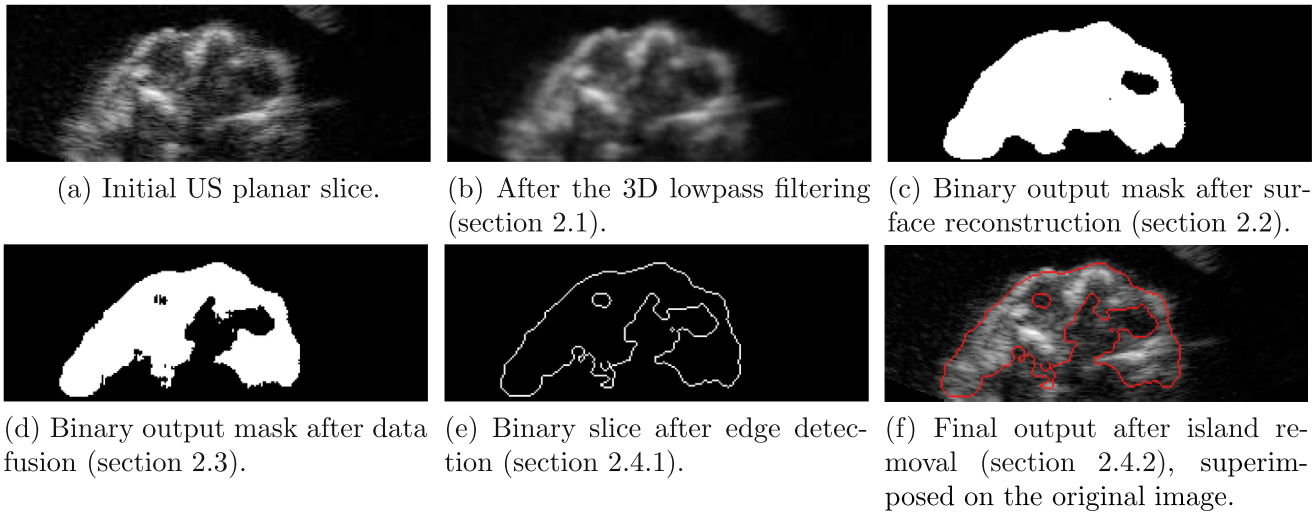
From September 2012 to January 2013, 30 three-dimensional volumes of 30 fetuses at 22—32 weeks' gestation were acquired. Written consent was obtained from the parents for publication of clinical details, clinical images, and videos. Principles outlined in the Declaration of Helsinki have been followed.



(a) Initial US planar slice. (b) After the 3D lowpass filtering (section 2.1). (c) Binary output mask after surface reconstruction (section 2.2).

(d) Binary output mask after data fusion (section 2.3). (e) Binary slice after edge detection (section 2.4.1). (f) Final output after island removal (section 2.4.2), superimposed on the original image.

Figure 9 Example of the output of the different steps. In this case, the fetuses face is completely laid on the mother's tissue (clearly visible on the right part of (a)), and the segmentation does not work as expected.



(a) Initial US planar slice. (b) After the 3D lowpass filtering (section 2.1). (c) Binary output mask after surface reconstruction (section 2.2).

(d) Binary output mask after data fusion (section 2.3). (e) Binary slice after edge detection (section 2.4.1). (f) Final output after island removal (section 2.4.2), superimposed on the original image.

Figure 10 Example of successful application of the algorithm.

Among these acquisitions, five were selected and processed for the purposes of the study. The remaining ones were excluded due to high noise or practical "inconveniences" such as baby's hands on face, or simply too much inaccuracy in the scanning process.

The ultrasound equipment was a Voluson system (GE Healthcare, Wauwatosa, WI, USA), with a RAB 4—8 (real time 4D convex transducer probe). The GE RAB 4—8 has a frequency range of 4 MHz to 8 MHz and is used for OB applications (Footprint 63.6×37.8 mm, FOV 70°, V 85°×70°).

Indeed, the algorithm has been tested on five 3D US scans. A rendered example of the resulting 3D model is shown in Figure 4. To show the clarity of the output, this model has also been 3D printed, as shown in Figure 5, proving how this technology might enable alternative business scenarios and applications, e.g., sales of the printed 3D fetus face. The intermediate steps are shown in Figures 6 to 10 (the rendered example corresponds to that of Figure 6), reporting a 2D slice in each phase for each of the five 3D US source scans. In all cases, the 2D slice corresponds to a transverse plane cutting the face at eye level, as shown by the dash-dotted line in Figure 5. The regions of interest are in the upper part of the surfaces where the nose is clearly visible; the rear part of the fetuses heads are instead full of noise (visible in the lower part of the images), and thus cannot be processed. However, please note that this does not affect the validity of the proposed method (e.g., it is visible also in the rendered examples shown in Figure 1), as it is due to the US scattering noise. Moreover, the surfaces of interest for the present work are those on the front part of the head (i.e., the face), as the final aim is to produce a large set of fetus face models for

the statistical pre-birth detection of face dysmorphic features [9].

Figure 6 shows the segmentation process; the effectiveness of all the steps is well-rendered. The validity of the proposed segmentation technique pointedly emerges in Figure 6c. The data fusion step removes noise from the regions of interest, i.e., in the back side, and the small island in the lower right corner of the slice is easily removed during post-processing.

In particular, the effectiveness of the data fusion step is self-evident in the example given in Figure 7. Here the main volume is properly extracted after the local 3D histogram procedure, but some of the mother's tissue is still visible on the right side of Figures 7a to 7c. On the contrary, the surface reconstruction from the other two perpendicular directions, together with the data fusion step, results in a complete elimination of all the unwanted tissue (Figure 7d).

The example in Figure 8 shows the importance of the island removal. Even though the dimensions are strongly reduced, two portions of unwanted tissue survive to the data fusion (in the upper left corner and in the center of the frame, as shown in Figure 8d). These two islands disappear after the last step; the suppression of the upper one is of utter importance, as it stands above the surface of interest. The only small island remaining, in the lower left corner in Figure 8f, survives to this phase because it is connected to the fetus in other slices. In this sense, it is important to underline the strong connection between the data fusion and the island removal. As seen in this example, the reduction of the noisy effects obtained by the former might be crucial for the success of the latter.

In Figure 9, the algorithm does not perform as expected. In this case the face is completely laid on the mother's tissue, as can be seen in Figure 9a, and the algorithm is not able to cope with this situation. In fact, the mother's tissue is segmented together with the desired one and, even though the profile is still evident, the extracted surface is less meaningful. However, with a reduced manual post-processing, the actual surface might still be extracted. In addition, we notice that in this case the island removal also fails, although an island is distinctly visible in Figure 9e; this is due to the connection of that portion of the mother's tissue with the fetuses tissue in the bottom slices. Finally, in Figure 10 another example of successful segmentation is reported.

The procedure is completely automatic, enabling fully unattended segmentation. This leads to the possibility of creating a large database of 3D facial point clouds, like the ones required for morphological studies through statistical analyses, without the necessity of human intervention. This kind of work has been already started [9]. Our tool is applied to build the database for clustering faces into healthy and unhealthy.

The algorithm works with standard protocols and file format (DICOM as input and any mesh file format as output), so it is completely platform-independent.

The algorithm is preliminary and might be refined, for instance by acting on the kernel shape or on the post-cleaning techniques (e.g., to obtain efficient cleaning also in situations like the one in Figure 9d). The same idea may also be extended to internal tissues segmentation to reach a complete 3D model of the fetuses face and of other organs (e.g., the heart). Furthermore, it can be extended to other applications of US imaging and for working with different 3D imaging techniques (CT or MRI), taking advantage, in this case, of the absence of the speckle noise effect.

Conclusions

An algorithm for facial automatic segmentation from 3D fetal ultrasound scans has been developed. The stack of 2D images is initially pre-processed, to remove speckle effects for a more assisted segmentation. This is achieved through a standard 3D Gaussian low-pass filtering.

The second step consists of the actual segmentation, which is performed with a new two-steps statistical processing, based on both 3D (volumetric) histogram processing and 2D (per-slice) global thresholding. In the first stage, an intermediate 3D structure is created, containing a local threshold chosen on the basis of the histogram of a 3D kernel. The histogram of each slice of this structure is then computed to find a global gray-scale threshold per each slice. The whole procedure is carried out three times, "cutting" the input slices among the three coordinate axes; the three binary masks are then fused together to reduce the number of false positives.

The surface point cloud is extracted from the fused binary stack through edge detection. Then, it is further "cleaned" by removing all the isolated edges not belonging to the tissue of interest. Finally, the cloud is meshed using the Poisson surface extraction procedure and saved in any standard mesh format (.ply in our case).

The whole procedure can be executed unattendedly, allowing the segmentation of large batches of 3D US scans.

References

[1] Levaillant JM. Intérêt de l'echographie 3D-4D en échographie fœtale et gynécologique: principes et indications. Journal de radiologie 2006;87(12-C2):1969—92.
[2] Vezzetti E, Marcolin F. Geometrical descriptors for human face morphological analysis and recognition. Rob Auton Syst 2012;60(6):928—39.
[3] Vezzetti E, Marcolin F. 3D human face description: landmarks measures and geometrical features. Image Vis Comput 2012; 30(10):698—712.
[4] Vezzetti E, Marcolin F. Geometry-based 3D face morphology analysis: soft-tissue landmark formalization. Multimed Tools Appl 2012:1—35.
[5] Vezzetti E, Moos S, Marcolin F, et al. A pose-independent method for 3D face landmark formalization. Comput Methods Programs Biomed 2012;198(3):1078—96.
[6] Vezzetti E, Marcolin F, Stola V. 3D human face soft tissues landmarking method: an advanced approach. Comput Ind 2013;64(9):1326—54.
[7] Vezzetti E, Marcolin F. 3D landmarking in multiexpression face analysis: a preliminary study on eyebrows and mouth. Aesthetic Plast Surg 2014;38:796—811.
[8] Vezzetti E, Speranza D, Marcolin F, et al. Exploiting 3D ultrasound for fetal diagnosis purpose through facial landmarking. Acta Stereol 2014;33(3):167—88.

[9] Conti D, Bonacina L, Froio A et al. Landmarking-based unsupervised clustering of human faces manifesting facial dismorphisms: the labio-schisis case. *Artif Intell Med* under review.
[10] Rousian M, Koning AHJ, van Oppenraaij RHF, et al. An innovative virtual reality technique for automated human embryonic volume measurements. Hum Reprod 2010;25(9):2210—6.
[11] Li X, Liu DC. Ultrasound speckle reduction based on image segmentation and diffused region growing. In *Proceedings of the 11th Joint Conference on Information Sciences*.
[12] Sarti A, Corsi C, Mazzini E, et al. Maximum likelihood segmentation of ultrasound images with Rayleigh distribution. IEEE Trans Ultrason Ferroelectr Freq Control 2005;52(6):947—60.
[13] Anquez J, Angelini ED, Grandé G, et al. Automatic segmentation of antenatal 3-d ultrasound images. IEEE Trans Biomed Eng 2013;60(5):1388—400.
[14] Feng S, Zhou SK, Good S et al. Automatic fetal face detection from ultrasound volumes via learning 3D and 2D information. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, pp. 2488—2495.
[15] GE. 4Dview, 2014. URL http://www.volusonclub.net/emea/4dview.
[16] NEMA. Digital Imaging and Communications in Medicine (DICOM). Rosslyn, VA, USA: NEMA Publications PS 3.1—PS 3.12; 2006.
[17] Bourke P. PLY — polygon file format. 2007. http://paulbourke.net/dataformats/ply/.
[18] Fenster A, Downey DB. 3D ultrasound imaging: a review. IEEE Eng Med Biol Mag 1996;15(6):41—51.
[19] Michailovich OV, Tannenbaum A. Despeckling of medical ultrasound images. IEEE Trans Ultrason Ferroelectr Freq Control 2006;53(1):64—78.
[20] Lorensen WE, Cline HE. Marching cubes: A high resolution 3D surface construction algorithm. In *ACM SIGGRAPH computer graphics*, volume 21. ACM, pp. 163—169.
[21] Sahoo PK, Soltani SAKC, Wong AKC, et al. A survey of thresholding techniques. Computer Vision, Graphics, and Image Processing 1988;41(2):233—60.
[22] Varshney PK. Distributed detection and data fusion. Secaucus, NJ: Springer-Verlag; 1996.
[23] Marr D, Hildreth E. Theory of edge detection. Proc R Soc Lond B Biol Sci 1980;207(1167):187—217.
[24] Kazhdan M, Bolitho M, Hoppe H. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, 7.
[25] Mian AS. Poisson surface reconstruction function. 2015. http://www.csse.uwa.edu.au/~ajmal/code.html.
[26] Cignoni P, Corsini M, Ranzuglia G. MeshLab: an open-source 3D mesh processing system. Ercim News 2008;73:45—6. http://meshlab.sourceforge.net.

Appendix A. Volume extraction algorithm

input: I (3D image array), l (proportional to kernel dimension), s (threshold), p (percentile for threshold selection)

for each slice k **do**
  for each i voxel in the kth slice **do** % *volumetric processing*
    if $I(i) \geq s$ **then** % *voxel is "bright enough"*
      apply the cubic kernel of side l centered in voxel i, as described in section 2.2.1;
      extract the voxels belonging to the main diagonals of the kernel;
      compute the histogram of this subset of voxels;
      identify the $(100-p)$th percentile of the histogram;
      set the ith local threshold equal to the percentile above;
    end if
  end for
  % *slice processing*
  set all local thresholds $< s$ equal to 0;
  compute the histogram of the kth threshold slice;
  estimate the histogram standard deviation σ;
  identify the highest value M appearing at least once;
  set the kth global threshold as $M - \left\lfloor \sigma \left(2^N - 1\right) \right\rceil$;
  create the binary output slice, putting ones in the pixels above threshold and zeros otherwise;
end for

Appendix B. Island removal algorithm

input: I (3D binary fused stack), r (number of connected components to keep), c (connectivity of the components to keep)

find all the 3D c-connected components in I;
for each connected component CC **do**
  find the bounding box $BB(CC)$;
  compute the volume of $BB(CC)$;
end for
set to `false` all the voxels not belonging to the r connected components with the largest bounding boxes;