Parallel 3D Printing Based on Skeletal Remeshing

Kuo-Wei Chen[1] Chih-Yuan Yao[1] You-En Lin[2] Yu-Chi Lai[3]
National Taiwan University of Science and Technology,National Tsing Hua University

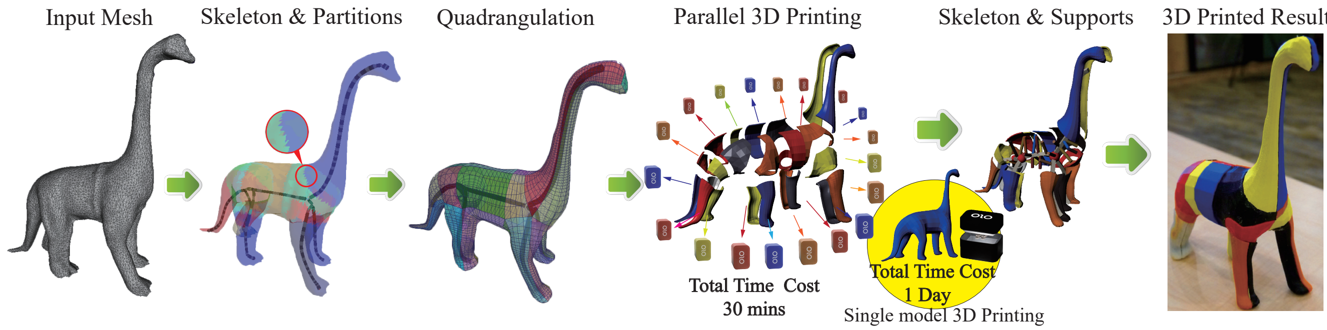


Figure 1: *Our System inputs a 3D model to calculates its skeleton and corresponding partitions. To get smoother boundaries for each partition, our system quadrangulation remeshes the model with an estimated surface flow field. We create printing shells by adding a default thickness to all partitions and then print them in parallel with multiple 3D printers. Finally, we can assemble the model by attaching shells onto the supported skeleton formed by bones and joints using our designed supporters.*

Keywords: 3D printing, skeleton, quadrangulation remeshing, field design

Concepts: •**Computing methodologies** → *Shape modeling;*

1 Introduction and Motivation

Although 3D printing is becoming more popular, but there are two major problem. The first is the slowness of the process because of requirement of processing information of an extra axis comparing to tradition 2D printers. The second is the printable dimension of 3D printers. Generally, the larger the model is printed, the larger a 3D printer has to be and the more expensive it is. Furthermore, it would also require a large amount of extra inflation materials. With the entrance of cheap 3D printers, such as OLO 3D printers [Inc. 2016], parallel printing with multiple cheap printers can possibly be the solution. In order to parallel print a 3D model, we must decompose a 3D model into smaller components. After printing out all the components, we assemble them together by attaching them to the skeleton through supporters and joints to form the final result. As shown in our results, our designed shell-and-bone-based model printing can not only save the printing time but also use lesser material than the original whole model printing.

2 Algorithm

Since our goal is to accelerate the 3D printing process of a model by parallel printing with multiple 3D printers, we must design a method to decompose a 3D model into small printable components and assemble the elements together to form the final result. When observing printed results, the interior of a 3D object does not affect the final appearance but only surfaces do. Therefore, we aim at segmenting an object into mesh patches and growing their thickness to become surface shells for the object. Our system can distribute these smaller and flatter shells to multiple printers for faster parallel printing. Additionally, the process consume less material because flatter structures require almost no supported structure. Later, we can use skeleton bones, joints, and skin supporters to hold the shells for the final result. In other words, our system decomposes the surface mesh into small components, skin shells, skeleton bones, joints, and supporters, to be parallel printed in multiple printers and assembles them for a complete model. Fig. 1 shows our decomposing and assembling process. First, the input to our system is a 3D surface triangular mesh, and our system apply mesh contraction [Au et al. 2008] by iteratively applying curvature-flow Laplacian smoothing on the model to estimate its skeleton of a model. Although Au's algorithm can construct a skeleton and partition corresponding mesh patches, bones are line instead of close surfaces and the partitioned mesh patches are rough for proper printing. Therefore, our system modify the thickness of the bones in the skeleton to ensure ease of printing and strength of supporting the weight of shells, and we need to ensure that the entire skeleton is still inside of the skin shells. We adjust the contraction weighting for proper bone growth after enlarge bone sizes.

Second, we must grow shells attached to these support skeleton bones for the final appearance of the model. Originally, Au's algorithm can also partition the model into mesh patches based on the estimated skeleton, the boundary of mesh patches are not smooth due to the triangular mesh structure. Generally, their algorithm label all triangles to one of the skeleton bones and this causes the rough boundaries of mesh patches because smooth partition boundaries should go through the boundary but their algorithm do not allow them to go through the triangles. Therefore, we design a remeshing algorithm to generate smooth partition boundaries. [Yao and Lee 2009; Usai et al. 2015] propose a quadrangulation algorithm using skeleton-based mesh partition. As shown in Fig. 2, our system adapts their quadrangulation concept to compute a flow field on the surface for a smoother boundary. According to the mesh partitions, when both joints of the mesh bone connect to only one bone, we place constraints on both sides of the partition and compute a

SIGGRAPH 2016 Posters, July 24-28, 2016, Anaheim, CA
ISBN: 978-1-4503-4371-8/16/07
DOI: http://dx.doi.org/10.1145/2945078.2945126

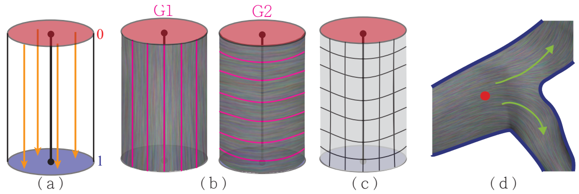


Figure 2: *(a) We compute the flow direction based on the skeleton and mesh partition estimated using the algorithm proposed by [Au et al. 2008]. (b) We define two directions: major and minor flow directions, based on being parallel/perpendicular to the flow direction of the estimated flow field and use both directions to generate G1 and G2 curves. (c) Our system remeshes the model using G1 and G2 curves. (d) While there are joints with more than two bones, the flow direction around the joint is chaotic. Therefore, our system adds extra constraints (red dot) along the sharing boundaries to have a proper surface flow field.*

		Time(min.)	Material.(g)
Dinosaur	Solid	1080	167.84
	Ours	36	115.09
BigBuddy	Solid	2998	546.01
	Ours	81	428.64

Table 1: *This shows the timing statistics of the 3D model printing and our shell-and-bone-based model printing. Our method increases the printing speed and reduces the inflation material.*

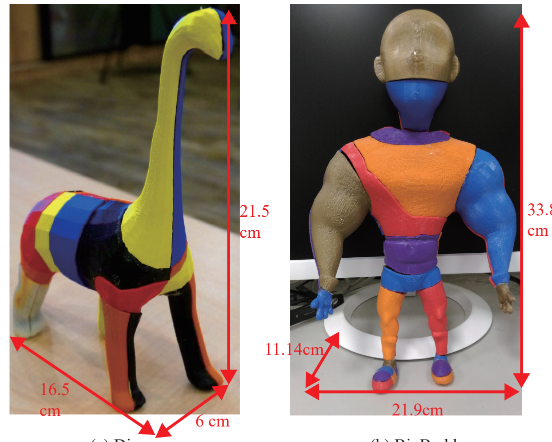
surface flow field using harmonious functions [Yao et al. 2012]. However, when a joint of the skeleton connects to more than two bones, the computed field become chaotic, and our system places extra constraints along the destined share boundaries of these connected patches to generate consistent flow fields for all partitions. After having the flow field, our system generates G1 curves by distributing particles uniformly along the boundaries and advecting them along the flow field and G2 curves by linking all points on G1 curves with an equal arc distance from their corresponding starting locations. In other words, G2 curves are perpendicular to the flow field. These G1 and G2 curves help use to quadrangulate the mesh patches for smooth boundaries of shells on the selected G2 curves. After remeshing, our system duplicates all mesh patches and move a distance of thickness, and then connected the boundary vertices to form the shell. In this way, we can print all shells at the same time to shorten the printing time of the entire model.

3 Results

Fig. 3 shows two printed results of our algorithm. Table 1 shows the timing statistics which demonstrates that our parallel printing method can accelerate the process and also saves a large amount of material. With the remeshing method, the boundary of each component is smooth to fit well to each other for easily assembling.

Acknowledgement

This work was supported by Grant MOST-104-2221-E-011-029-MY3, Grant MOST-103-2221-E-011-114-MY2, Grant MOST-104-2218-E-011-006, Grant MOST-103-2218-E-011-014, Grant MOST-104-2221-E-011-092 and Grant MOST-103-2221-E-011-076, Taiwan.



(a) Dinosaur (b) BigBuddy

Figure 3: *(a)This shows a 16.5cm × 6.0cm × 21.5cm dinosaur model. Our system decomposes the model into 23 components. When printing it in parallel, the process are 30 times faster than the original one. (b)This shows a 21.9cm × 33.8cm × 11.14cm human figure model. Our system decomposes it into 34 components. When printing it in parallel, the process are 37 times faster than the original one.*

References

AU, O. K.-C., TAI, C.-L., CHU, H.-K., COHEN-OR, D., AND LEE, T.-Y. 2008. Skeleton extraction by mesh contraction. *ACM Trans. Graph. 27*, 3, 44:1–44:10.

INC., O. D., 2016. Olo 3d printer http://www.olo3d.net/.

USAI, F., LIVESU, M., PUPPO, E., TARINI, M., AND SCATENI, R. 2015. Extraction of the quad layout of a triangle mesh guided by its curve-skeleton. *ACM Trans. on Graphics*, to appear. https://sites.google.com/site/marcolivesu/quadlayouts.

YAO, C.-Y., C. H.-K. J. T., AND LEE, T.-Y. 2009. Compatible quadrangulation by sketching. *Comp. Anim. Virtual Worlds 20*, 101–109.

YAO, C.-Y., CHI, M.-T., LEE, T.-Y., AND JU, T. 2012. Region-based line field design using harmonic functions. *IEEE Transactions on Visualization and Computer Graphics 18*, 6 (Jun), 902–913.