2D Rationalization

Jonathan Bachrach

EECS UC Berkeley

September 15, 2015

2D Rationalization

- simplification of 3d shape into few simple + cheap 2d parts

Hildebrand + Bickel + Alexa

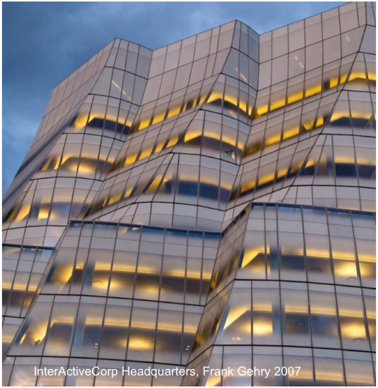
Taxonomy

- pre and post rationalized
  - when design when rationalization
- 1d, 2d, or 3d parts
  - geometric outcome
- reduced form parts
  - classes: planes, cones, ...
  - manufactured parts: plates, legos
- construction techniques
  - feasible
  - joinery
  - seams
  - fidelity

Pre and Post Rationalization

London City Hall, Foster and Partners 2002

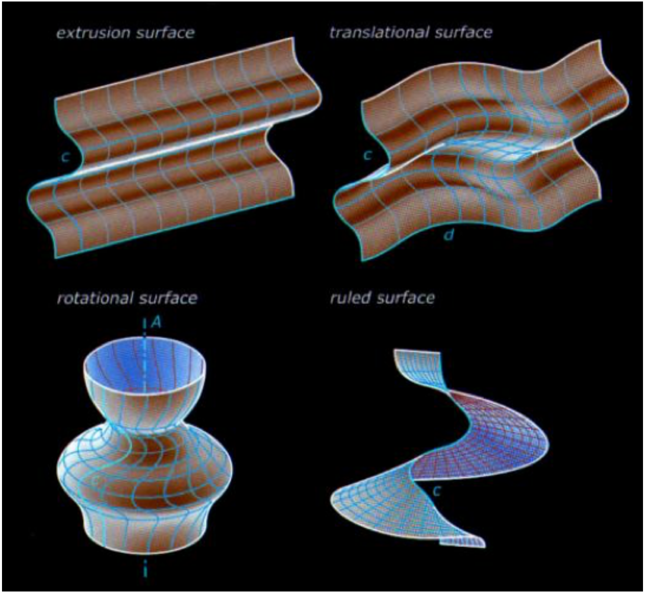
pre rationalization
computationalist first
foster

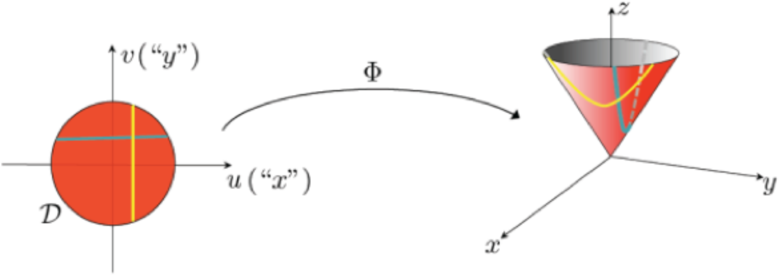


InterActiveCorp Headquarters, Frank Gehry 2007

post rationalization
designer first
gehry

Fabricatible Surfaces




v (“y”)
u (“x”)
$\mathcal{D}$
Φ
z
y
x

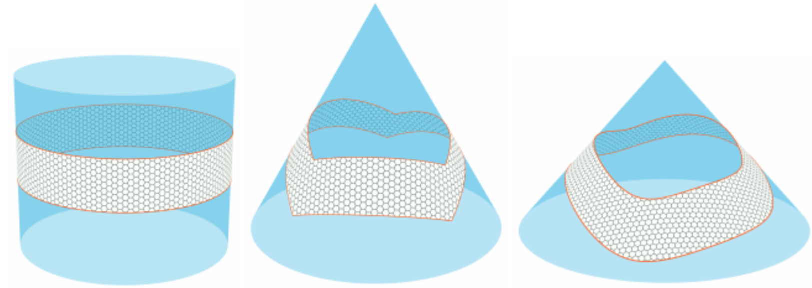
Surface panelization using periodic conformal maps

Thilo Rörig, Stefan Sechelmann

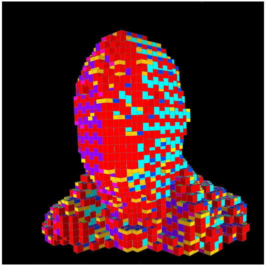
Institut für Mathematik, Technische Universität Berlin

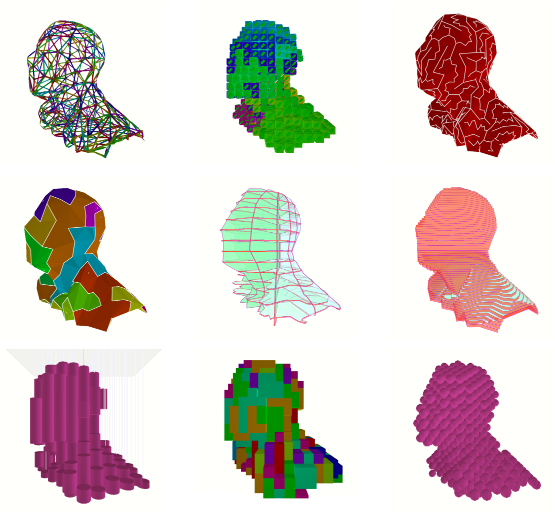
Agata Kycia, Moritz Fleischmann

HENN Research, HENN Architekten



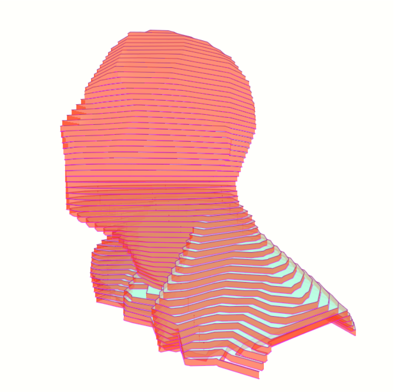
Post Rationalization

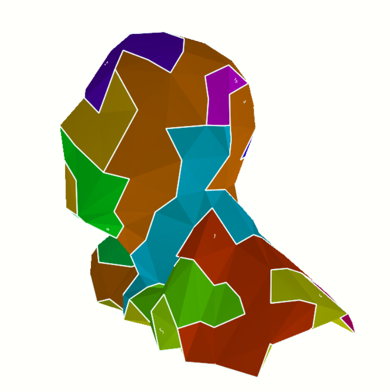




post rationalization

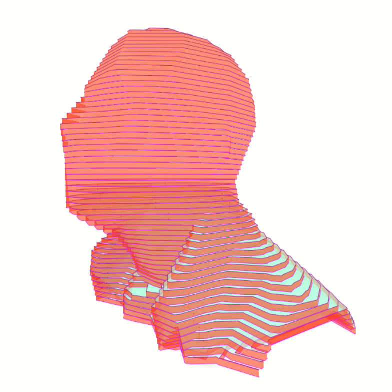
- 3d -> 2d
  - slicing
  - panelization

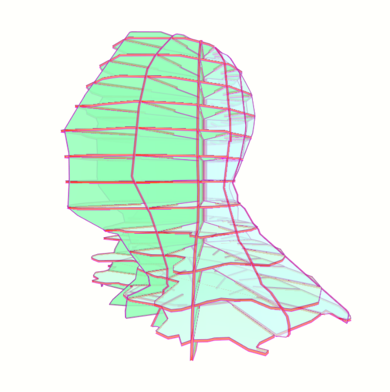




Simple Slicing

- regular slices
- either tab or slot joinery
- simple assembly ordering





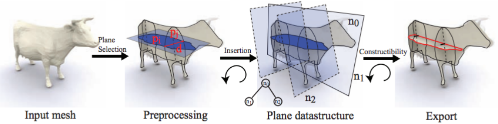
crdbrd: shape fab by sliding planar slices

- break model into interlocking slices
- intelligently find planes / polygons – greedily select them
- quickly check insertion feasibility

crdbrd: Shape Fabrication by Sliding Planar Slices

Kristian Hildebrand Bernd Bickel Marc Alexa

TU Berlin

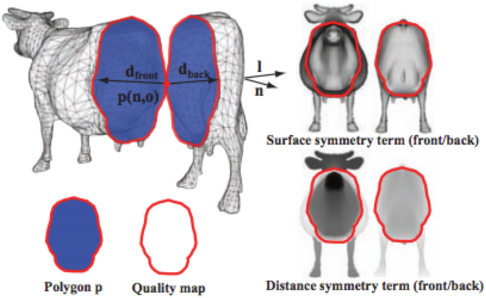


plane selection

how well a polygon covers geometric shape features

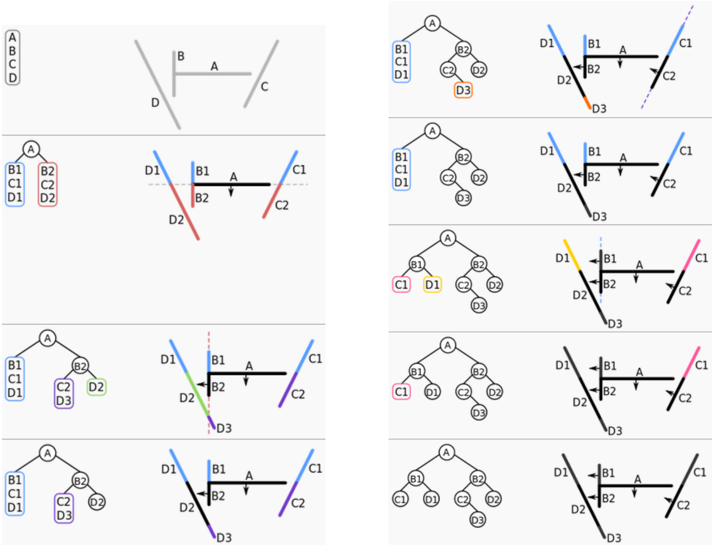
- *D* distance symmetry term
  - select planes that are centered
- *I* surface symmetry term symmetry term
  - favor planes with normals oriented in its average surface direction

$$u = \int_{\Omega} V \cdot I \cdot$$

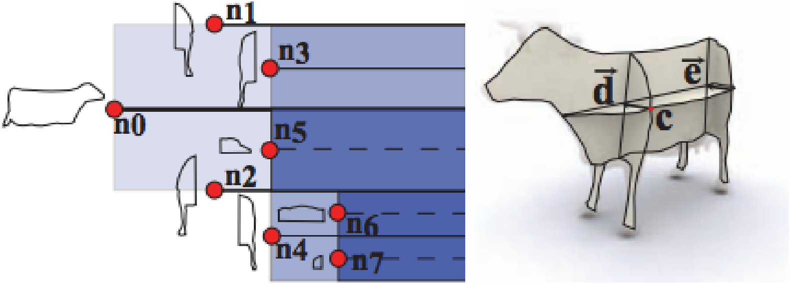


BSP

recursively divide shape into two



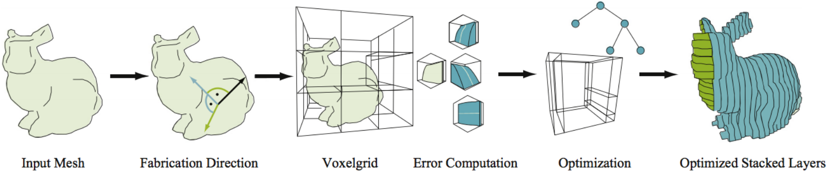
- fast BSP type data structure for checking



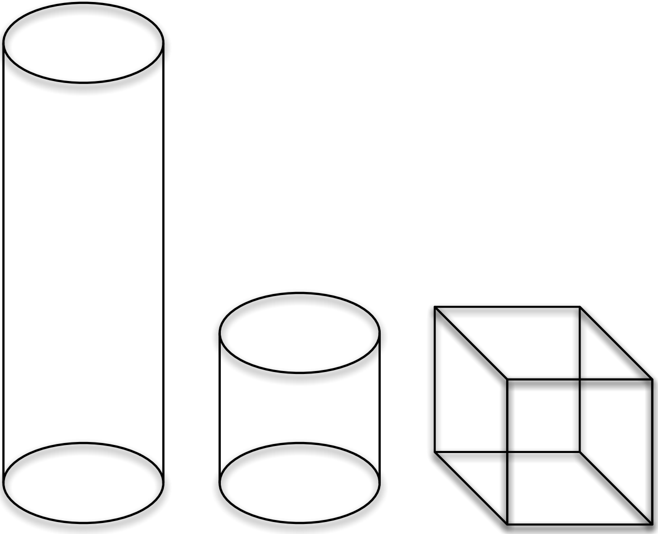
- decides on slice axes
- optimizes for best slice direction

Orthogonal slicing for additive manufacturing

Kristian Hildebrand *, Bernd Bickel, Marc Alexa
TU Berlin, Einsteinufer 17, 10587 Berlin, Germany

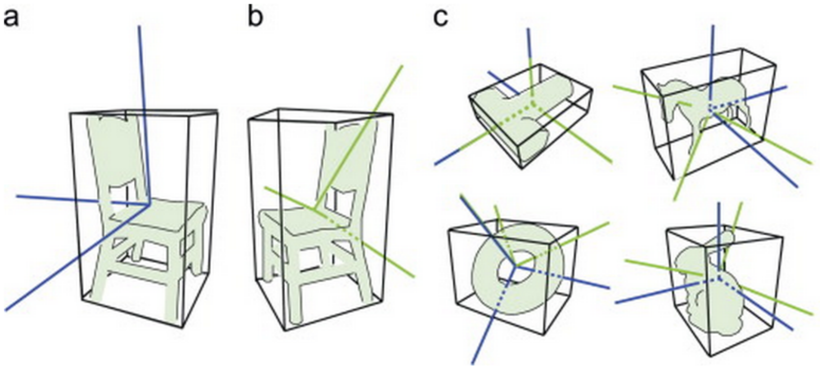


slice axis

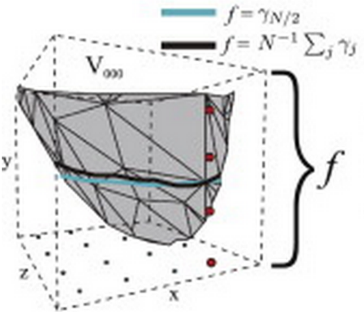


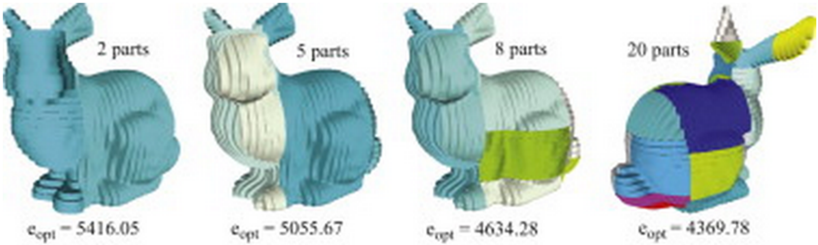
a planar surface with normal direction *n* should be sliced in a direction orthogonal to *n*, because the accuracy in the tangests of a slice is supposed to be significantly higher than normal to a slice.

- clustering on triangle normals
- turn into orthogonal vectors using SVD



- voxelize shape
- try all three directions for each voxel
- decomposition into half spaces
  - sum over all boundary voxel errors
  - choose direction with minimal error

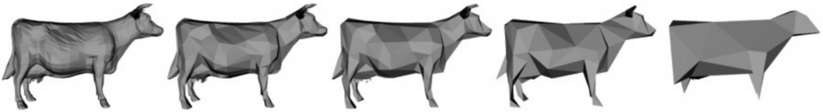




Simple Panelization

- simplify mesh to low triangle count
- greedily grow flattenable pieces

- fit quadric to each vertex
- measure edge collapse affect on quadric error
- collapse lowest cost edges first



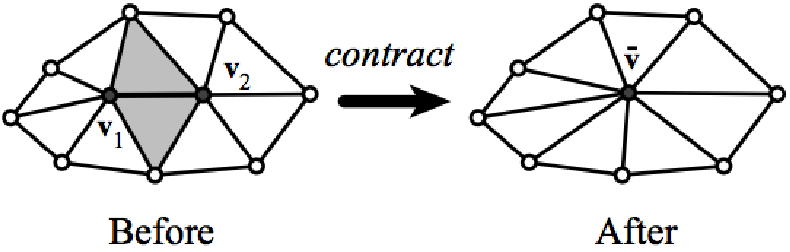
Surface Simplification Using Quadric Error Metrics

Michael Garland* Paul S. Heckbert†

Carnegie Mellon University

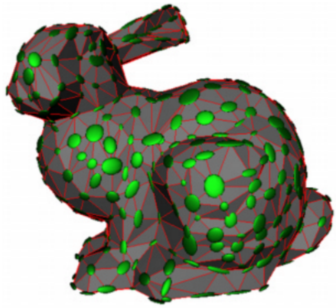
Edge Collapse

- fit quadric to each vertex
- measure edge collapse affect on quadric error
- collapse lowest cost edges first



Quadric Fitting

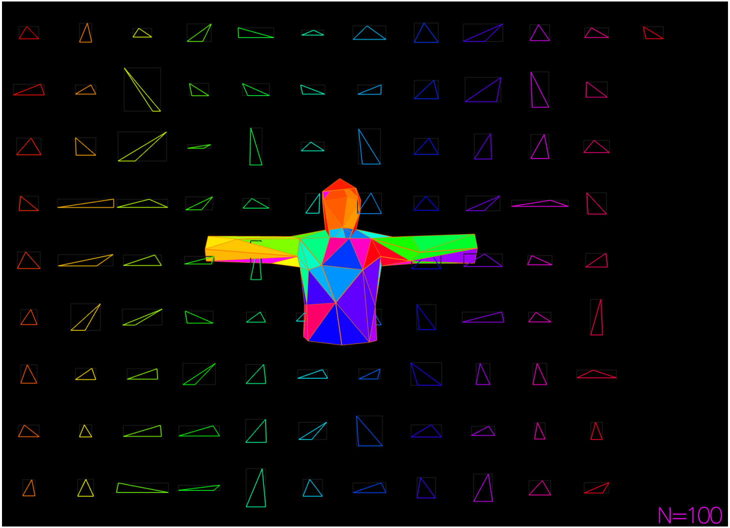
- fit quadric to each vertex based on neighboring vertices
- ellipsoid
- simple to calculate



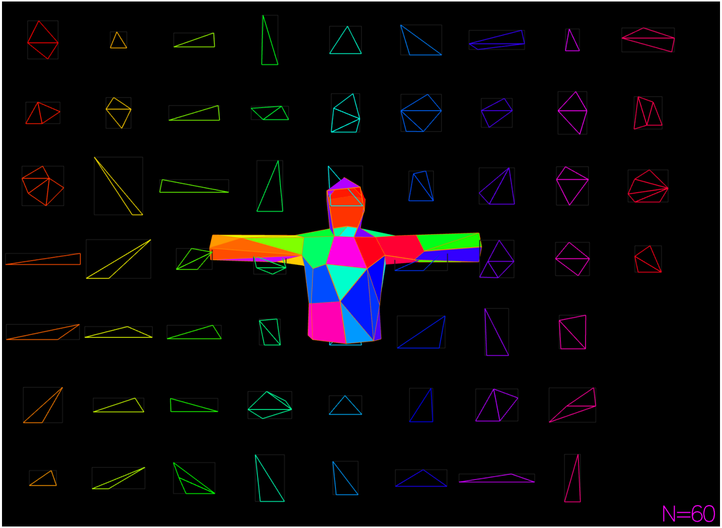
Greedy Panelization from Simple Meshes

- start with triangles being clusters
- repeatedly merge neighboring clusters if can flatten
  - project onto plane with normal down
  - no self-intersections
  - fits on stock
  - *could add darts*
- measure cost as complexity etc
- choose best cost first

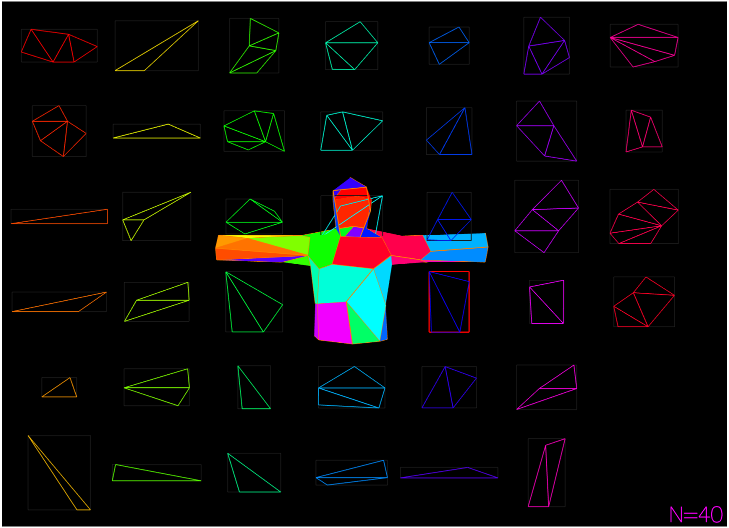
Shirt Example 100 Triangles



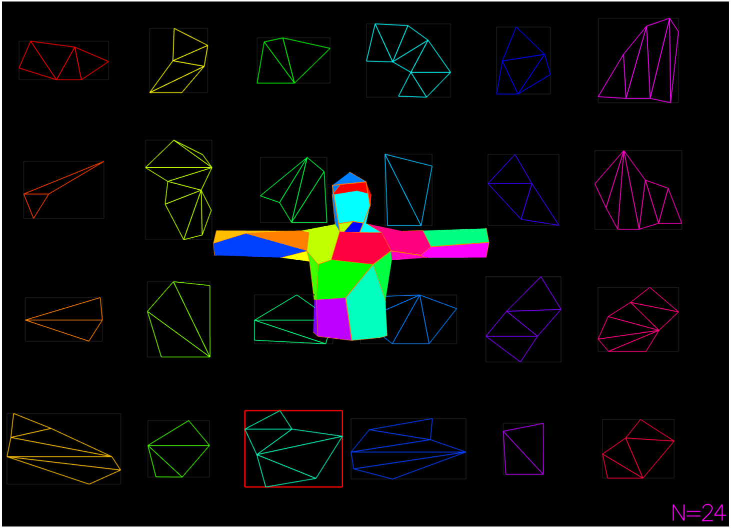
Shirt Example 60 Triangles

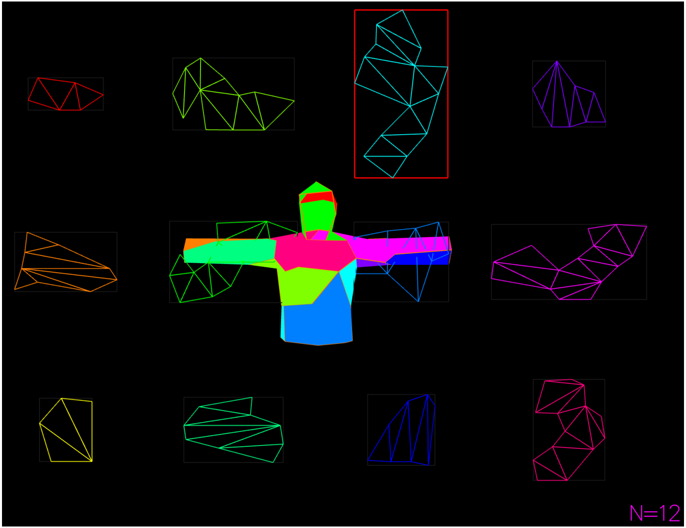


Shirt Example 40 Triangles

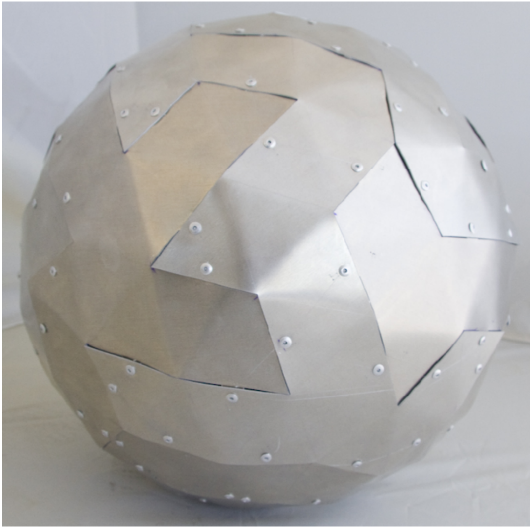


Shirt Example 24 Triangles



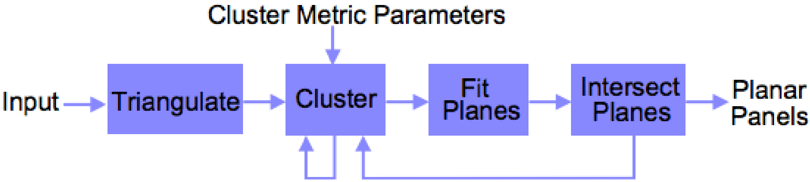

N=12

Darpa Hoodie



Constrained Planar Remeshing for Architecture

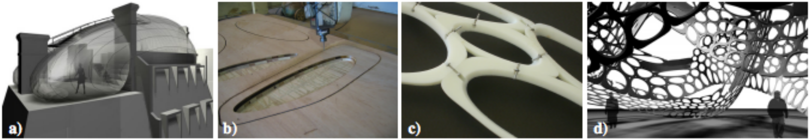
- clustering based on error
- fit plane through clusters
- start over if bad



Constrained Planar Remeshing for Architecture

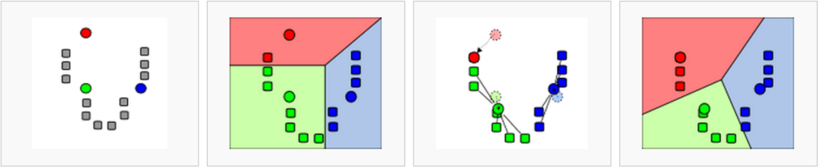
Barbara Cutler
Department of Computer Science
Rensselaer Polytechnic Institute

Emily Whiting
Departments of Architecture and Computer Science
Massachusetts Institute of Technology

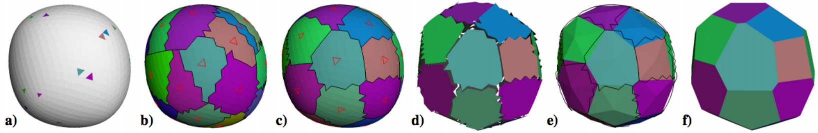


K-Means

- initialize cluster means
- assign points to clusters based on proximity
- move clusters means as centroids
- repeat until converged



- initialize triangle seeds
- clustering based on error to distance to plane
- fit plane through clusters
- find next seeds
- repeat until converges



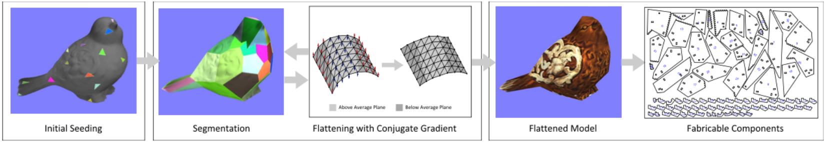
Computing and Fabricating Multiplanar Models

Desai Chen, Pitchaya Sitthi-amorn, Justin T. Lan and Wojciech Matusik

MIT CSAIL

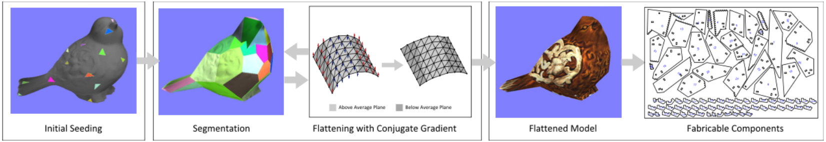
- clustering with adjustable number k-means++
- adjust vertices to maintain texture coordinates
- ensure admissibility

Input Mesh (10k faces) Multiplanar Model (85 polygons) Fabricated Model

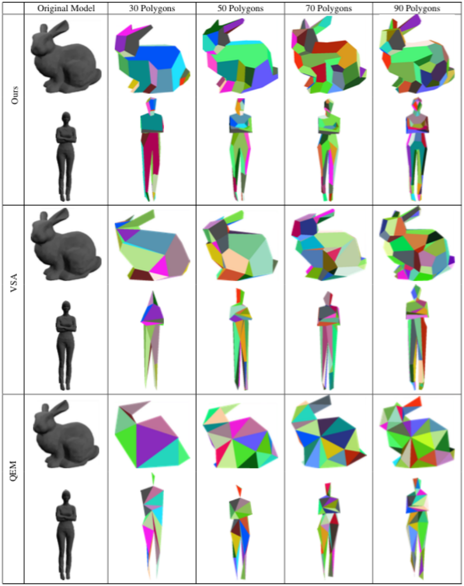


Core Rationalization Algorithm

- segmentation
- deformation



Multiplanar Results



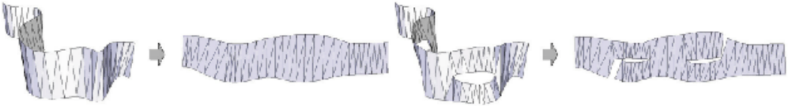
Multiplanar Constructions

Making Papercraft Toys from Meshes using Strip-based Approximate Unfolding

- find feature lines forming regions
- break parts into zonal regions using topological distance
- add internal cut-lines to maintain important features
- smooth cutting lines
- edge merge within regions to create triangle strips
- project to plane and split strips if they self-intersect

Triangle Strips Have Beneficial Features for Papercraft:

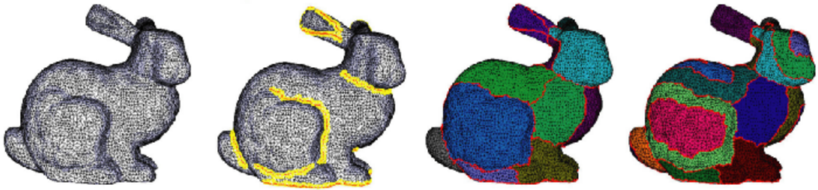
- to unfold is straightforward
- need no cut (or few cuts)



goal is to approximate mesh by set of smooth and wide strips

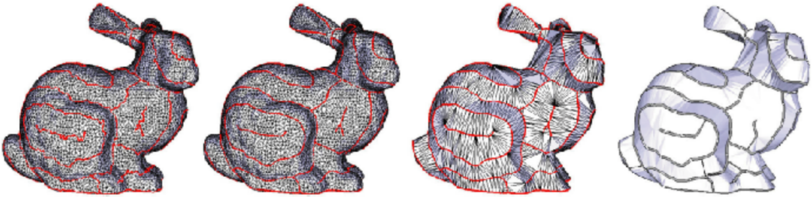
- smoothness: can avoid bending and make surface smooth
- wideness: can reduce construction time

- (a) initial mesh
- (b) extracted feature lines
- (c) partitions based on the feature lines
- (d) zonal regions segmented according to topological distances from part boundaries



(a) (b) (c) (d)

- (e) cut-lines: part boundaries, feature cut-lines, center cut-lines
- (f) smoothed cut-lines
- (g) strips generated by constrained mesh simplification
- (h) stips with cut-lines enhanced



(e) (f) (g) (h)

Strips Bunny

Original model: 19996 triangles

Height: 17cm

Assemble time: 2h45m

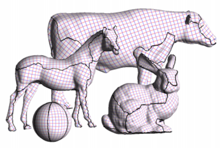
# of Parts: 35

Original model: 18496 triangles
Height: 14cm
Assemble time: 3h30m
# of Parts: 27

D-Charts

Quasi-Developable Mesh Segmentation

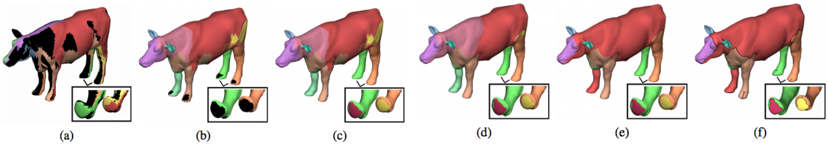
- clustering based on develope mapping error
- hole filling
- post processing



D-Charts: Quasi-Developable Mesh Segmentation

Dan Julius[†] Vladislav Kraevoy[‡] Alla Sheffer[§]

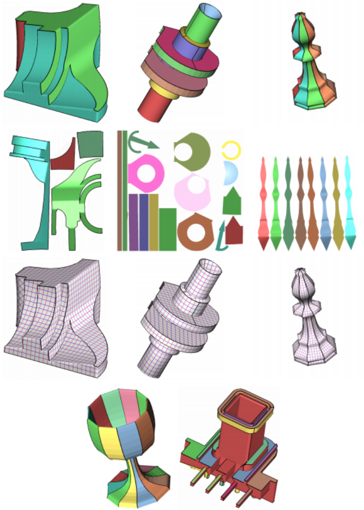
University of British Columbia, Vancouver, B.C., Canada



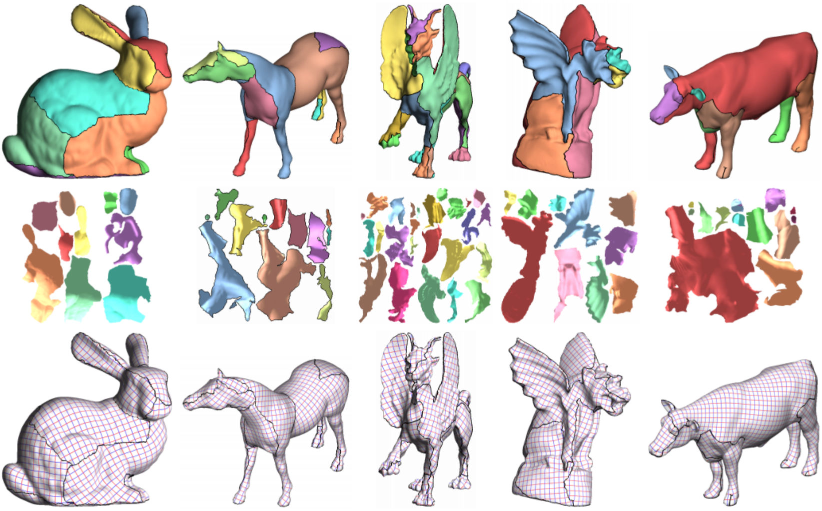
Mesh Developable Parts K-Means

- assign initial clusters triangle seeds
  - use farthest points algorithm for choosing
- grow out clusters from neighbors
  - assign cost to triangle of adding to cluster
  - use priority queue to choose lowest cost triangle
  - stop when hit error threshold
- reassign proxy and seed
  - solve for conic proxy
  - compute new seed as low error and near center

D-Charts Mechanical Results



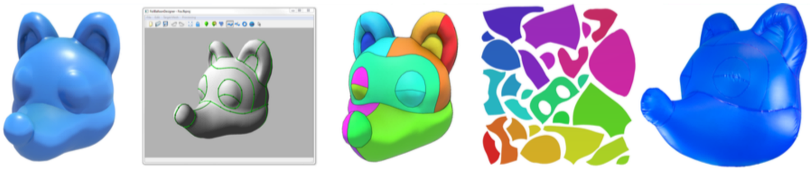
D-Charts Free Results



otherlab

Designing Inflatable Structures

- find panels that minimize inflation error
- user or automatic initial seam locations
- moves points to decrease inflation error
- minimize seam matching error

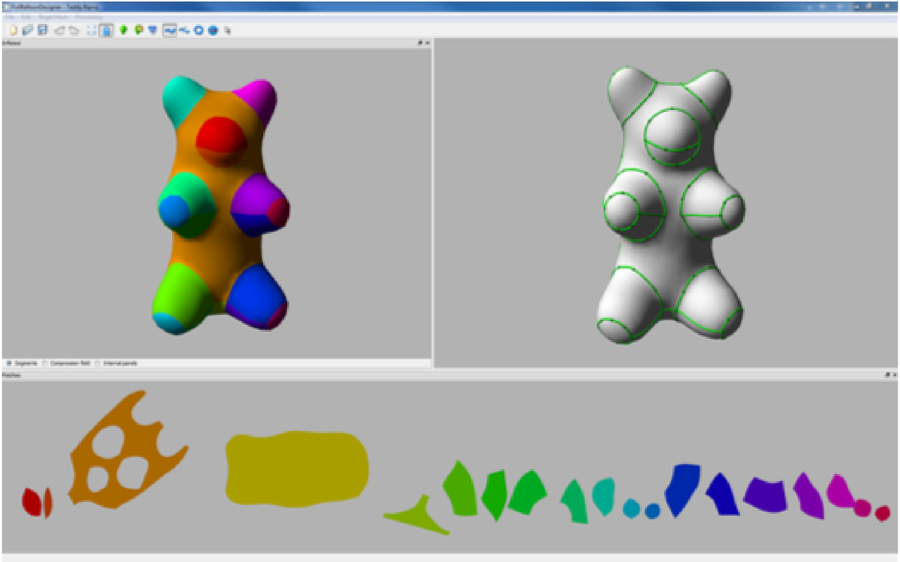


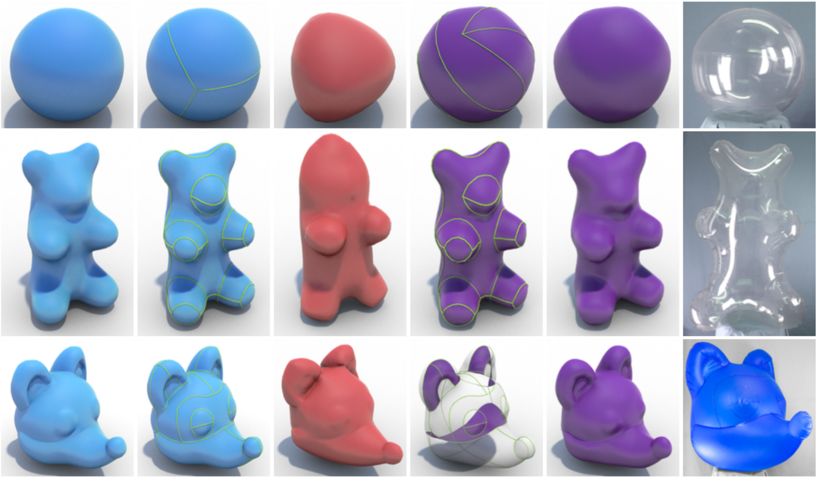
Designing Inflatable Structures

Mélina Skouras[1,2] Bernhard Thomaszewski[2] Peter Kaufmann[2] Akash Garg[3] Bernd Bickel[2]
Eitan Grinspun[3] Markus Gross[1,2]

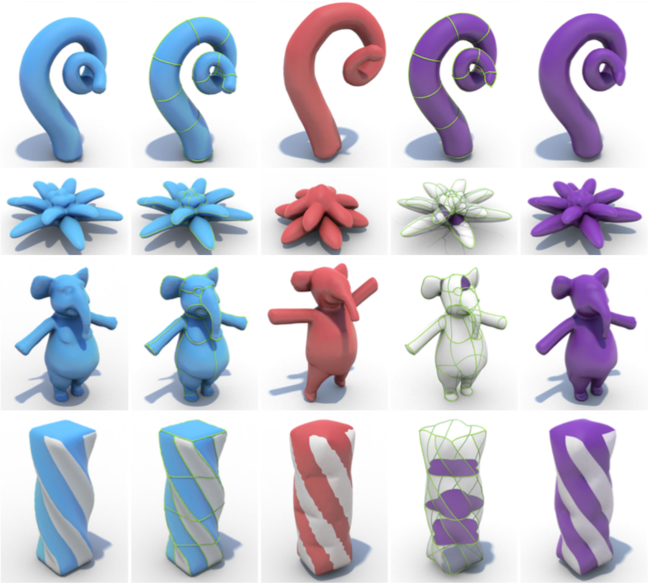
[1]ETH Zurich [2]Disney Research Zurich [3]Columbia University

Inflatables GUI





Inflatables Results



Future Panelization Directions

- fit multiple shapes
- use darts to build bigger panels
- more simulation in loop

- *Crdbrd: Shape Fabrication by Sliding Planar Slices* by Hildebrand + Bickel + Alexa
- *Orthogonal Orthogonal Slicing for Additive Manufacturing* by Hildebrand + Bickel + Alexa
- *Surface Simplification Using Quadric Error Metrics* by Garland + Heckbert
- *Constrained Planar Remeshing for Architecture* by Cutler + Whiting
- *Making Papercraft Toys from Meshes using Strip-based Approximate Unfolding* by Mitani + Suzuki
- *D-Charts: Quasi-Developable Mesh Segmentation* by Julius + Kraevoy + Sheffer
- *Designing Inflatable Structures* by Skouras + Thomaszewski + Kaufmann + Garg + Bickel + Grinspun + Gross