Pacific Graphics (2012)
C. Bregler, P. Sander, and M. Wimmer (Editors)

Short Papers

Optimizing Geometry-aware Pants Decomposition

Kang Zhang and Xin Li[†]

School of Electrical Engineering and Computer Science, Louisiana State University, USA.

Abstract

We present a computational framework to optimize the pants decomposition of surfaces with non-trivial topology. A pants decomposition segments a surface into a set of sub-patches each of which is genus-0 with 3 boundaries. A given surface usually admits infinitely-many pants decompositions that are topologically inequivalent. Given some pre-determined geometric criteria, our algorithm enumerates different classes of pants decompositions and search for the optimal one. The proposed framework is general, and can be used to generate different suitable segmentations according to different applications. We also generalize our algorithm for consistent decomposition of multiple surfaces. This algorithm can be used in constructing compatible cross-surface mapping, and facilitate many computer graphics tasks.

Keywords: Mesh Segmentation; Consistent Segmentation; Cross-surface Mapping.

1. Introduction

Surface segmentation is ubiquitous in geometric modeling. When the shape has complicated geometry or topology, partitioning it into several solvable sub-patches is effective for divide-and-conquer. To process multiple surfaces together, a *consistent decomposition* is usually needed. A consistent decomposition of two surfaces, for example, is two sets of sub-patches sharing a coherent adjacency relationship, namely, their dual graphs are isomorphic. Direct graphics applications of consistent decomposition include cross-surface parameterization, morphing, animation transfer/synthesis, and many others [KZW11, KS04, LBG*08, LGQ09]

Many existing surface segmentation algorithms (see the survey paper [Sha08]) are guided by local geometry and paying less attention to global shape topology and the topology of resultant decomposition. Local-geometry guided approaches can not easily provide uniform decomposition for shapes with complex topology; also, they often can not be easily generalized to consistently segment multiple surfaces. Consistent segmentation of multiple objects has been studied for partitioning multiple objects into similar salient parts. A popular scheme is to partition the surfaces into sets of topological-disks [KZW11, KS04, NGH04] by tracing shortest paths between feature points. This scheme has been effectively used in cross-surface mapping, where well-studied topological-disk parameterization can be computed and used to compose the global mapping. However, manual marker labeling is required, which is laborious and potentially unreliable. Part analogies [SSSCo08] segments each model into parts independently and then creates a distance measure between parts that takes into account both local shape signatures and the context of the parts within a hierarchical decomposition. A consistent segmentation is then created based on this catalog of parts with inter-part distances. Golovinskiy and Funkhouser [GF09] create segments and correspondences between objects simultaneously, so that it can better identify salient segments that are shared across the set of objects. Kraevoy et al. [VDA07] develop a modeling tool that partition two meshes into sub-patches consistently, these sub-patches are one-to-one corresponded and hence can be transplanted from one model onto the other. Unlike [GF09] that allows outlier segments, this algorithm rigorously partitions different models consistently. This coherency is necessary for applications such as cross-surface mapping, where outliers should not be allowed.

1.1. Pants Decomposition for Consistent Segmentation

Our algorithm is based on the computation of *pants decomposition* (PD) [VL07, HS94]. PD provides us an elegant topological tool to study the segmentation (and consistent segmentation) of high-genus surfaces systematically.

Definitions. A *pants decomposition* (PD) is a set of curve cycles that partitions a surface into disjoint *pants patches*.

[†] Email: xinli@cct.lsu.edu

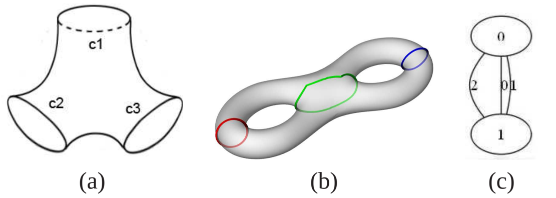

Figure 1: *A Pants Patch and Pants Decomposition. (a) A pants patch; (b) A pants decomposition on the two-torus. (c) Its corresponding PD-Graph.*

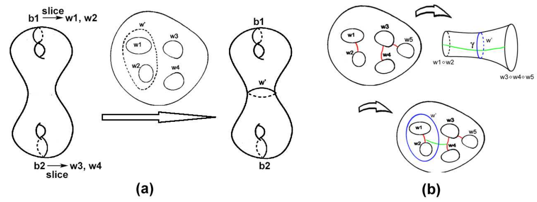

Figure 2: *(a) Computing Initial Pants Decomposition. (b) Computing the Homotopic Cycle.*

Each *pants patch* is a topological sphere with three boundaries (one "waist" and two "legs", see Fig. 1 for examples of a pants patch and pants decomposition). A genus-g surface patch with b boundaries is said to be of the type (g,b). A *maximal cut system* of M is a set C of simple and pairwise disjoint cycles $\{c_i\}$ that partitions M into sub-patches of type-$(0,3)$.

A surface M admits infinitely many pants decompositions. Two PDs are *homotopic* to each other if the cycles of the first decomposition can deform to the cycles of the second, without leaving the surface. We aim to develop an algorithm to traverse among different topological equivalence classes to search for the optimal pants decomposition.

1.2. Main Contributions

The main contributions of this work include:
(1) We develop a computation framework to traverse the homotopy classes of the pants decomposition for searching specific locally optimal PD.
(2) We demonstrates this PD computation framework by showing the integration of several useful geometric criteria in order obtain nice geometry-awareness.
(3) Based on the above tools, we design a reliable consistent PD framework for multiple surfaces. After $2g+b$ initial cycles are indexed, a unique, locally-optimal, consistent PD across multiple objects can be obtained. This consistent decomposition can be used for inter-surface mapping.

Compared with our previous PD algorithm [LGQ09] (in which only one pants decomposition is computed following an predetermined tracing order and is often not geometrically desirable), a key improvement of this work is that we can now traverse among different topological equivalence classes to search for the optimal pants decomposition.

2. Topology Operations for PD Computation

2.1. Pants Decomposition Graph

To analyze homotopy classes of pants decompositions, we define the following *Pants Decomposition Graph* (PD-graph). A PD-graph G is a dual graph of the pants decomposition. We create a node for each pants patch. Each pants patch has 3 boundaries; if two pants patches p and q share one or more boundary, we say p and q are *adjacent*. Then we create an edge in G for each pair of adjacent pants patches. Hence each edge corresponds to a cycle shared by two adjacent pants patches. On such a PD-graph, we can enumerate topologically inequivalent pants decompositions. Fig. 1(b,c) illustrate the PD and corresponding PD-graph on the genus-2 eight model. Note that a node can be adjacent to itself if two of its boundaries share a common cycle; gluing along this cycle we will get a $(1,1)$-typed patch.

2.2. Computing Initial Pants Decomposition

Before optimization, we shall compute an initial pants decomposition on a given surface M. Existing pants decomposition method such as [LGQ09] can be used. Here we present a general algorithm which is more efficient and reliable.

The homology basis of a genus-g surface M consists of $2g$ cycles. We can use the homology basis formed by the *handle* and *tunnel* cycles [DLSCS08]. From this homology basis, we can pick a subset B composed of g simple and pairwise disjoint cycles $\{b_1, b_2, \ldots, b_g\}$.

For a type-(g,b) surface, slicing all cycles in their B will lead to a type-$(0,2g+b)$ surface $\hat{M}$. We denote these $2g+b$ boundaries as $W=\{w_1, w_2, \ldots, w_{2g+b}\}$, and then iteratively, pick two boundaries w_i and w_j from W and compute a new simple cycle w' to bound them, i.e., w' homotopic to $w_i \circ w_j$. Then, w_i, w_j and w' form a pants patch M_k. We remove this M_k from $\hat{M}$; the left patch $\hat{M}' \leftarrow \hat{M} \backslash P_k$, $\hat{M}'$ is still genus-0 but its boundary number reduces by 1 (with two cycles w_i and w_j removed, one new cycle w' inserted). This is iteratively performed until $|W|=3$. This idea is formulated in Algorithm 1 and illustrated in Fig 2(a).

Algorithm 1. Initial PD computation.
In: A type-(g,b) surface M, $B(M)=\{b_1,\ldots,b_g\}$;
Out: Its PD $\{M_1, M_2, \ldots, M_{2g+b-2}\}$, where $M=\bigcup M_i$.
1). Cut M open all cycles in B and get a type-$(0,2g+b)$ surface $\hat{M}$ with boundaries $W=\{w_1, w_2 \ldots, w_{2g+b}\}$.
2). Pick two cycles in W, compute a shortest cycle w' homotopic to $w_i \circ w_j$.
3). $\{w_i, w_j, w'\}$ bound a pant patch M_k; remove M_k from $\hat{M}$, remove w_i, w_j from W and add w' into W.
4). If the remaining surface $\hat{M} \leftarrow \hat{M} \backslash M_k$ is not a pants patch, go to STEP 2, otherwise STOP.

2.3. Homotopic Cycle Computation

In Algorithm 1 and in the following sections, we need a basic operation that traces a cycle w' homotopic to cycle $w_i \circ w_j$. Given a surface M with boundaries $w_i, (i=1,2,\ldots,n)$, the following algorithm elaborates the computation of a cycle w' homotopic to the cycle $w_i \circ w_j$:

1). Trace a shortest path on M to connect w_i and w_j; then connect all other boundaries together using shortest paths (red curves in Fig. 2(b)).
2). Slice computed shortest paths apart; M becomes a topological cylinder.
3). Connect the two boundaries of the cylinder using a shortest path γ (the green curve in Fig. 2(b)).
4). Slice γ apart: each point p_k on γ splits into a pair $(p_k, \widetilde{p}_k)$, trace a shortest path to connect them. Among all shortest paths connecting $(p_k, \widetilde{p}_k)$, pick the one that has the minimal length, and this is cycle w' (blue cycle in Fig 2(b)).

2.4. Enumerating Pants Decompositions

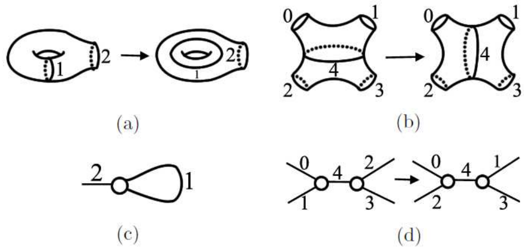

Figure 3: *A-Move and S-Move. (a) shows an S-move conducted on a handle patch, whose corresponding PD-graph is a self-connected node (c); (b) shows an A-move, whose PD graph and its local modification are shown in (d).*

We need to search a desirable pants decomposition from different homotopic classes. We perform this enumeration using two topological operations, denoted as *Associativity Move* (A-Move) and *Simple Move* (S-Move) [HLS00]. As illustrated in Fig. 3, take a cycle c and its corresponding edge e on the PD-graph: If e links two different nodes $p \neq q$, then gluing pants patches p and q along c will lead to a patch whose type is $(0,4)$ where we can do an *A-move*. If e links a node to itself $p = q$, then gluing p and q along c will lead to a patch of type $(1,1)$, where we can do an *S-move*. By sequentially applying one of these two operations, any two topologically different PDs can transform to each other and the entire topological space of PDs can be enumerated.

2.4.1. Computing the A-Move

Consider two adjacent pants patches p,q; let $\Gamma_1,\Gamma_2,\Gamma_3$ be 3 boundaries of p and $\gamma_1,\gamma_2,\gamma_3$ be 3 boundaries of q, and Γ_3 and γ_3 be the same boundary $\Gamma_3 = \gamma_3 \in C$. An A-Move will glue $\Gamma_3 = \gamma_3$ and generate a new cycle $\Gamma_4 = \gamma_4$ with a different homotopy type; and C will be updated: $C' \leftarrow C \backslash \{\Gamma_3\} \bigcup \{\Gamma_4\}$. Fig 4 illustrates this process, which is also elaborated in Algorithm 2.

Algorithm 2. Computing an A-Move.
In: Adjacent patches p,q, and their ordered boundaries $(\Gamma_1,\Gamma_2,\Gamma_3) \subset p$, $(\gamma_1,\gamma_2,\gamma_3) \subset q$, $\Gamma_3 = \gamma_3$.
Out: $\Gamma_4 = \gamma_4$ and updated maximal cuts.

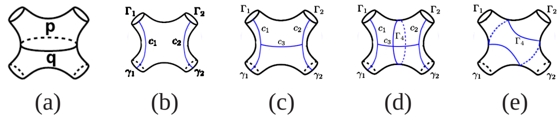

Figure 4: *Computing the A-Move. (a) The adjacent pants patches p (bounded by Γ_1,Γ_2, Γ_3) and q (bounded by γ_1,γ_2, $\gamma_3 = \Gamma_3$). (b) Glue p, q and get a type-(0,4) patch. The shortest paths c_1 and c_2 connect corresponding pairs of boundaries. (c) The shortest path c_3 connects c_1 and c_2. (d) Find the point pair with minimal length of shortest path Γ_4. (e) If q's boundary order is given as γ_2,γ_1, $\gamma_3 = \Gamma_3$, the A-Move result is different.*

1). Glue p and q along $\Gamma_3(\gamma_3)$.
2). Compute the shortest path c_1 connecting Γ_1 and γ_1 and the shortest path c_2 connecting Γ_2 and γ_2.
3). Compute the shortest path c_3 connecting c_1 and c_2.
4). Slice c_3 apart: each point $t_k \in c_3$ splits to a pair of points $(T_k, \widetilde{T}_k)$. Compute the shortest paths s_k connecting each pair of T_k and $\widetilde{T}_k$), pick the shortest path that has the minimal length. This path is the corresponding cycle Γ_4.
5). Update the maximal cut system $C' \leftarrow C \backslash \{\Gamma_3\} \bigcup \{\Gamma_4\}$.

All the traced shortest pathes are enforced to circumvent c_1, c_2 (in step 3), boundaries, and all the cycles in C (similar to the homotopic cycle computation discussed in the previous section). This guarantees the resulting Γ_4 is simple, non-trivial, and not homotopic to any other boundaries. The computed c_1, c_2, c_3 and the result Γ_4 are shown in blue in Fig. 4 (b,c,d). Note that the indices of boundary cycles matter: if the input boundaries of q becomes $(\gamma_2, \gamma_1, \gamma_3)$, then the result is shown in Fig. 4 (e).

2.4.2. Computing the S-Move

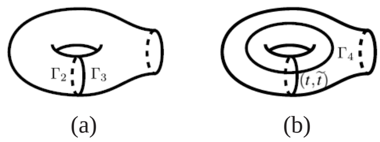

Figure 5: *Computing the S-Move. (a) The initial pants patch p with boundaries $\Gamma_2 = \Gamma_3$; (b) After slicing $\Gamma_2(\Gamma_3)$ open, each point splits into two corresponding points, the pair that has the shortest path connecting each other is selected and the path is denoted as the new cutting boundary Γ_4.*

Consider a pants patch p with 3 boundaries $\Gamma_1,\Gamma_2,\Gamma_3$ (suppose $\Gamma_2 = \Gamma_3$ are the same boundaries). To perform an S-move, we first slice Γ_2 open. For each point $t \in \Gamma_2$, after the split it has a corresponding point $\widetilde{t} \in \Gamma_3$. We trace the shortest path $Path(t,\widetilde{t})$ between t and $\widetilde{t}$. Since Γ_2 is cut open, the path will go around the handle (or tunnel). We trace shortest paths between all the point pair $(t,\widetilde{t})$ from (Γ_2,Γ_3), and pick the pair $t',\widetilde{t}'$, whose $Path(t',\widetilde{t}')$ has the shortest length. This shortest path is a new cycle Γ_4 with a different homotopy type. C should be updated accordingly: $C' \leftarrow C \backslash \{\Gamma_3\} \bigcup \{\Gamma_4\}$. Fig 5 illustrates the process of computing S-move.

3. Geometrically Optimizing Pants Decomposition

The previous section introduces topological operations to enumerate pants decompositions of different homotopy types. We can integrate different geometric criteria to guide the PD optimization.

All the tracing of cutting cycles are from Dijkstra's algorithm conducted on the weighted triangle meshes. These criteria can therefore be integrated into the weight of each triangle edge. The favored edges will have a smaller weights so that traced cycles will more likely go through them. We perform a breadth-first search on the PD-Graph and pick the decomposition whose cost is minimal.

Shortest Length. Shortest cutting cycles discretely approximate the geodesics and are simply desirable in most scenarios. One can minimize the total length of all the cycles in the maximal cut system: on edge $e = (v_i, v_j)$, simply take the Euclidean distance, $\sigma_l(e) = |v_i - v_j|^2$, between two vertices as its weight for the Dijkstra tracing. Then the optimized PD has the minimal total cutting cycle length.

Minima Rule. Human perception often cuts the surfaces along concave regions, which is known as the *minima rule* [LLS*05]. One may want to trace the cutting boundaries along the concave regions, i.e., the regions with salient negative minimal curvature. We can integrate the minima rules into our PD computation: (1) compute the minimal curvature $\kappa(v)$ for each vertex on the surface; (2) normalize the minimal curvature by $r(v) = (\kappa(v) - \mu)/\alpha$, where μ is the mean and α is the standard deviation of $\kappa(v)$ over all vertices of the surface; then (3) further normalize $r(v)$ into range $(0,1)$. The minima-rule weight of an edge $e = (v_i, v_j)$ is then defined as: $\sigma_m(e) = \frac{r(v_i)+r(v_j)}{2}$. Here $l(e)$ denotes the length of the edge and $c(e)$ is the average of the normalized minimal curvature of its two endpoints.

Symmetry. Many models have intrinsic symmetric patterns. One may prefer to cut the surface along its symmetry plane. We can define a scalar value on each vertex $d(v)$ using its distance to the symmetry plane; we can set the edge weight: $\sigma_s(e) = \frac{d(v_i)+d(v_j)}{2}$. Fig. 7 shows an example of symmetry-guided decomposition.

Edge weights can be combined to guide different PD optimization:

$$\sigma(e) = \sigma_l(e) \cdot (\sigma_m(e))^{\alpha_m} \cdot (\sigma_s(e))^{\alpha_s}, \qquad (1)$$

where weights α_m, α_s indicate importance of different terms.

4. Consistent Decomposition of Multiple Models

We generalize the above PD optimization framework from one surface to multiple surfaces. We elaborate the idea on two models M_1 and M_2, whose generalization to more surfaces is straightforward.

4.1. Initial Consistent PD Computation

We can compute the consistent decomposition of M_1 and M_2 using the algorithm in Section 3.

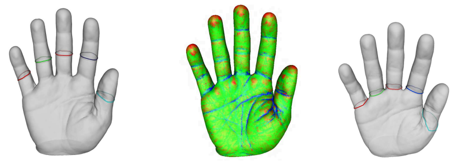

Figure 6: *PDs of the Hand model using different criteria. (a) PD cycles traced via shortest lengths; (b) the minimal curvature field and (c) PD cycles traced via minima rule.*

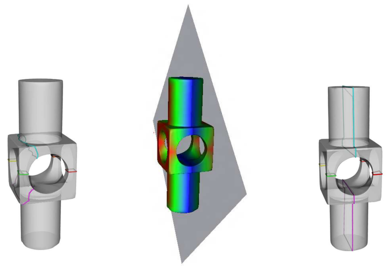

Figure 7: *PD of the Mechanical part. (a) PD with shortest path, (b) the symmetry plane, (c) PD guided by symmetry.*

The correspondence between pants decompositions of surfaces is uniquely determined by the indexing of our starting $2g+b$ boundaries W of each surface. An unique indexing given to these $2g+b$ boundaries determines a valid topological type of the initial PD.

The correspondence between $2g+b$ boundaries can be specified by user, or computed using heuristic methods such as [LGQ09] (by spatially matching the mass centers of handle loops). Now, given an arbitrary indexing of $W(M_1)$ and $W(M_2)$, say, $T_1, \ldots, T_{2g+b}$ and $T'_1, \ldots, T'_{2g+b}$, respectively. We follow the same consistent order to compute the partitioning cycles, i.e., trace $w' \sim w_{T'_i} \circ w_{T'_j}$ in M_2 if and only if we trace cycle $w \sim w_{T_i} \circ w_{T_j}$ in M_1. The correspondence between these two maximal cut system will be preserved and we can obtain a consistent PD on both M_1 and M_2.

4.2. Consistently Optimizing Multiple Decompositions

Using a similar aforementioned greedy searching scheme, we simultaneously optimize both decompositions on PD-Graphs $G_0(M_1)$ and $G_0(M_2)$. The same S-Move or A-Move should be applied on both the cycle $c_i \in C(M_1)$ and its corresponding cycle $c'_i \in C(M_2)$. Then we will obtain the new PD-Graphs $G_i(M_1)$ and $G_i(M_2)$, isomorphic to each other. During this searching, we consider weights on each edge to be the sum of the weights defined on $G_i(M_1)$ and $G_i(M_2)$. Note that each initial topological correspondence indicates a homotopy type of consistent PD of M_1 and M_2. If geometrically or semantically, some specific homotopy type is preferred, we can also simply adopt a preprocessing (using e.g.

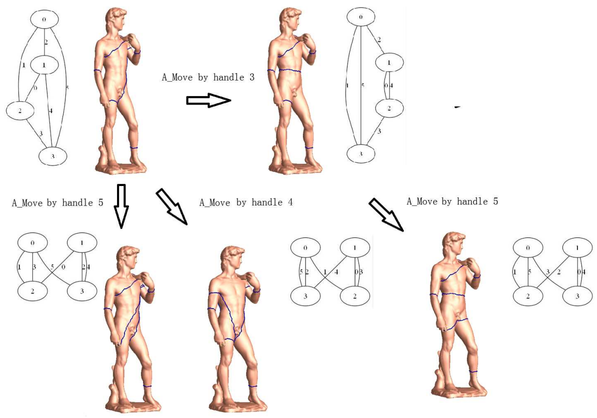

Figure 8: *Enumerating PD of the David model, with both the decomposition and corresponding PD-graph illustrated. This figure shows part of the enumeration space. The first row indicates a path to a PD with the shortest total length.*

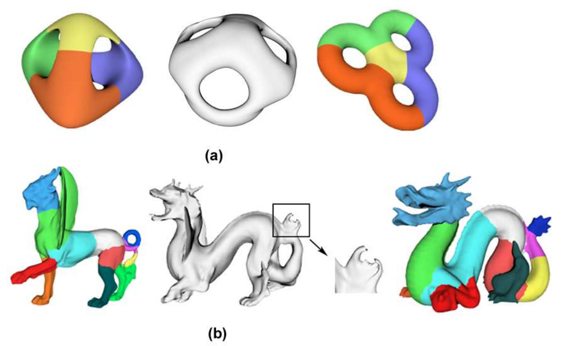

Figure 9: *Consistent PD, cross-surface map, and morphing. The source and target models are decomposed consistently; the cross-surface map is computed and used to generate the linear interpolation. The original,* 50% *morphed, and the target models are shown.*

RANSAC or spectral matching) to modify the correspondence before applying this above simultaneous optimization.

5. Experimental Results

The computational complexity to compute the initial pants decomposition is $O(gn^{3/2}\log n)$ where g is the genus and n is the number of vertices, while it takes $O(gn^{3/2}\log n)$ to finish one ring search.

Fig. 8 shows our enumeration of different PDs (and corresponding PD-graphs) of the genus-3 Michelangelo-David model. On every state, both the maximal cut system and corresponding PD-graph are illustrated. Fig. 9 shows some examples of consistent PD on multiple objects (a) genus-3 part and 3-torus; (b) genus-2 Feline and genus-1 dragon. The cross-surface mappings are then computed, and the results are visualized using the inter-shape morphing computed by the simple linear interpolation (see the accompanied video). In (a), consistent PDs are computed using $\alpha_m = 0$ and $\alpha_s = 0$; surface mapping are computed automatically with no user involvement. In (b), an additional pair of surgery points are placed on the tail of the dragon corresponding to the blue handle of Feline. We set $\alpha_m = 1$ and $\alpha_s = 0$. The handle opens up during its morphing to the dragon's tail.

6. Conclusion

We propose a computational framework to search for the pants decomposition of different topological classes. Compared with other consistent segmentation methods, PD decomposition has a key advantage that its topological structure (the dual graph of the decomposition) is simple and canonical: each sub-patch is topologically uniform (type-(3,0)) and shares 3 simple boundary cycles with adjacent patches. Sub-patches can therefore be modeled using a unified algorithm.

A limitation of this work is that two PDs that are not homotopic to each may have the same corresponding PD-graph. This means, to enumerate all the PDs, we need to (re)visit all the generated PD-graph. To track the enumeration without ambiguity, we will explore a new descriptor to characterize PD's topology (e.g. by further encoding the precise homotopy information of each cycle on the corresponding edge of PD graph).

Acknowledgements

This work is partially supported by Louisiana Board of Regents Research Competitiveness Subprogram (RCS) LEQSF(2009-12)-RD-A-06, LA-BOR PFund:NSF(2011)-PFund-236, National Natural Science Foundation of China No. 61170323, and the open program of the CAD&CG state lab, Zhejiang University, China.

References

[DLSCS08] Dey T. K., Li K., Sun J., Cohen-Steiner D.: Computing geometry-aware handle and tunnel loops in 3d models. *ACM Trans. Graph. 27* (August 2008), 45:1–45:9. 2

[GF09] Golovinskiy A., Funkhouser T.: Consistent segmentation of 3D models. *Computers and Graphics (Shape Modeling International 09) 33*, 3 (June 2009), 262–269. 1

[HLS00] Hatcher A., Lochak P., Schneps L.: On the Teichmüller tower of mapping class groups. *J. Reine Angew. Math. 521* (2000), 1–24. 3

[HS94] Hershberger J., Snoeyink J.: Computing minimum length paths of a given homotopy class. *Comput. Geom. Theory Appl. 4*, 2 (1994), 63–97. 1

[KS04] Kraevoy V., Sheffer A.: Cross-parameterization and compatible remeshing of 3D models. *ACM Trans. Graph. 23*, 3 (2004), 861–869. 1

[KZW11] Kwok T.-H., Zhang Y., Wang C. C.: Efficient optimization of common base domains for cross-parameterization. *IEEE Trans. on Visualization and Computer Graphics 99*, PrePrints (2011). 1

[LBG*08] Li X., Bao Y., Guo X., Jin M., Gu X., Qin H.: Globally optimal surface mapping for surfaces with arbitrary topology. *IEEE Trans. on Visualization and Computer Graphics 14*, 4 (2008), 805–819. 1

[LGQ09] Li X., Gu X., Qin H.: Surface mapping using consistent pants decomposition. *IEEE Trans. on Visualization and Computer Graphics 15*, 4 (2009), 558–571. 1, 2, 4

[LLS*05] Lee Y., Lee S., Shamir A., Cohen-Or D., Seidel H.-P.: Mesh scissoring with minima rule and part salience. *Comput. Aided Geom. Des. 22*, 5 (July 2005), 444–465. 4

[NGH04] Ni X., Garland M., Hart J. C.: Fair morse functions for extracting the topological structure of a surface mesh. In *ACM SIGGRAPH* (2004), pp. 613–622. 1

[Sha08] Shamir A.: A survey on mesh segmentation techniques. *Computer Graphics Forum 27*, 6 (2008), 1539–1556. 1

[SSSCo08] Shalom S., Shapira L., Shamir A., Cohen-or D.: Part analogies in sets of objects, 2008. 1

[VDA07] V. K., D. J., A S.: Shuffler: Modeling with interchangeable parts. *The Visual Computer* (2007). 1

[VL07] Verdière E. C. D., Lazarus F.: Optimal pants decompositions and shortest homotopic cycles on an orientable surface. *J. ACM 54*, 4 (2007). 1