EG UK Theory and Practice of Computer Graphics (2010)
John Collomosse, Ian Grimstead (Editors)

3D Modelling of Complex Biological Structures: The Oviduct

M. Burkitt[1,2], D. M. Romano[1], D. C. Walker[1] and A. Fazeli[2]

[1]Department of Computer Science, University of Sheffield, UK
[2]Academic Unit of Reproductive and Developmental Medicine, University of Sheffield, UK

Abstract

A novel technique using a particle system constrained by Newtonian forces is presented for the algorithmic construction of small scale, complex 3D biological structures based on real world biological data. This allows models of structures too small to be accurately recreated using medical imaging technologies such as Magnetic Resonance Imaging (MRI) to be created. The resulting model provides a geometrically realistic 3D environment which can be used to study the biological interactions which occur within. The technique is used to create a model of an oviduct, but could also be applied to similar organs such as the colon. The model is validated using measurements and visual comparisons from biological data. Finally, the technique is implemented using single-core and multi-core CPU techniques and using GPU acceleration. The performance of each implementation is then compared.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling

1. Introduction

The mammalian oviduct, called the fallopian tube in humans, is a section of the female reproductive system which connects the uterus to the ovary. The tube can be split into three main regions, the Isthmus, Ampulla and Infundibulum, as shown in Figure 1. The oviduct is the site of fertilisation, and recent studies have highlighted the importance of interactions between the soft tissue folds of the oviductal walls, the sperm and egg with regards to successful fertilisation, early embryo development, and problems during pregnancy [FP08].

The oviduct itself is a muscular tube containing interweaving folds of soft tissue. In mammals such as the pig, the tube is several centimetres long, only a few millimetres wide and has a complex internal structure only visible using a microscope. Figures 2(a) and 2(b) show two cross sectional slides from different regions of the oviduct taken from a pig, and Figure 2(c) shows a photograph of the real tissue. Reproductive biologists commonly conceptualise the oviduct as a simple tube, failing to consider the impact that such a complex internal structure might have on the interactions that occur inside. The creation of a 3D model provides an environment that can be used to understand the real structure of the organ, and allows computational experiments of the biological interactions within the oviduct to be performed.

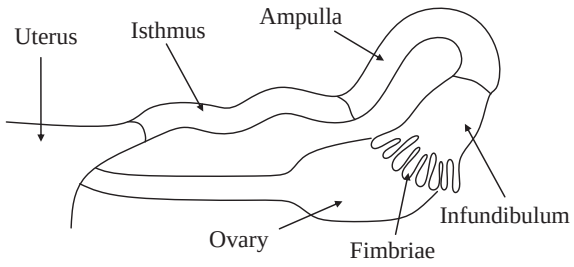
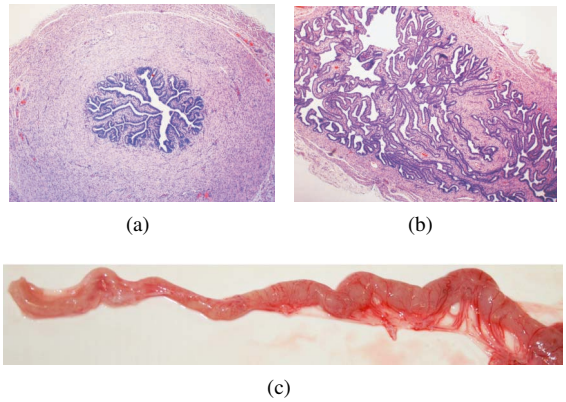


Figure 1: *The female reproductive tract. The oviduct consists of the isthmus and ampulla regions as shown.*

Here we present a novel technique to generate an accurately scaled 3D model of the internal structure of an oviduct using a force based particle system. A simplification of the same technique is used to approximate the external structure. As no two real world oviducts are the same, this method provides a more useful representation of an oviduct when compared to an exact copy, as would be achieved with the other techniques. This technique could also be adapted to create other soft tissue structures, such as the intestines.

DOI: 10.2312/LocalChapterEvents/TPCG/TPCG10/255-262



(a) (b)

(c)

Figure 2: *Shows how the complexity of the internal structure of the pig oviduct increases from the (a) Isthmus region to (b) the Ampulla region. The relative length and width of the pig oviduct is shown in (c).*

2. Related Work

Large organs such as the heart and lungs are routinely recreated in 3D using medical imaging techniques such as Magnetic Resonance Imaging (MRI), Computed Tomography (CT) and 3D Ultrasound [LCM*08]. These scanning techniques allow accurate measurements of biological entities to be taken while the subject is still alive, and are most successful on organs which are made out of muscle or thick tissue. The oviduct however is made almost entirely of soft tissue, and is very small in relation to other organs in the body. In this case, the contrast and detail obtained from medical imaging techniques is usually poor, and the resolution of the scanner is insufficient to capture the fine detail of its structure [HT02, PKB*05].

Another widely used approach to recreate models of internal organs is from histology slides using edge detection [ARD*06, QPCJ07]. This involves fixing the tissue in a wax block, cutting it into thin slices only a few micrometres thick, mounting them onto slides, staining them and then photographing them. After this process, each image has to be processed using edge detection, and then algorithmically reconstructed. This is possible for such small and soft organs, however for larger mammals, tens of thousands of slices are produced, making this a manually intensive process. The resultant model is dependent on the quality and frequency of the slides, and the capabilities of the chosen edge detection method. This technique could be used for the external structure of the oviduct, but the large variations of the internal structure make the techniques even more labour intensive.

Due to the success of the aforementioned imaging techniques for most biological systems, procedural reconstruction of organs within the body has received little attention. Work has been carried out on the modelling of deformable objects, the main focus of which has been on how the surfaces respond to external forces, self collisions and cutting in relation to surgical simulations [BS03, RCFC03, NMK*06]. With regard to creating 3D representations of internal organs, the main focus is on the external surface in relation to surgical planning [MDH*03, NMK*06].

Procedural techniques based on medical imaging have also been investigated. A skeletal pathway for blood vessels or other tubular structure has been identified from CT data, with both the internal and external form determined procedurally based on that pathway [GZ03]. 3D deformable organisms which follow the path of tubular structures in CT data have been used to create a 3D mesh of that structure [MH06].

Procedural techniques have also been used to generate models of biological systems where the complexity and scale mean that medical imaging techniques are not viable. The branching of tubular networks is a common structure in biological systems, and several procedural methods have been developed to recreate this complex structure. The airway structure of the lungs has been successfully modelled in 3D using a fractal based procedural method [KTS99]. A similar method has since been used to model the flow of blood in the circulatory system [GRK05]. The tubular structure of blood vessels have been 'grown' using a technique based on cell division [Smi03].

Originally proposed to model objects without solid, well defined boundaries [Ree83], particle systems have since been used to simulate a wide range of phenomena in 3D applications such as fire, smoke, water and other similar special effects. Particle systems have been used to create 3D objects with solid boundaries. The draping of cloth and clothing has been simulated using a particle system controlled using a 2D mass spring model [Bre00]. A technique using self-orienting particles has been used to construct dynamic surfaces instead of solid objects [ST92].

Particle systems have also been used to model biological systems. The elastic properties of muscle tissue has been modelled using particle system in combination with a 2D mass spring system [NT98]. A particle system modelling the surface of deformable objects with elastic properties has been used to simulate the deformation of biological organs such as the heart [SET02]. Multiple layers of particles with different properties have been used to provide a more realistic simulation of 3D tissue [AS00].

Although traditionally implemented on the CPU, more recent implementations of particle systems run on the GPU [KKKW05]. Several recent studies have demonstrated high performance, real time processing using the GPU for physics simulations and other computationally intensive calculations [CLT07, Har07].

A force based N-Body simulation that represents 16,384 particles and runs over 50 times faster than a comparable CPU has been run using the GPU [NHP07]. A 16,384

rigid body collision simulation using chess pieces which ran in real time on modern hardware has also been implemented [Har07].

3. Our Approach

The interweaving folds of soft tissue within the oviduct have a similar appearance to folds in a large piece of cloth, connected along one side to form a cylindrical structure. When compressed into a small space, cloth forms folds that wrap over each other, further deforming the shape of the structure. This is shown in Figure 3.

Figure 3: *Folds in a piece of cloth have a similar appearance to the oviductal folds.*

Due to the complexity and the overall length of the oviduct in proportion to the size of individual folds, it is not practical to simulate the entire surface as a single piece of cloth. Instead, a series of 2D cross sections are generated procedurally, using a 2D particle physics simulation, and connected together to form the complete model, as will be shown in the sections below.

A closed loop of particles connected together in a 1D mass spring system is used to represent a single wall of the oviduct. This is analogous to a single thread in a cloth simulation [Bre00]. Two disconnected threads, one inside the other, are used to create a cross section for the oviduct. An example of this is shown in Figure 4, which should be visually compared with the real images of the organ in Figure 2.

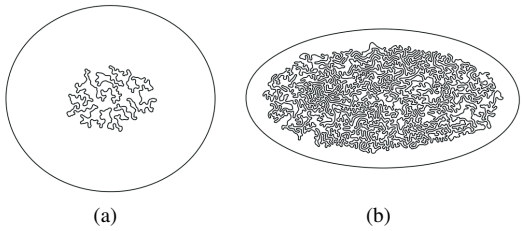


Figure 4: *Cross sectional images generated using our approach for (a) the Isthmus region and (b) the Ampulla region of the oviduct.*

The position of each particle in a thread is determined by evaluating the relative strengths of forces that act within the simulation. At any moment in time, the position in 2D space of each particle within both threads determines the cross section at a specific position along the oviduct.

Below is a brief description of the different types of force applied to the inner loop during the simulation. The type of force and the way it is used is domain specific, and has been tailored for the creation of an oviduct. If applied to a different structure, the force requirements are likely to be different and need to be determined empirically or by research.

A spring force F_s, which is based on Hooke's law of elasticity, is used to maintain a connection between two adjacent particles. This force is defined as

$$F_s = -((k \times (x - r)) + (c \times v))$$

where k is the spring strength, x is the current distance between the two particles, r is the rest length (the distance between particles where no force is applied), c is the damping strength, and v the current velocity of the connected particles. The influence that this force has on adjacent particles is shown in Figure 6(a).

A containment force is used to constrain particles to a specific region. For the purposes of modelling the oviduct, the containment region is defined with a position, outer radius, inner radius and squash amount. The squash amount modifies the radius along the Y axis to squash the containment region as shown in Figure 5. This is used to maintain the overall shape of the oviduct, and only particles in-between the inner and outer radius are influenced. This force F_n is defined as

$$F_n = d \times k$$

where d is the direction of the centre of the containment region and k is the strength of the force strength k. The influence that this force has on particles within the containment region is shown in Figure 6(c).

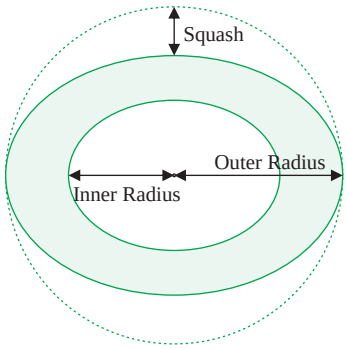


Figure 5: *The containment region used to control the structure of the model.*

In combination with the containment force, an additional force called the regulatory force is applied to each particle

to maintain a link between the soft tissue folds and the inner wall of the oviduct. This force F_g is defined as

$$F_g = d \times k$$

where d is the direction of the nearest point on the inner radius ellipse when the particle was initially created and k is the strength of the force. This ensures that individual particles remain in a realistic proximity to the oviductal wall throughout the simulation, as shown in Figure 6(d).

In order to simulate the random contortions of the folds, an additional force, called the contortion force is applied to a small percentage of the particles, causing their positions to change randomly. This force F_c is defined as

$$F_c = d \times \frac{\theta}{180} \times m \times c$$

where d is the direction of the contortion force, which is the normal vector between both adjacent particles, θ is the angle between the current particle and both adjacent particles, which has a range of [0, 180], m is the magnitude of the force and c is a global strength modifier that fluctuates repeatedly through the range [-1, 1] as the simulation progresses. Multiplying the force by θ and dividing by 180 ensures that the strength of the contortional force is relative to the angle between the particle and adjacent particles. This means that a strong force is applied when the angle is close to 180°, when the particles form a straight line, and a weaker force is applied as the angle approaches 0°, when the particles are at a sharp angle. The influence that this force has on individual particles is shown in Figure 6(b).

A simple frictional force F_r which is applied as a constant damping force to all particles is defined as

$$F_r = -v \times k$$

where v is the current velocity of the particle and k is the friction strength, which has a range of [0, 1]. This serves to prevent forces from becoming too large, and destabilising the simulation.

Self intersection is where the loop of particles crosses over itself. Robust prevention of self intersections can be computationally intensive, so a simple, force based method was used. A comparison is made between every particle in the loop, and every pair of particles in the same loop. If a particle moves too close to another particle, or a line connecting two adjacent particles, then a force is applied to that particle to move it in the opposite direction, as shown in Figure 6(e).

For each particle, the total force is calculated and then used to update the current velocity, which in turn determines the new position. Several numerical methods exist for performing this type of calculation. The simplest approach, called the Euler method performs a single set of calculations per particle per time step. This method is fast, but is can result in large fluctuations in values, reducing the accuracy and stability of the simulation. An alternative approach, called

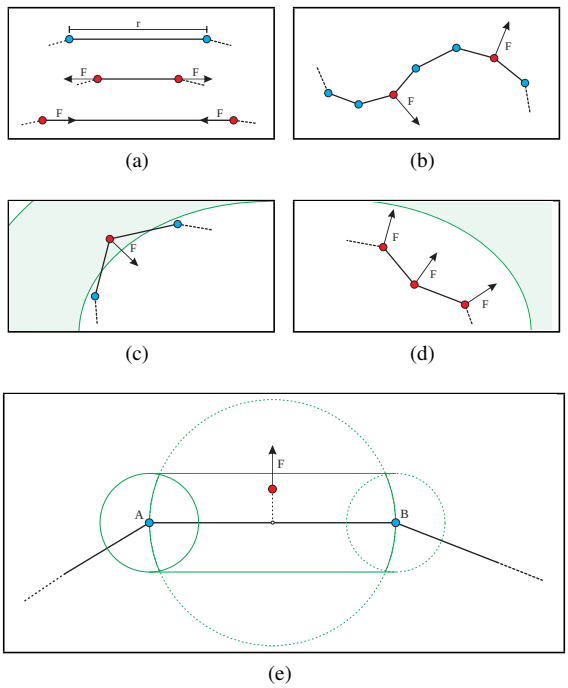


Figure 6: *The force F applied to each particle depending on the type of force being processed. (a) Spring Force, if the distance between particles equals the rest length r, then no force is applied. (b) Contortion Force, particles move relative to the position of adjacent particles. (c) Containment Force, particles within the containment region are pushed towards the centre. (d) Regulatory Force, particles are pushed towards the containment force to maintain structural realism. (e) Self Intersection Force, particles are pushed away from opposing particles, and the line connecting them.*

the Runge-Kutta 4th order method, was chosen for this simulation. This approach increases the simulation time by a factor of four, but also significantly increases accuracy and stability.

Two sets of parameters need to be defined before the simulation can run. The first set defines the global parameters, which determine the relative scale of the model in relation to the position of the particles on screen, the total length of the model and the number of time-steps that the simulation should run for. The second set of parameters are properties of the different forces that act within the simulation. Before the simulation begins, the initial value and the manner in which each parameter changes is determined. The internal structure of the oviduct increases in complexity along the length. To allow this to happen, the number of particles also increases over time. The parameters used to define the forces within the system fluctuate to cause distortions in folds, and are based on measurements taken from histology slides. The

length and number of particles was also determined from histology slides as taken from different parts of the oviduct.

3.1. Implementation

Three versions of the simulation were implemented. The first to run on a single core CPU, the second to take advantage of multi-core abilities of modern CPUs and the third to run on a commercial Graphics Card (GPU). The CPU versions were both written to use efficient data structures to improve performance, and the multi-core CPU implementation was written to take advantage of multi-core processors, so the forces calculations of each particle are evenly distributed amongst the cores. The GPU version was written to take advantage of the massively parallel processing power of a graphics card, By mapping arrays of data to textures, the data was then manipulated using shader programs in CG programming language. Although CUDA could potentially provide superior performance, we wanted the system to work on all modern commercial graphics card, where as CUDA is limited to NVidia cards. Where possible, the CPU and GPU versions were implemented in the same way, to ensure consistency and to make for a fair comparison.

As the simulation progresses, the position of all particles is recorded at fixed intervals, resulting in a series of cross sections, a selection of which are shown in Figure 7(a). Once a complete set of cross sections has been generated, the complete 3D mesh can be constructed. The cross sections are lined up in 3D space, the distance between them is determined by the length of the oviduct being generated and the number of cross sections available. As the changes between cross sections are gradual and sequential, it is possible to generate triangle strips in-between cross sections, which reduces the number of vertices being processed per triangle, greatly increasing mesh efficiency, as shown in Figure 8(b).

The generation process results in two disconnected meshes, one for the external wall and one for the internal structure. In order to create a whole 3D object, the two meshes need to be connected using end capping techniques. Due to the complexity of the internal structure of the cross sections, the Fast Industrial-Strength Triangulation of Polygons (FIST) [Hel01] method was used, the results of which are shown in Figure 8(a). To generate the final model, the simulation was run for 2000 time steps with positions recorded every 10 steps, resulting in 201 cross sections. Once generated, the final mesh contained $790,211$ vertices and $1,1514,538$ triangles.

4. Evaluation

4.1. Geometric Validation

As the model was generated by specifying measurements taken from the histology slides the overall scale of the model is accurate. This accuracy applies to the overall length; the width and height; the amount the structure is squashed for both the internal and external walls; and the area covered by the internal folds. However, it does not guarantee that the internal folds are of an accurate scale. This can only be demonstrated through a comparison between the histology slides and the mesh cross sections. A visual comparison between progressive sections of the oviduct and the generated cross sections is shown in Figure 7.

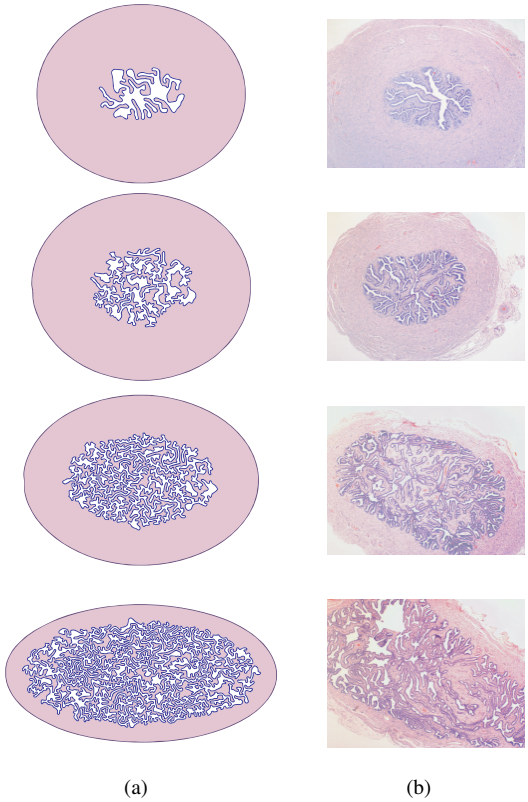


Figure 7: *Comparison of progressive cross sections of the oviduct between (a) the generated cross section and (b) the corresponding histology image.*

Figure 7(a) shows the generated cross sections and Figure 7(b) shows the corresponding histology image at a similar position within the real organ. Similarities between the scale of the folds and the way they wrap around each other are immediately obvious comparing in Figures 7(a) and 7(b). Long, thin strands are observable in both sets of images, as well as wider and narrower areas of both tissue and the gap in-between tissue. There are however several differences between the two. The histology slides have large areas of thick tissue and areas with large gaps between tissues, whereas the generated images only have relatively small variations in thickness. This discrepancy could be due to anomalies in the histology slides. The gap between folds in some areas of

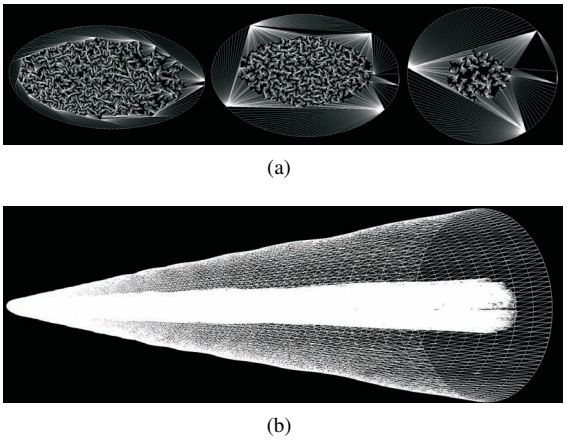


Figure 8: *(a) End Caps generated using FIST, (b) Wire-frame view of the complete mesh.*

the histology images are very small, with the walls touching in parts. In the generated images, there is a constant, distinct gap between folds. Although random, there is a distinct uniformity to the changes in the generated images, where as the histology images appear to be completely random. These differences could be addressed by varying randomly the parameters that control the forces acting on the structure.

4.2. Overall Appearance

Figures 9(a) and 9(b) show the isthmus and ampulla ends of the generated model. Figure 9(c) shows the complete length of the model, and when visually compared with the real image in Figure 2(c), the similarities in scale between the biological tissue and the final model are also apparent. A lengthwise cross-section from one segment of the ampulla region was created, which shows the complexity of the internal folds, as shown in Figure 9(d). Figure 9(e) shows a high quality render of the completed model, with procedural texturing and bump mapping for visual effect.

4.3. Performance Comparison

The simulation was run three times on each version using identical parameters using the following configuration:

- Windows 7 x64
- Intel Core 2 Quad Q6700 @2.66Ghz
- 4GB RAM
- NVidia GeForce GTX 280

Each simulation ran for 2000 time steps, and the total number of particles gradually increased from 710 to 8300 as the simulation progressed. The results are shown in Figure 10(a). Due to the vast difference in time between the CPU and GPU implementations, these results are shown in Figure 10(b) using a logarithmic scale. The logarithmic view

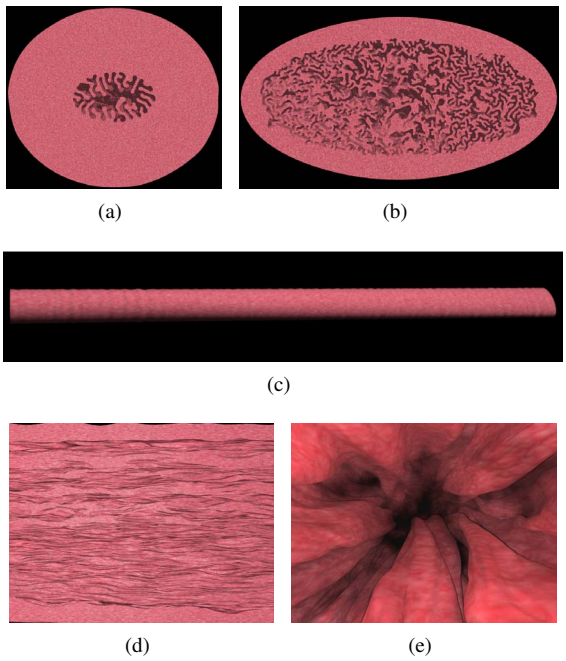
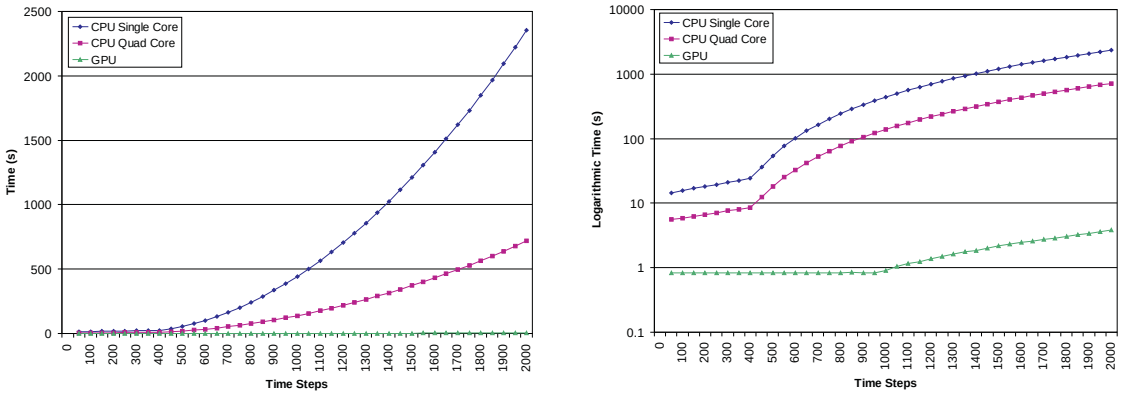


Figure 9: *(a) Viewed from the isthmus end. (b) Viewed from the ampulla end. (d) Lengthways cross-section showing the internal folds. (e) High quality image rendered using procedural texturing and bump mapping.*

of the results shows a sharp change in the gradient of the curve for both the single-core and multi-core CPU implementations at 400-450 time-steps. This corresponds to the point where the Isthmus joins the Ampulla region of the oviduct, and the complexity of the internal structure, and therefore the number of particles, increases faster. For the GPU implementations, the gradient of the curve begins to change at 1000-1050 time-steps. This is the point where the ratio between the number of calculations performed by the GPU and it's parallel capacity becomes so high that the calculation time becomes significantly large.

The table in Figure 11 shows the total running times for each simulation. The multi-core CPU version is 3.25 times faster than the single-core CPU version, however, the GPU version was significantly faster than both CPU versions, with the complete simulation running 143 times faster than the multi-core CPU version and 463 times faster than the single-core CPU version.

The main advantage of our approach is that it provides a unique representation of an oviduct every time the model is created, similarly to what happens in nature. Furthermore, the dynamic nature of our solution means that it is far more flexible than the exact reconstruction techniques from medical imaging. The structure of the oviduct is constantly changing, the folds move with muscular contractions, and vary



(a) (b)

Figure 10: *The relative performance of each implementation as the complexity of the simulation increases. (a) The linear performance difference. (b) The logarithmic performance difference.*

in complexity during the different stages of the menstrual cycle. By varying dynamically the parameters used within the simulation the model can represent these different conditions. The use of GPU implementation allows the creation of a large number of models within a short space of time.

Version	Running Time (hh:mm:ss)
Single-Core CPU	08:03:27.491
Quad-Core CPU	02:28:52.576
GPU	00:01:02.646

Figure 11: *Table shows the total time taken to generate a complete model for each implementation.*

There are however several limitations. The complexity of the internal structure, and the gaps in between folds, are even in the simulated model. To more accurately represent nature, the simulation needs to permit more variation in the thickness and uniformity of the tissue and the gaps. Better results could be achieved using more particles. The current limit is due to the maximum width of the frame buffer on the used GPU. This could also be overcome by implementing the system using CUDA. An alternative to this is to change the implementation of some of the forces.

Islands, which are observable in the histology slides caused by the folds of the oviduct moving down the Z axis, cannot be represented using this approach. The generated model is straight, whereas a real oviduct has bends and curves in the overall structure. Binding the structure to a deformable spline would be one way to resolve this, but care would need to be taken to prevent self intersections. The current collision detection strategy used to prevent self intersections is not very efficient. One possible optimisation would be to perform the calculations as a two step process. The first step to perform the initial collision test, the second pass to be performed only on pairs that can collide. To improve the realism of the final mesh, post-processing operations such as surface subdivision could be applied. This would removed any sharp angles, increasing the realism.

5. Conclusion

We presented a technique for generating 3D representations of the internal structure of a small scale, complex, soft tissue organ, and created a 3D model of an oviduct using this technique. This approach allows oviducts from different species and of different sizes and complexity to be recreated based on measurements taken from histology slides. The resulting model is an accurately scaled 3D representation of a complex internal biological structure, which is optimised for efficient rendering. The algorithm has been implemented on the CPU, quad-core CPU and GPU, and the GPU version has demonstrated a 143 times performance increase of the quad-core CPU and 463 times increase compared to the single-core CPU implementation.

This approach could be used to generate similar soft tissue organs, such as the bowels. Also by changing the bounding forces, soft tissue folds can be created to be bound to any region with a mathematically definable intersection test, allowing the modelling of the fine detailed structure of many larger organs, such as the heart, which are lined with small scale soft tissue folds.

References

[ARD*06] ARCHIP N., ROHLING R., DESSENNE V., ERARD P.-J., NOLTE L. P.: Anatomical structure mod-

eling from medical images. *Computer Methods and Programs in Biomedicine 82*, 3 (June 2006), 203–215. 2

[AS00] AMRANI M., SHARIAT B.: Deformable organ modeling with multi-layer particle systems. *Information Visualisation, International Conference on 0* (2000), 351+. 2

[Bre00] BREEN D. E.: A survey of cloth modeling methods. In *Cloth modeling and animation*, House D. H., Breen D. E., (Eds.). A. K. Peters, Ltd., Natick, MA, USA, 2000, pp. 19–53. 2, 3

[BS03] BALANIUK R., SALISBURY K.: Soft-tissue simulation using the radial elements method. In *Proceedings of International Symposium on Surgery Simulation and Soft Tissue Modeling*, Ayache N., Delingette H., (Eds.). 2003, pp. 48–58. 2

[CLT07] CRANE K., LLAMAS I., TARIQ S.: Real-time simulation and rendering of 3d fluids. In *GPU Gems 3*, Nguyen H., (Ed.). Addison Wesley Professional, August 2007, ch. 30. 2

[FP08] FAZELI A., PEWSEY E.: Maternal communication with gametes and embryos: a complex interactome. *Brief Funct Genomic Proteomic* (February 2008), eln006+. 1

[GRK05] GABRYS E., RYBACZUK M., KEDZIA A.: Fractal models of circulatory system. symmetrical and asymmetrical approach comparison. *Chaos, Solitons & Fractals 24*, 3 (May 2005), 707–715. 2

[GZ03] GIACHETTI A., ZANETTI G.: 3d reconstruction of large tubular geometries from ct data. 2003, p. 1000. 2

[Har07] HARADA T.: Real-time rigid body simulation on gpus. In *GPU Gems 3*, Nguyen H., (Ed.). Addison Wesley Professional, August 2007, ch. 29. 2, 3

[Hel01] HELD M.: Fist: Fast industrial-strength triangulation of polygons. *Algorithmica 30*, 4 (2001), 563–596. 5

[HT02] HOLDSWORTH D. W., THORNTON M. M.: Micro-ct in small animal and specimen imaging. *Trends in Biotechnology 20*, 8 (August 2002), S34–S39. 2

[KKKW05] KRUGER J., KIPFER P., KONCLRATIEVA P., WESTERMANN R.: A particle system for interactive visualization of 3d flows. *Visualization and Computer Graphics, IEEE Transactions on 11*, 6 (2005), 744–756. 2

[KTS99] KITAOKA H., TAKAKI R., SUKI B.: A three-dimensional model of the human airway tree. *J Appl Physiol 87*, 6 (December 1999), 2207–2217. 2

[LCM*08] LI G., CITRIN D., MILLER R. W., CAMPHAUSEN K., MUELLER B., MYCHALCZAK B., SONG Y.: 3d and 4d medical image registration combined with image segmentation and visualization. In *Encyclopaedia of Healthcare Information Systems*, Wickramasinghe N., Geisler E., (Eds.). IGI Global, Hershey, PA, 2008. 2

[MDH*03] MESEURE P., DAVANNE J., HILDE L., LENOIR J., FRANCE L., TRIQUET F., CHAILLOU C.: A physically-based virtual environment dedicated to surgical simulation. 2003, p. 1002. 2

[MH06] MCINTOSH C., HAMARNEH G.: Vessel crawlers: 3d physically-based deformable organisms for vasculature segmentation and analysis. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on* (2006), vol. 1, pp. 1084–1091. 2

[NHP07] NYLAND L., HARRIS M., PRINS J.: Fast n-body simulation with cuda. In *GPU Gems 3*, Nguyen H., (Ed.). Addison Wesley Professional, August 2007, ch. 31. 2

[NMK*06] NAKAO, MEGUMI, KURODA, TOMOHIRO, OYAMA, HIROSHI, SAKAGUCHI, GENICHI, KOMEDA, MASASHI: Physics-based simulation of surgical fields for preoperative strategic planning. *Journal of Medical Systems 30*, 5 (October 2006), 371–380. 2

[NT98] NEDEL L. P., THALMANN D.: Real time muscle deformations using mass-spring systems. In *CGI '98: Proceedings of the Computer Graphics International 1998* (Washington, DC, USA, 1998), IEEE Computer Society, pp. 156+. 2

[PKB*05] PODDAR A. H., KROL A., BEAUMONT J., PRICE R. L., SLAMANI M. A., FAWCETT J., SUBRAMANIAN A., COMAN I. L., LIPSON E. D., LIPSON E. D., FEIGLIN D. H.: Ultrahigh resolution 3d model of murine heart from micro-ct and serial confocal laser scanning microscopy images. In *Nuclear Science Symposium Conference Record, 2005 IEEE* (2005), vol. 5, pp. 2615–2617. 2

[QPCJ07] QIANG W., PAN Z., CHUN C., JIANJUN B.: Surface rendering for parallel slices of contours from medical imaging. *Computing in Science & Engineering 9*, 1 (2007), 32–37. 2

[RCFC03] RAGHUPATHI L., CANTIN V., FAURE F., CANI M.-P.: Real-time simulation of self-collisions for virtual intestinal surgery. 2003, pp. 1004–1005. 2

[Ree83] REEVES W. T.: Particle systems—a technique for modeling a class of fuzzy objects. *ACM Trans. Graph. 2*, 2 (April 1983), 91–108. 2

[SET02] STAHL D., EZQUERRA N., TURK G.: Bag-of-particles as a deformable model. In *VISSYM '02: Proceedings of the symposium on Data Visualisation 2002* (Aire-la-Ville, Switzerland, Switzerland, 2002), Eurographics Association, pp. 141–150. 2

[Smi03] SMIRNOV A.: Tool-assisted mesh generation based on a tissue-growth model. *Medical and Biological Engineering and Computing 41*, 4 (July 2003), 494–497. 2

[ST92] SZELISKI R., TONNESEN D.: Surface modeling with oriented particle systems. *SIGGRAPH Comput. Graph. 26*, 2 (1992), 185–194. 2