Directional Field

Xiao-Ming Fu

Outlines

- Introduction
- Discretization
- Representation
- Objectives and Constraints

Outlines

- Introduction
- Discretization
- Representation
- Objectives and Constraints

Definition

- Spatially-varying directional information, assigned to each point on a given domain.
  - A field on a domain is the assignment of a directional to each point in the domain.

- Magnitude + direction

1. Direction
2. Length

Multi-valued field

- Multiple directions per point with some notion of symmetry
- A set of directions or vectors at every point.
  - Rotationally-symmetric direction fields (*RoSy fields*)
  - $N = 1, 2, 4, 6$
  - Four directions with $\pi/2$ RoSy
  - Two independent pairs of directions with π RoSy within each pair.

- *Vector fields*
- *Direction fields*
- *Line fields*
- *Cross fields*
- *Frame fields*
- *RoSy fields*
- *N-symmetry fields*
- *PolyVector fields*
- *Tensor fields*

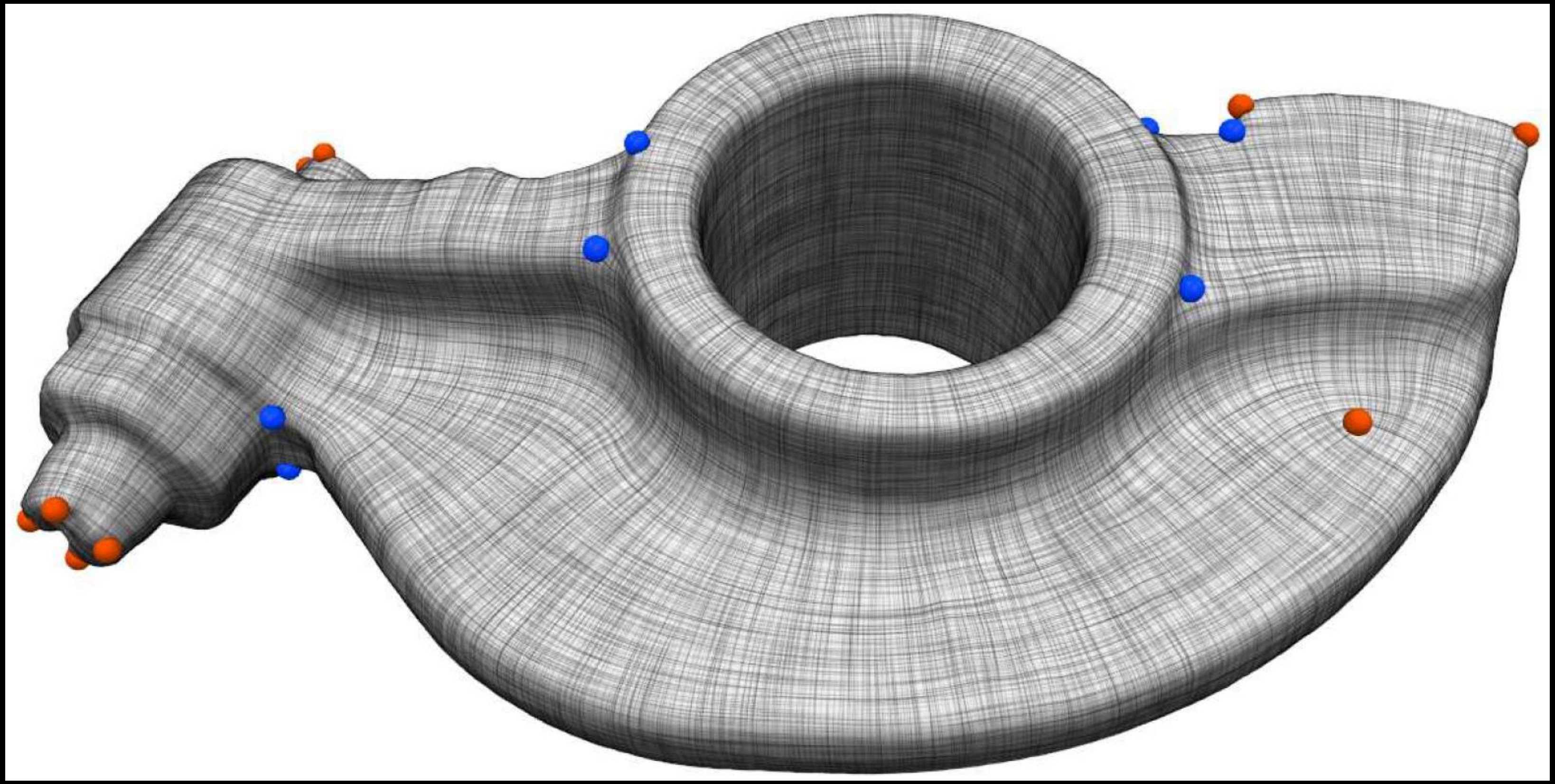
	1-vector field	One vector, classical “vector field”
	2-direction field	Two directions with π symmetry, “line field”, “2-RoSy field”
	1^3-vector field	Three independent vectors, “3-polyvector field”
	4-vector field	Four vectors with $\pi/2$ symmetry, “non-unit cross field”
	4-direction field	Four directions with $\pi/2$ symmetry, “unit cross field”, “4-RoSy field”
	2^2-vector field	Two pairs of vectors with π symmetry each, “frame field”
	2^2-direction field	Two pairs of directions with π symmetry each, “non-ortho. cross field”
	6-direction field	Six directions with $\pi/3$ symmetry, “6-RoSy”
	2^3-vector field	Three pairs of vectors with π symmetry each

Some concrete examples

- Principal directions of a shape
- Stress or strain tensors
- The gradient of a scalar field
- The advection field of a flow
- Diffusion data from MRI

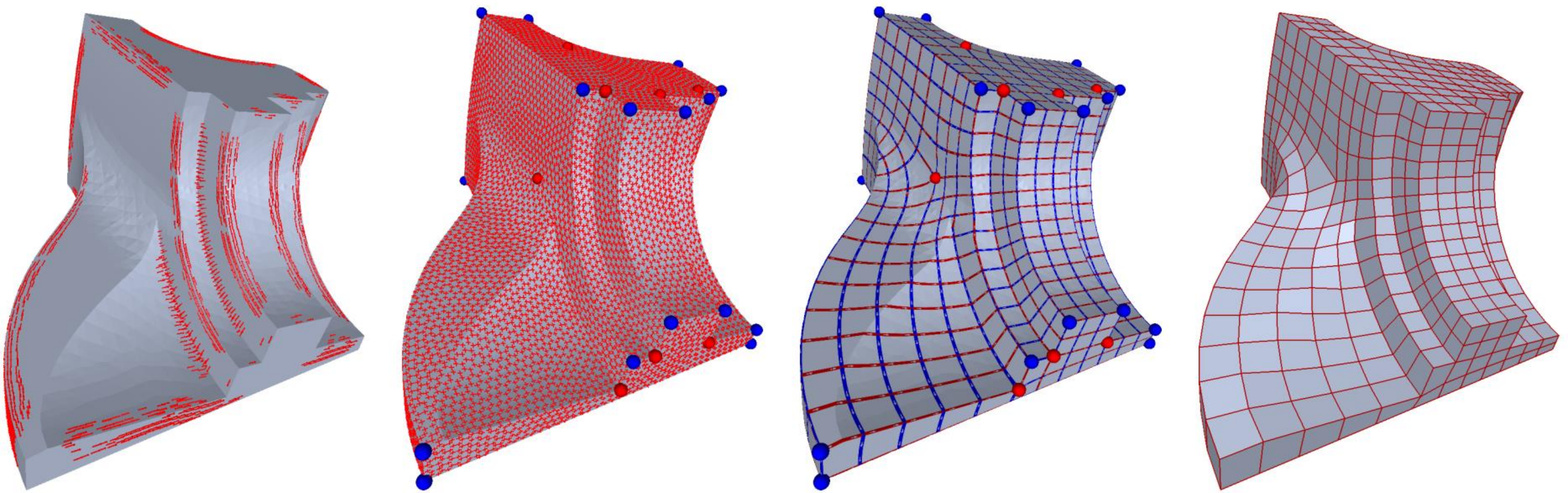
Synthesis or design

- User constraints
- Alignment conditions
- Fairness objectives
- Physical realizations



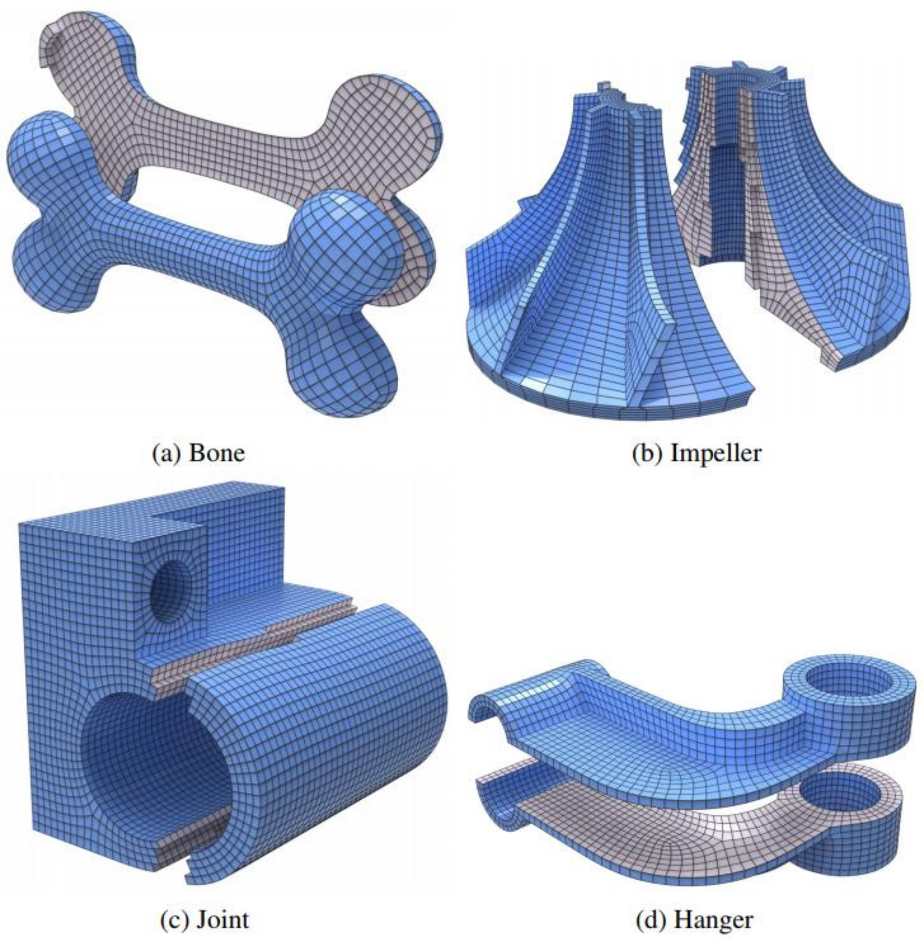
Applications

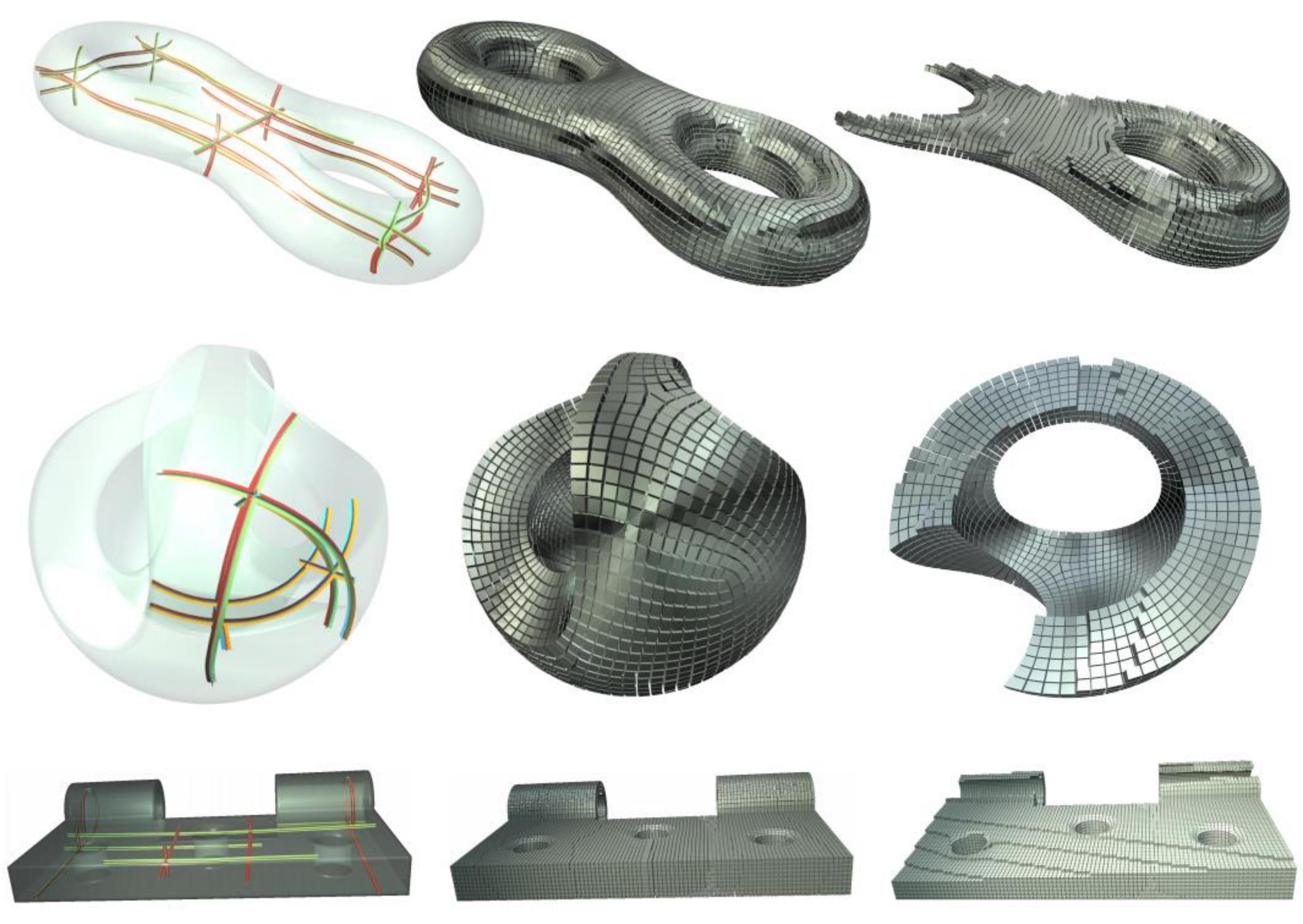
- Mesh Generation



Applications

- All-hex meshing





Deformation

- Deformations which are as isometric as possible can be generated using approximate Killing vector fields.

Texture Mapping and Synthesis

- Seamless texture

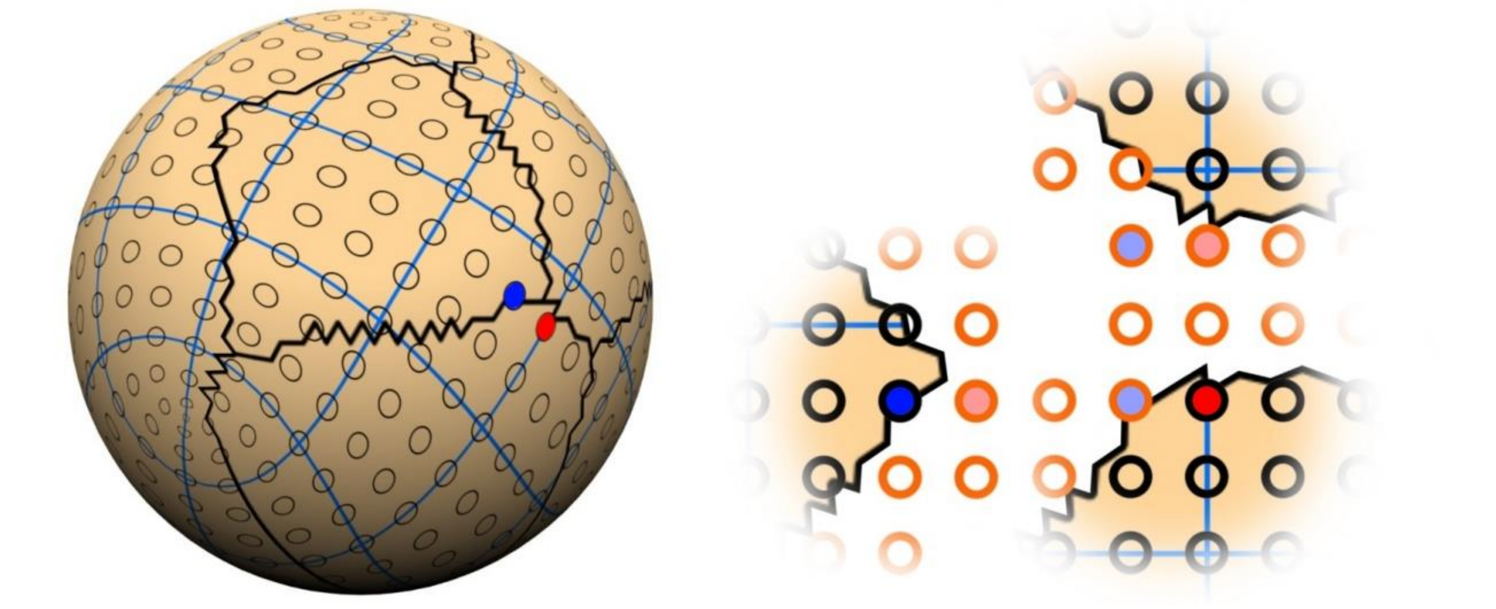
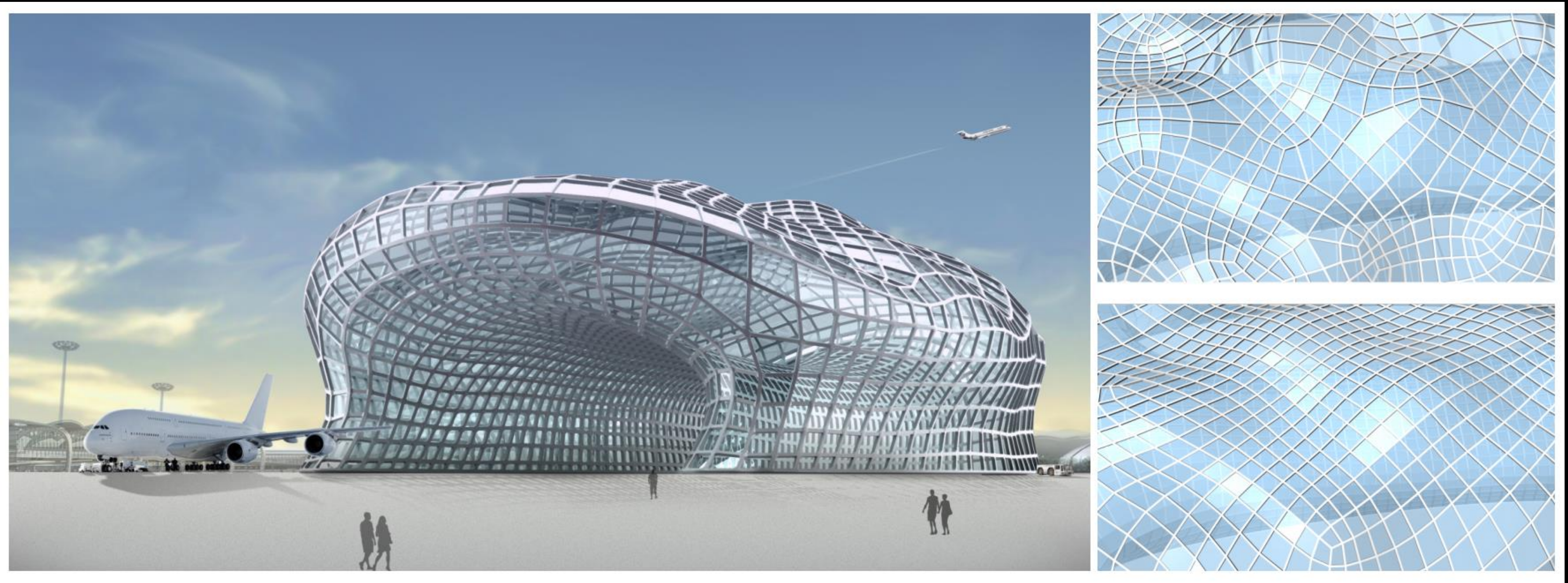


Figure 1: *Neighboring texels on the surface (left, red/blue dots) are not neighbors in the atlas (right). To make the seam invisible, texels have to align across chart boundaries and their colors have to be duplicated (light colored texels).*

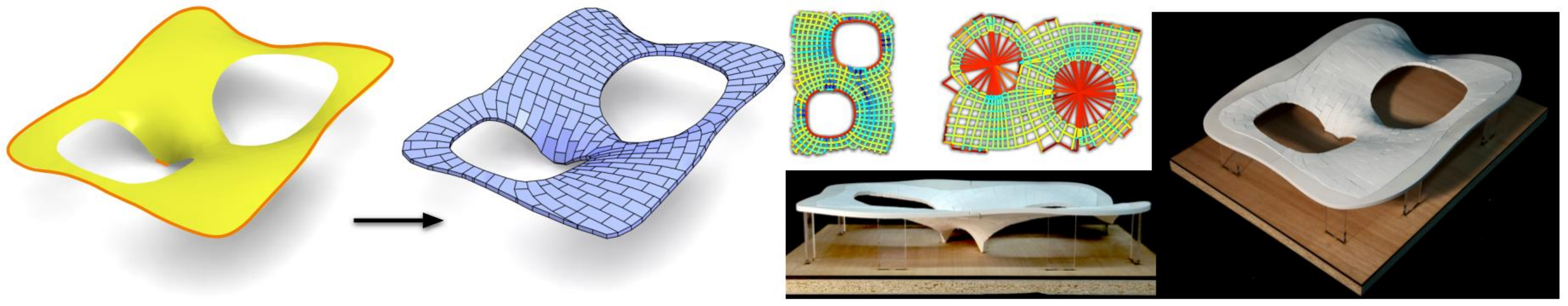
Architectural Geometry

- Conjugate directions



Architectural Geometry

- Self-Supporting Structures



Outlines

- Introduction
- Discretization
- Representation
- Objectives and Constraints

Tangent Spaces

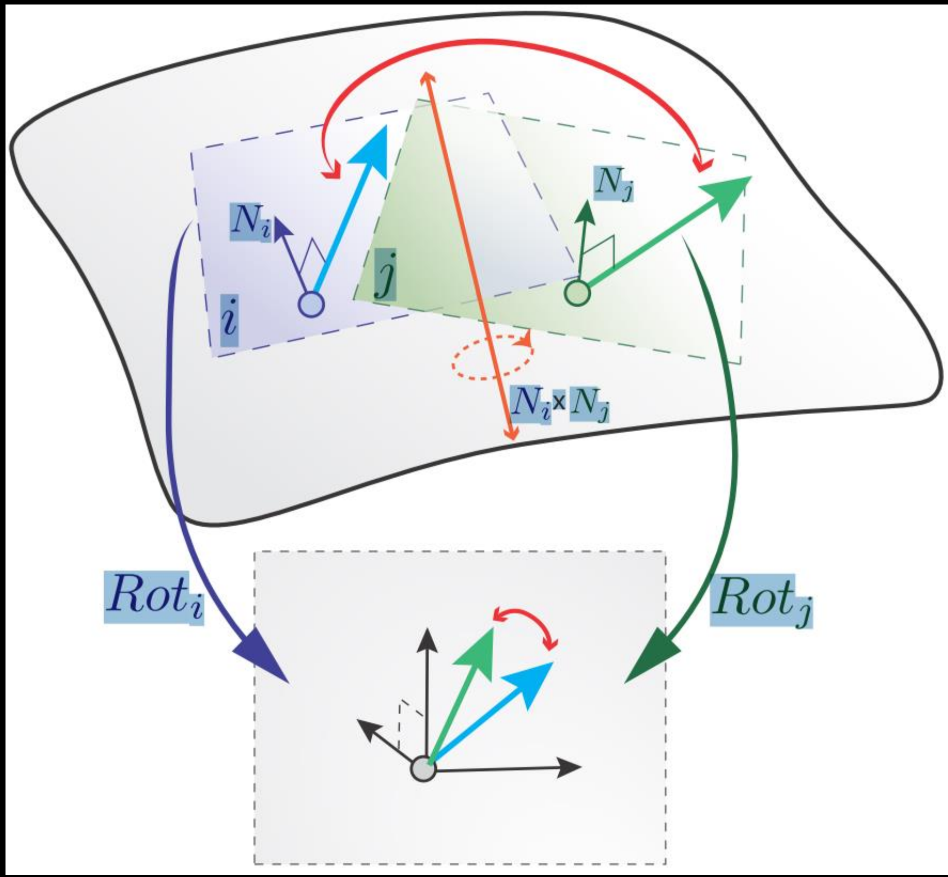
- The tangent spaces of the a triangle mesh can be located on the faces, edges, or vertices of a triangle mesh.

- One way to construct a tangent space at a point is to assign a surface normal vector to the point.
  - Face: normal vector
  - Vertex and edge: local average of the adjacent triangle normal vectors

- Local coordinate system
  - Two orthogonal tangent vectors

Discrete Connections

- Given two adjacent tangent spaces i and j, we need a notion of *connection* between them in order to compare two directional objects that are defined on them.
- A straightforward discretization of the Levi-Civita connection is made by "flattening" the two adjacent tangent planes.
- X_{ij}: angle difference between corresponding axes



Vector Field Topology – continuous

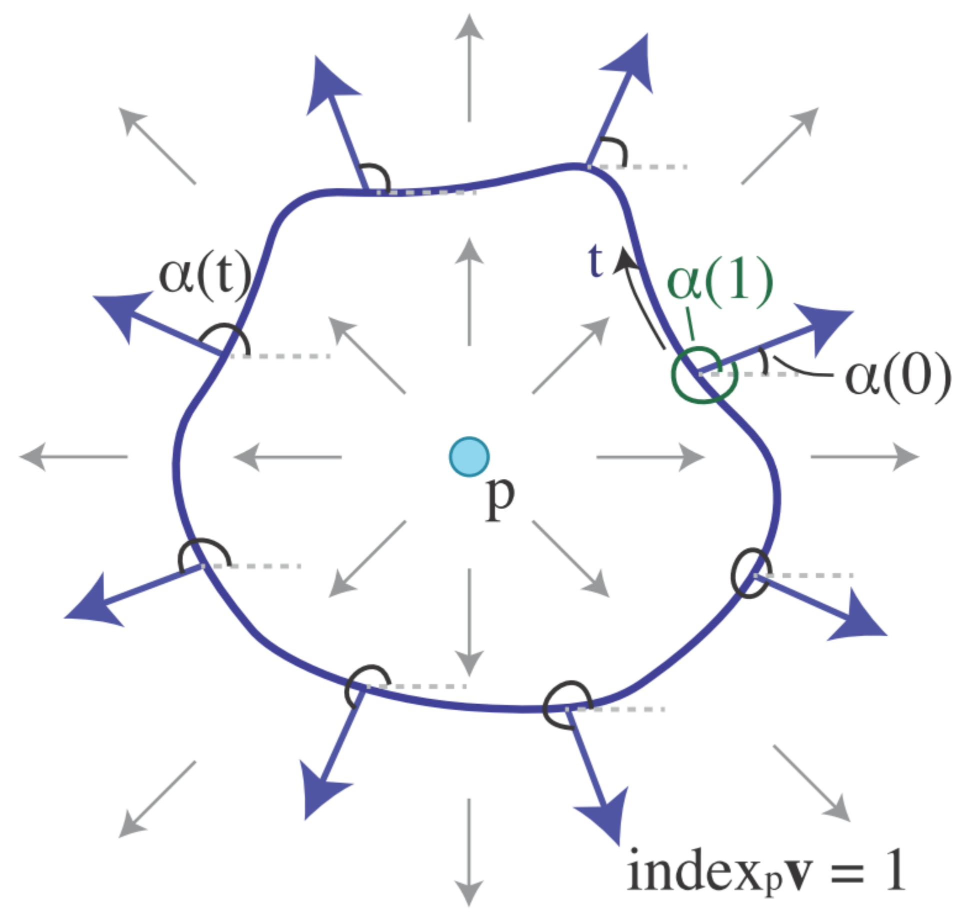
- A vector field has a singularity at a point p if it vanishes or is not defined at this point.
- 2D case:
  - Parameterized curve: $c\colon [0,1] \to R^2$
  - A smooth angle function: $\alpha\colon [0,1] \to R$
  - Vector field:

$$\boldsymbol{v}(c(t)) = \|\boldsymbol{v}(c(t))\| \begin{pmatrix} \cos(\alpha(t)) \\ \sin(\alpha(t)) \end{pmatrix}$$

  - Define the index (an integer) of the singularity at p:

$$\text{index}_p = \frac{1}{2\pi}(\alpha(1) - \alpha(0))$$

Note: Since α is smooth, the difference $\alpha(1) - \alpha(0)$ is unique and it is a multiple of 2π.

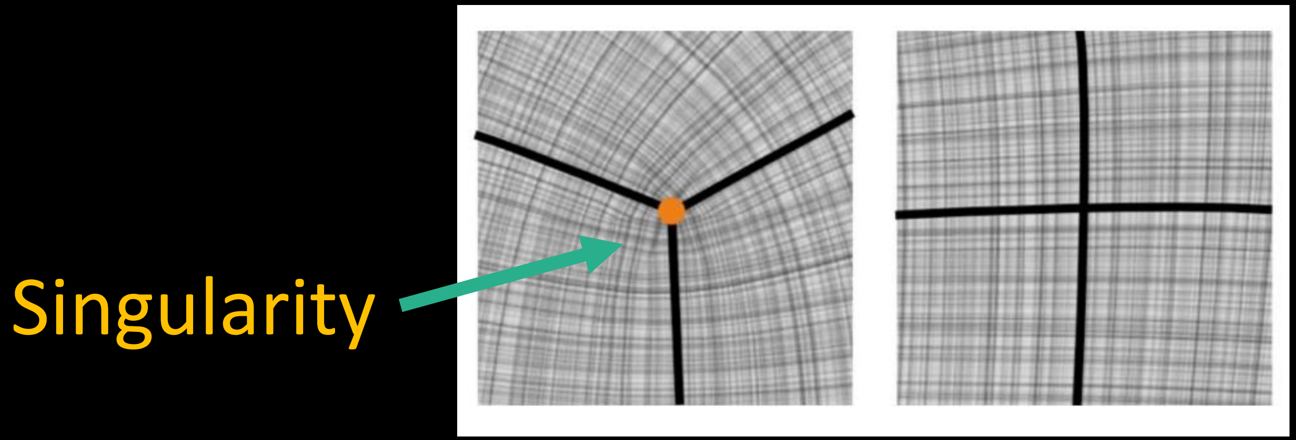


Singularity

$$\text{index}_p = \frac{1}{2\pi}(\alpha(1) - \alpha(0))$$

- The index measures the number of times the vectors along the curve c rotate counterclockwise, while traversing the curve once.

- It is common to consider only points p with index $\text{index}_p \neq 0$ as singular.
  - It vanishes or is not defined at this point.



Singularity

- The definition does not directly extend to surfaces, because there is no global coordinate system (the tangent bundle is not trivial).
- Calculate the index at a point p of a vector field v on a surface M by using an arbitrary chart around p.
- A chart for a topological space M (also called a coordinate chart, coordinate patch, coordinate map, or local frame) is a homeomorphism from an open subset of M to an open subset of a Euclidean space.

Singularity

- Poincaré–Hopf theorem: the sum of all the indices of a vector field equals $2-2g$ for a surface without boundary.
- For N-vector fields, the index is a multiple of $1/N$. Some examples:

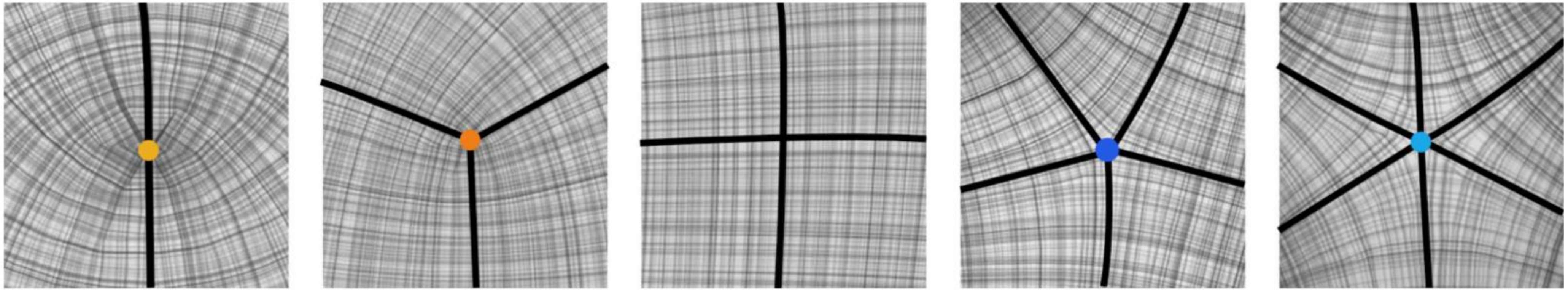
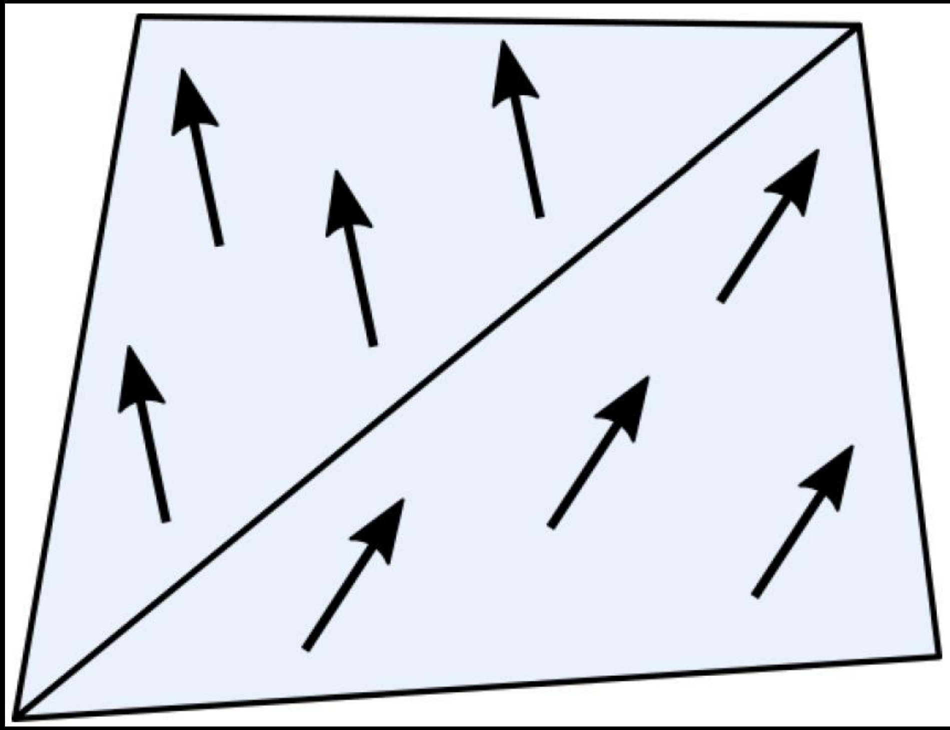


Figure 2: Singularities of index $\frac{1}{2}$, $\frac{1}{4}$, 0 (non-singular), $-\frac{1}{4}$, $-\frac{1}{2}$ in a 4-vector field. The black curves are the so-called separatrices – integral curves (cf. Section 10.3) of the field intersecting the singularity.

Discrete Field Topology

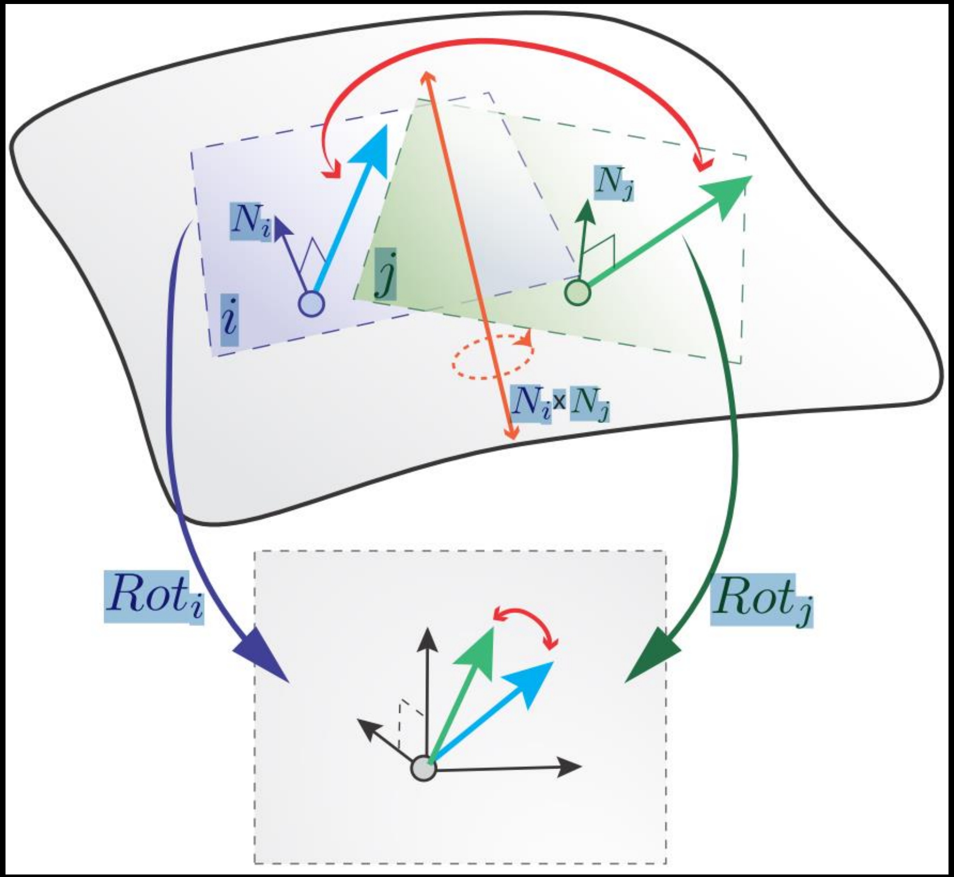
- Piecewise constant face-based 1-direction field
  - Constant on face
  - Discontinuous on edges
- Extension from continuous setting
  - Define rotation between adjacent triangles to define angle difference
    - It is intuitive to assume that the field undergoes a rotation $\delta_{ij} = \frac{\pi}{4}$ clockwise.
    - $\delta_{ij} = \frac{\pi}{4} + 2\pi k$ would be a valid assumption.



Rotation

- Principal rotation
  - $\delta_{ij} \in [-\pi, \pi)$

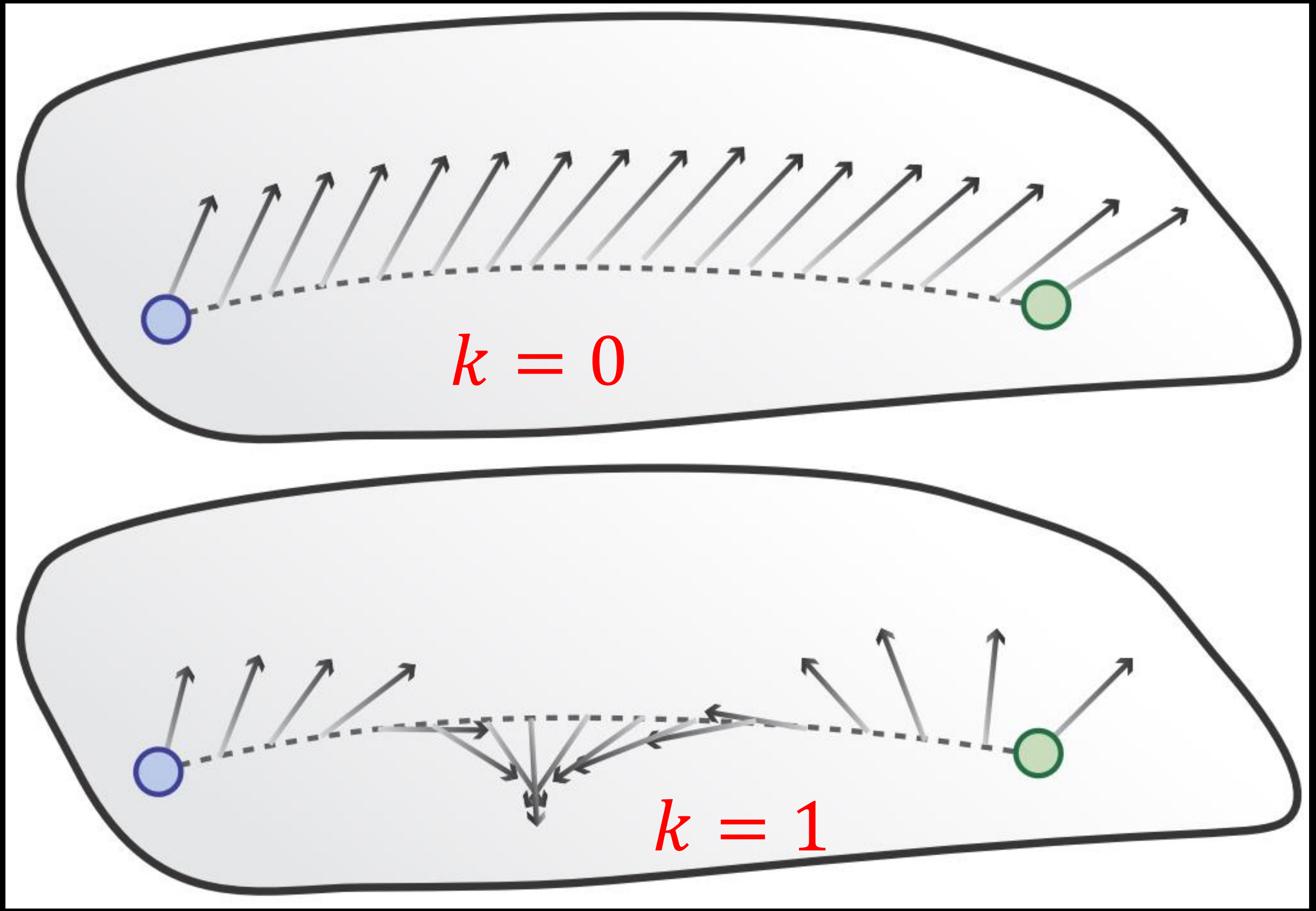
- Summarize all rotation angles on edges that are incident to the vertex
  - $\mathrm{index}_p = \frac{1}{2\pi} \sum \delta_{ij}$



Period Jumps

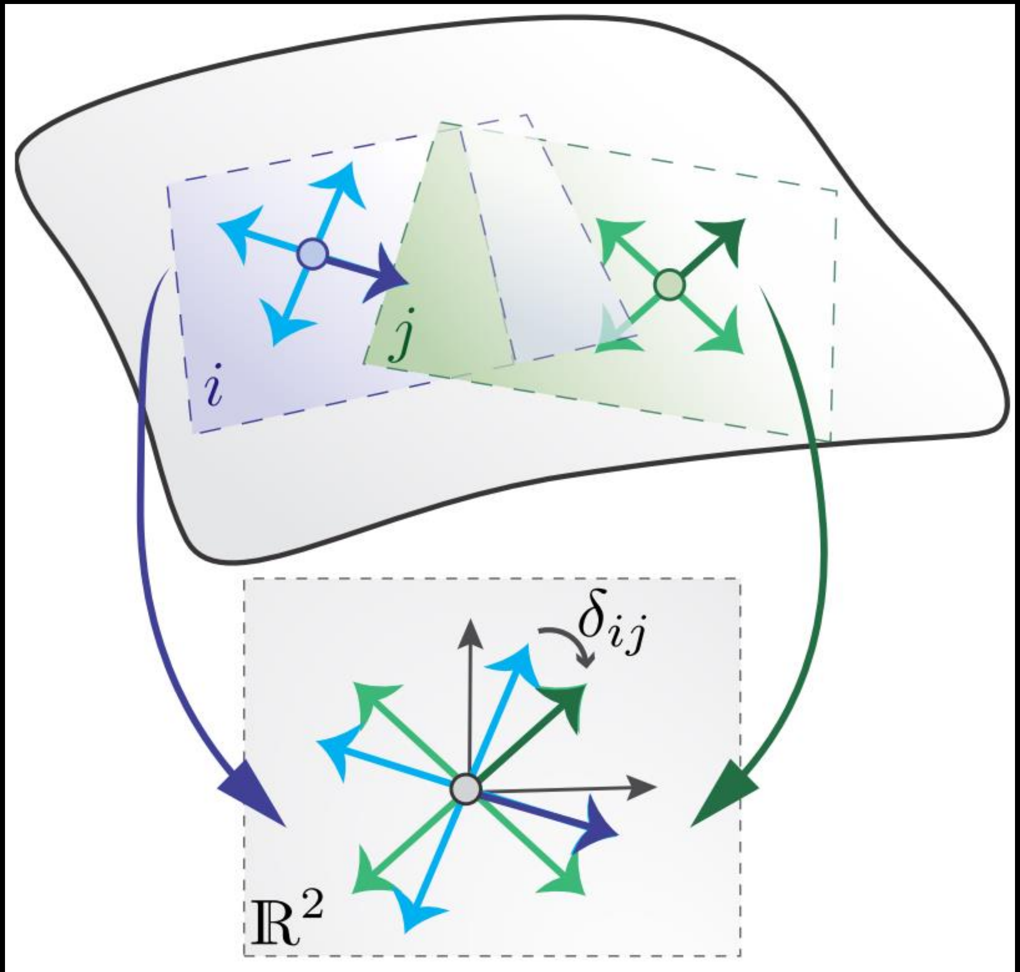
- Non-principal rotation:
  - $\delta_{ij} + 2\pi k$
  - k full period rotations

- The values of k are denoted as *period jumps*.



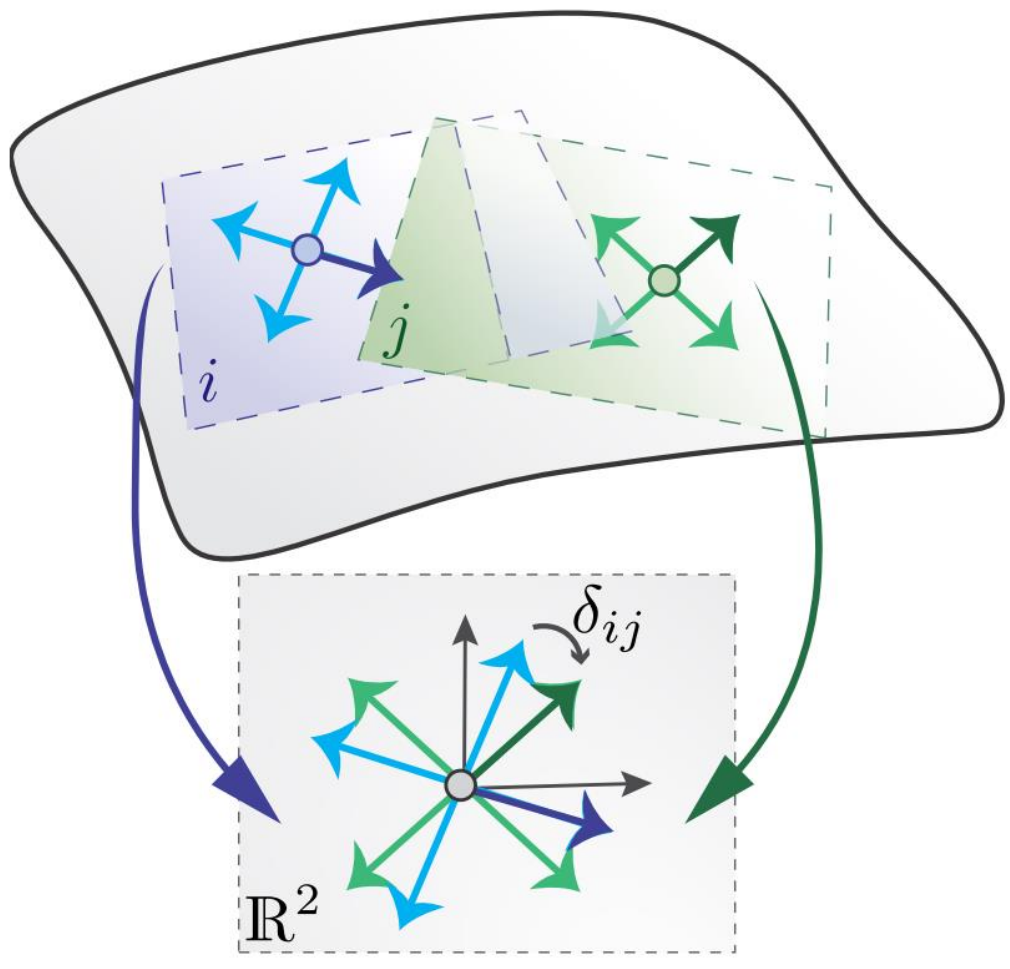
Matching - multi-valued field

- $N > 1$ directionals per tangent space
- An additional degree of freedom:
  - The correspondence between the individual directionals in tangent space i to those in the adjacent tangent space j.
- A *matching* between two N-sets of directional is a bijective map f between them (or their indices).
  - It preserves order: $f(u_r) = v_s \Leftrightarrow f(u_{r+1}) = v_{s+1}$



Effort

- Based on a matching f, the notions of rotation and principal rotation can be generalized to multi-valued fields.
- δ_{ij}^r: rotation between u_r and $f(u_r)$
- *Effort* of the matching f: $Y_{ij} = \sum_{r=1}^{N} \delta_{ij}^r$
- Symmetric N-directional field
  - $\delta_{ij} = \delta_{ij}^r$ for every r
- The efforts of different (order-preserving) matchings differ by 2π.

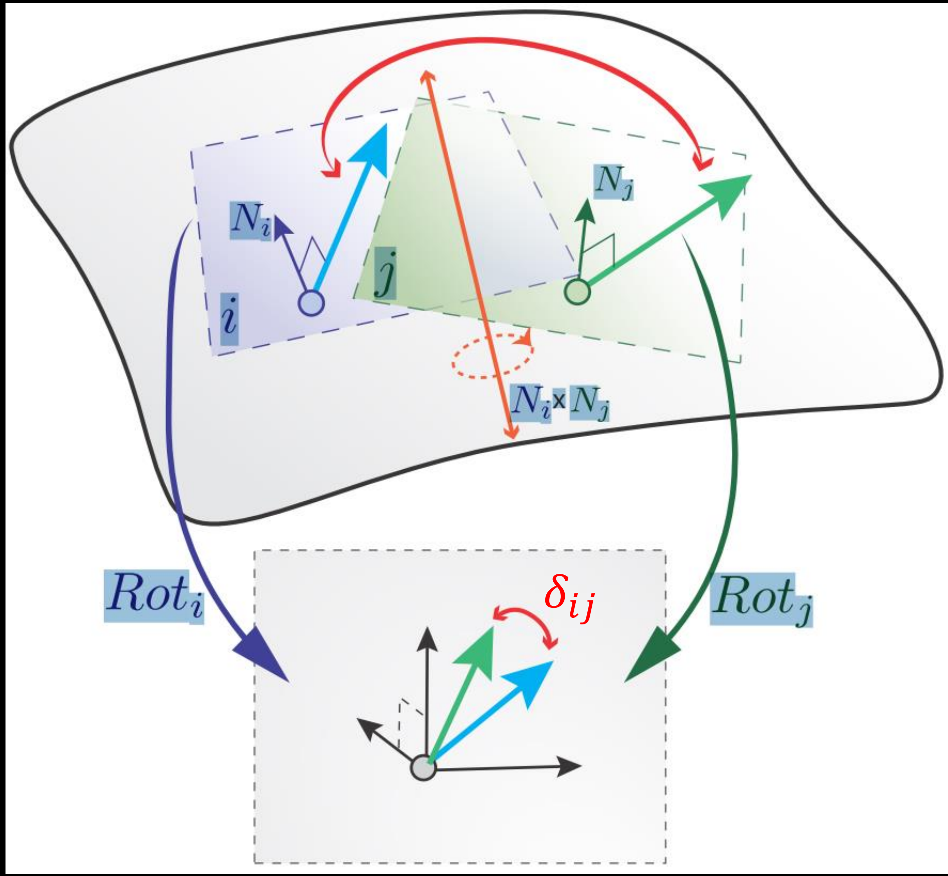


Outlines

- Introduction
- Discretization
- Representation
- Objectives and Constraints

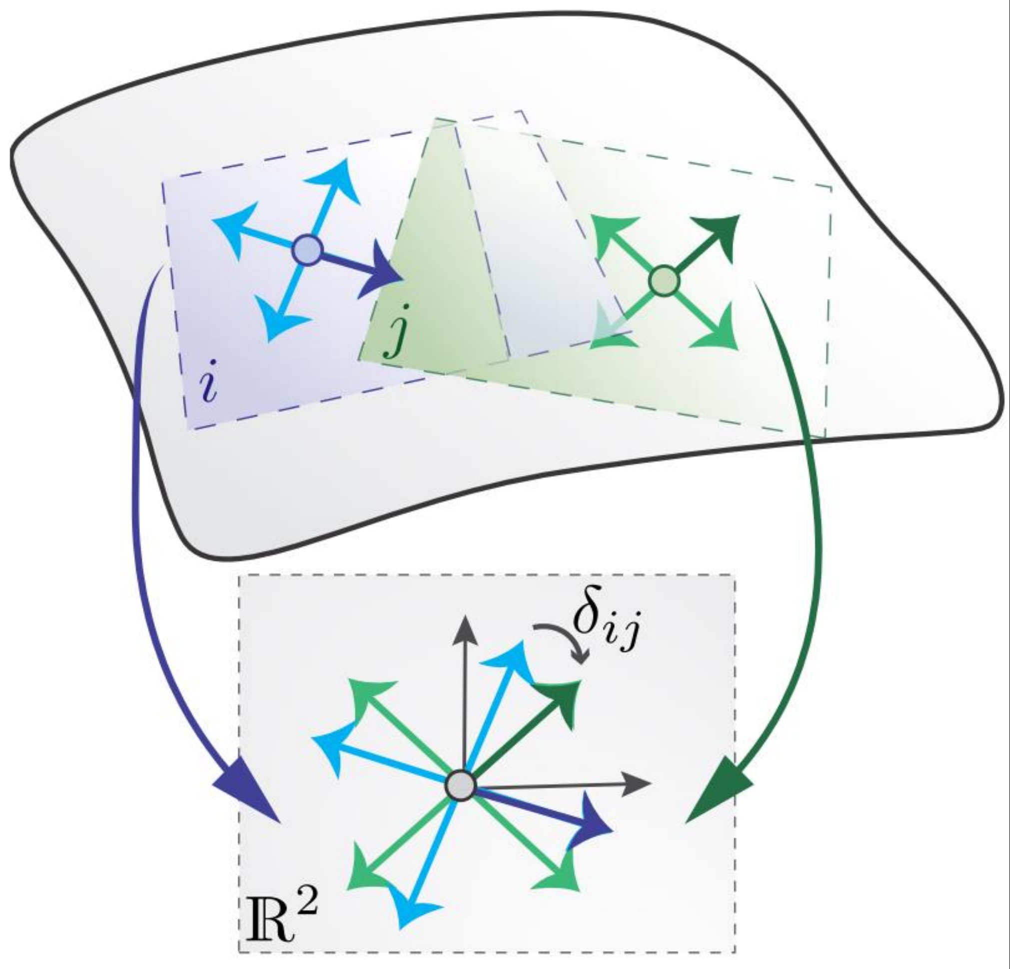
Angle-Based

- Local thonormal frame $\{e_1, e_2\}$ on each tangent space
- 1-direction (unit vector) fields can be concisely described within each tangent space by *a signed angle φ* that is relative to e_1.
- Rotation angle:
  - $\delta_{ij} = \phi_j - (\phi_i + X_{ij} + 2\pi k_{ij})$
  - X_{ij}: the change of bases e between the flattened tangent spaces i and j
  - k_{ij}: period jump



Angle-Based N symmetry directions

- A single φ representing the set of N symmetry directions.
  - $\{\phi + l \cdot 2\pi/N | l \in \{0, \dots, N-1\}\}$
- Period jump to be an integer multiple of $\frac{1}{N}$.
- Rotation angle:
  - $\delta_{ij} = \phi_j - (\phi_i + X_{ij} + \frac{2\pi}{N} k_{ij})$



Pros and cons

- Advantage
  - Directions, as well as possible period jumps, are represented explicitly.
  - A linear expression of the rotation angle.

- Disadvantage
  - The use of integer variables, which leads to discrete optimization problems.

Cartesian and Complex

- A vector v in a two-dimensional tangent space can be represented using Cartesian coordinates (from R^2) in the local coordinate system $\{e1, e2\}$, or equivalently as complex numbers (from C).

- Connection to angle-based representation

$$v = \begin{pmatrix} \cos\phi \\ \sin\phi \end{pmatrix} = e^{i\phi}$$

- The change of bases from one tangent space to another

$$\begin{pmatrix} \cos X_{ij} & -\sin X_{ij} \\ \sin X_{ij} & \cos X_{ij} \end{pmatrix} \text{ or } e^{iX_{ij}}$$

N-directional fields

- By multiplying the argument of the trigonometric functions, or taking the complex exponential to the power of N
$$v^N = \begin{pmatrix} \cos N\phi \\ \sin N\phi \end{pmatrix} = e^{iN\phi}$$
- $e^{iN\phi_l} = e^{iN\phi}$
    - $\phi_l = \phi + l \cdot \frac{2\pi}{N}, \forall l \in \{0, \dots, N-1\}$
    - v^N becomes a 1-directional field.

- X_{ij} becomes NX_{ij}:
$$\begin{pmatrix} \cos NX_{ij} & -\sin NX_{ij} \\ \sin NX_{ij} & \cos NX_{ij} \end{pmatrix} \text{ or } e^{iNX_{ij}}$$

Complex Polynomials

- Analogously, every N-vector set $\{u_1, \dots u_N\}$, in the complex form $u_i \in C$, can be uniquely identified as the roots of a complex polynomial $p(z) = (z - u_1) \dots (z - u_N)$.

- Writing p in monomial form, $p(z) = \sum_i c_n z^n$, the coefficient set $\{c_n\}$ is thus an order-invariant representative of a 1 N–vector.
  - N-PolyVector
  - A generation of former representation.

Comparison

- Comparing between PolyVectors on adjacent tangent spaces amounts to comparing polynomial coefficient.

- Every coefficient c_n contains multiplications of $N - n$ roots.

Tensors

- Real-valued 2×2 matrices in local coordinates

$$T = \begin{pmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{pmatrix}$$

- Symmetric Tensors
  - an eigen-decomposition $T = U\Lambda U^T$
  - $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2)$, two real eigenvalues
  - $U = [u_1, u_2]$, two (orthogonal) eigenvectors with $||u_i|| = 1$
  - Since eigenvectors are only determined up to sign, a rank-2 tensor field can in fact be interpreted as two orthogonal 2-direction fields $\pm u_i$.

Outlines

- Introduction
- Discretization
- Representation
- Objectives and Constraints

Objectives & Constraints

- Different applications have different requirements.
    - various objectives can be used for vector field optimization

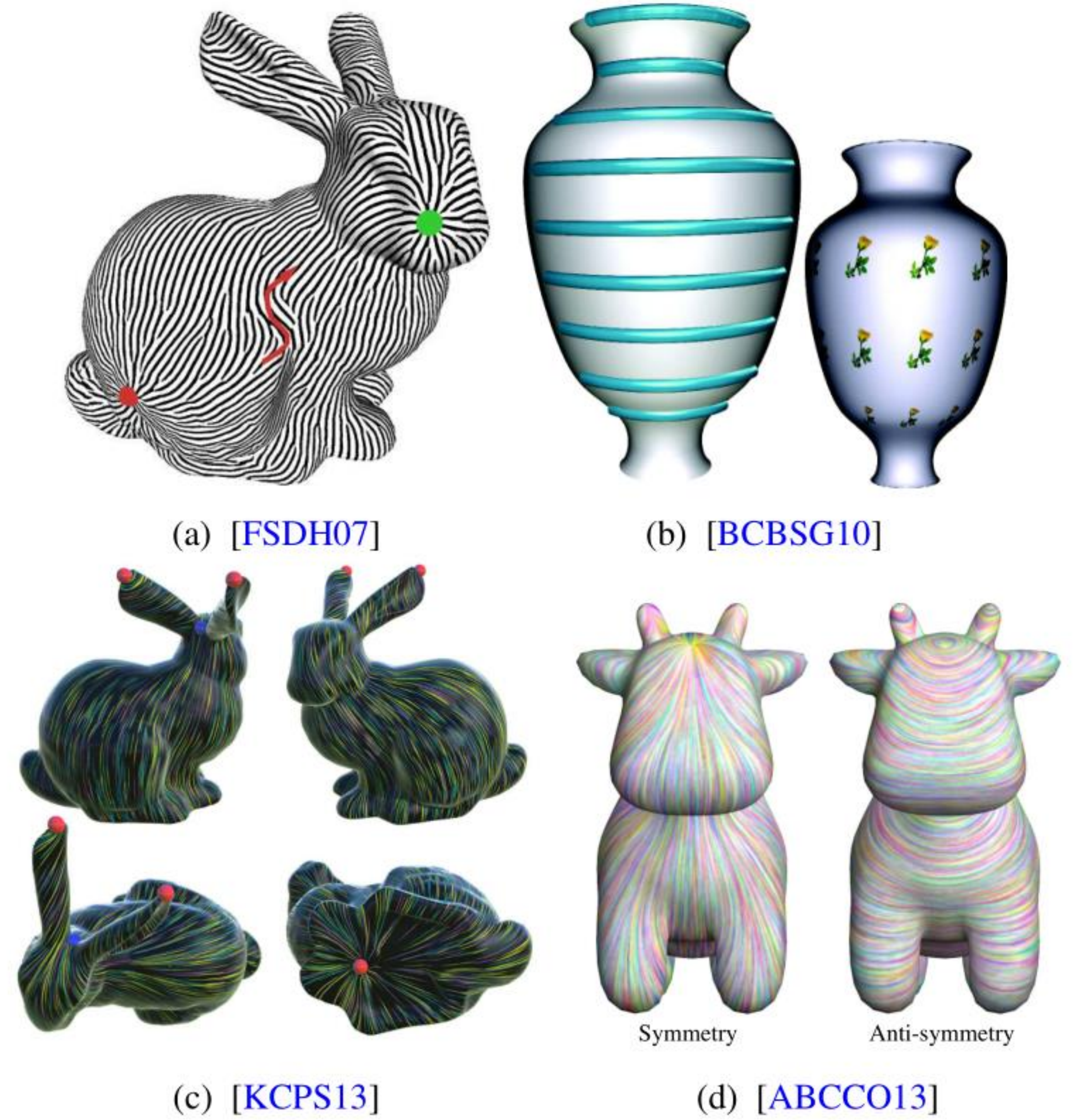


Figure 5: Various objectives used for vector field optimization - (a) alignment to constraints, (b) Killing energy for isometric on-surface deformations, (c) smoothness (Dirichlet energy) (d) commutativity with the symmetry/antisymmetry self maps.

Objectives

- Fairness
  - measuring how variable, or rather, non-similar, the field is between adjacent tangent spaces.
- Parallelity
- Orthogonality
- Minimization of curl
- ……

Parallelity – as-parallel-as-possible

- Parallel
  - The direction in one tangent space is obtained via parallel transport from the directions in adjacent tangent spaces.

- As-parallel-as-possible goal:

$$E_{fair-N} = \frac{N}{2} \sum_{e} w_e (\delta_e)^2$$

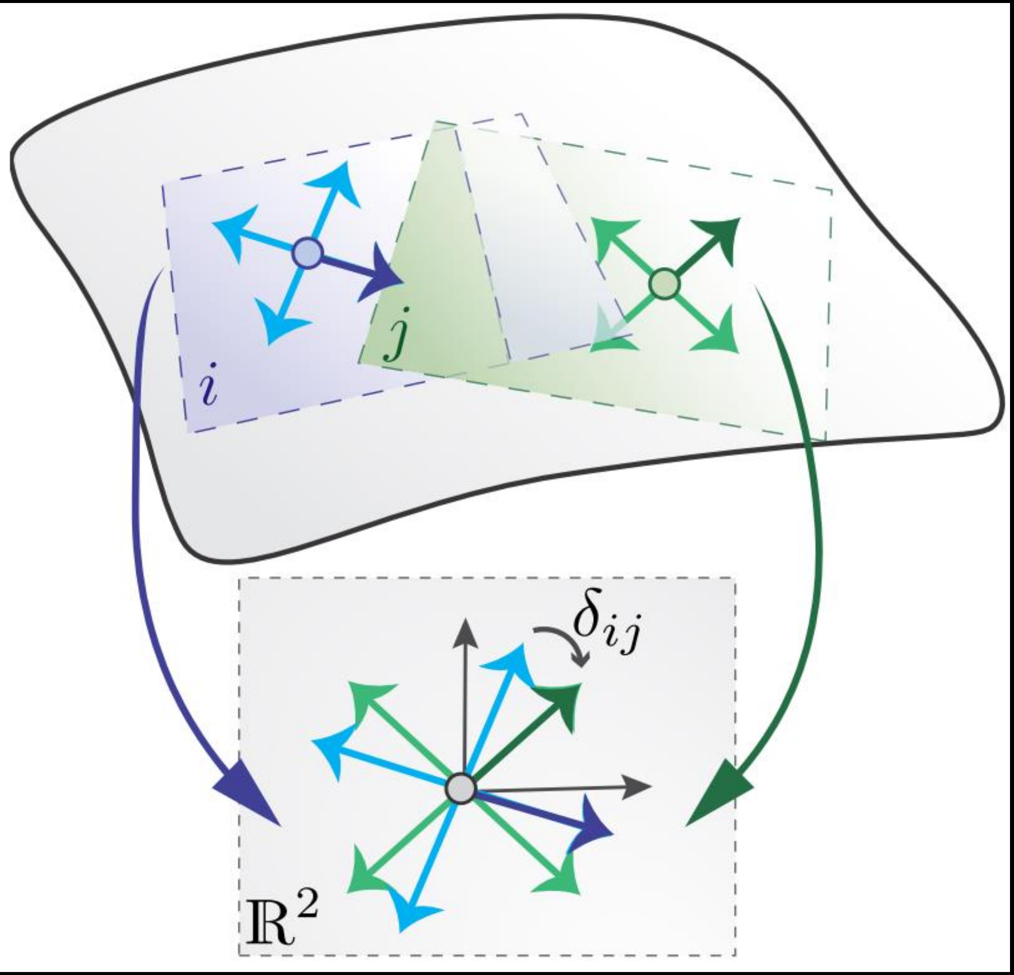
An example of cross field

$$E_{fair-4} = \frac{4}{2}\sum_e w_e \left(\phi_j - (\phi_i + X_{ij} + \frac{2\pi}{N} k_{ij})\right)^2$$

Variables: ϕ_i and k_{ij} (integers)

Greedy solver:

1. Treat k_{ij} as floating number, minimize E_{fair-4}
2. Round the variable which causes the smallest absolute error if we round it to the nearest intege
3. Repeat above two steps.

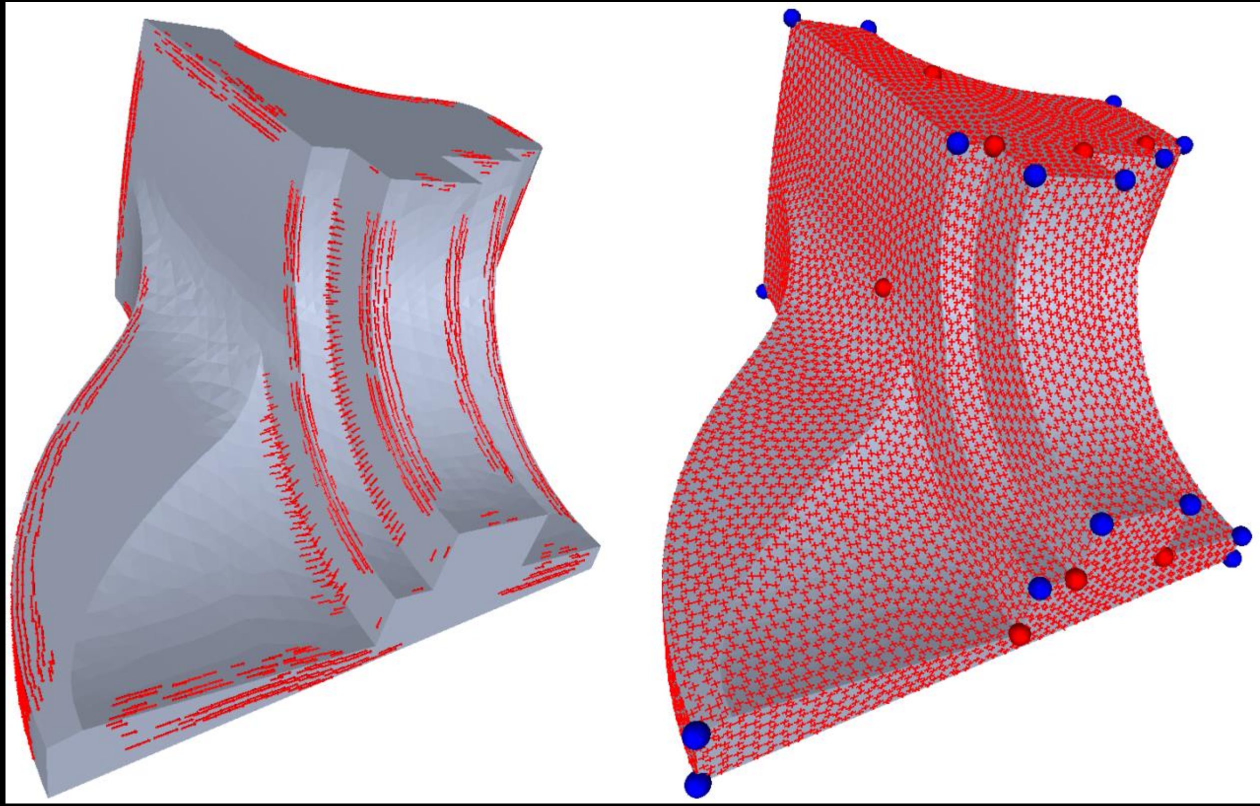


Constraints

- Alignment
  - fit certain prescribed directions
    - principal curvature
    - strokes given by an artist on the surface
    - boundary curves
    - feature lines
  - Soft data term
    - Least square
  - Hard constraint

Constraints

- Alignment
  - fit certain prescribed directions
    - principal curvature
    - strokes given by an artist on the surface
    - boundary curves
    - feature lines
  - Soft data term
    - Least square
  - Hard constraint



Constraints

- Symmetry:
  - If the surface has bilateral symmetry, it is advantageous if the designed directional fields adhere to the same symmetry, allowing field-guided applications to preserve the symmetry as well.

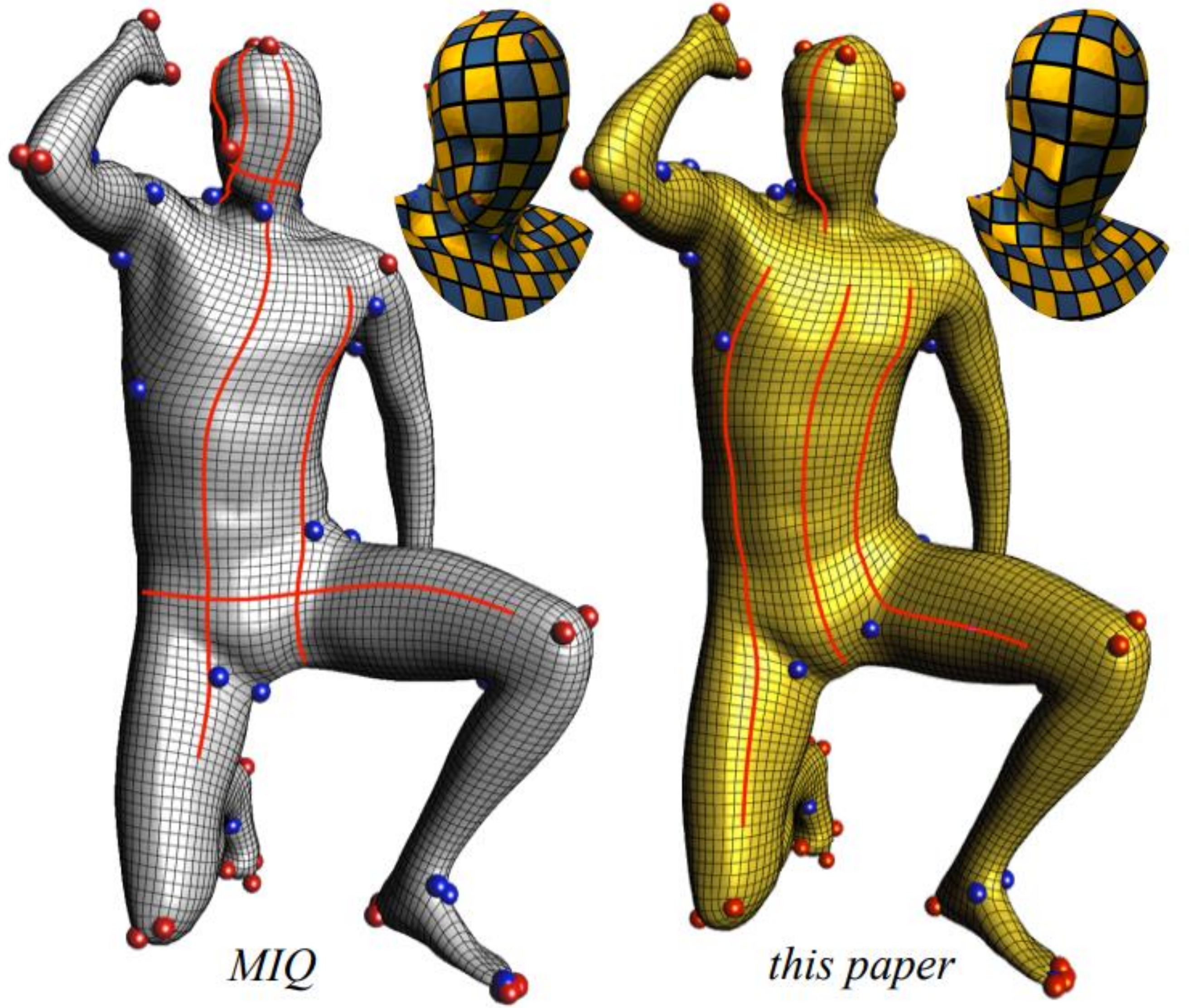


Figure 1: *Field-aligned parametrization of the knelt human model using the symmetry field construction method developed in this paper, and using the MIQ technique of Bommes* et. al.*[2009]. Red/blue bullets represent field singularities with positive/negative index. Red lines trace flows of the cross field.*

Constraints

- Surface mapping:
  - Given multiple shapes with a correspondence between them, we could require that the directional fields commute with the correspondence, effectively designing directional fields jointly on multiple shapes.

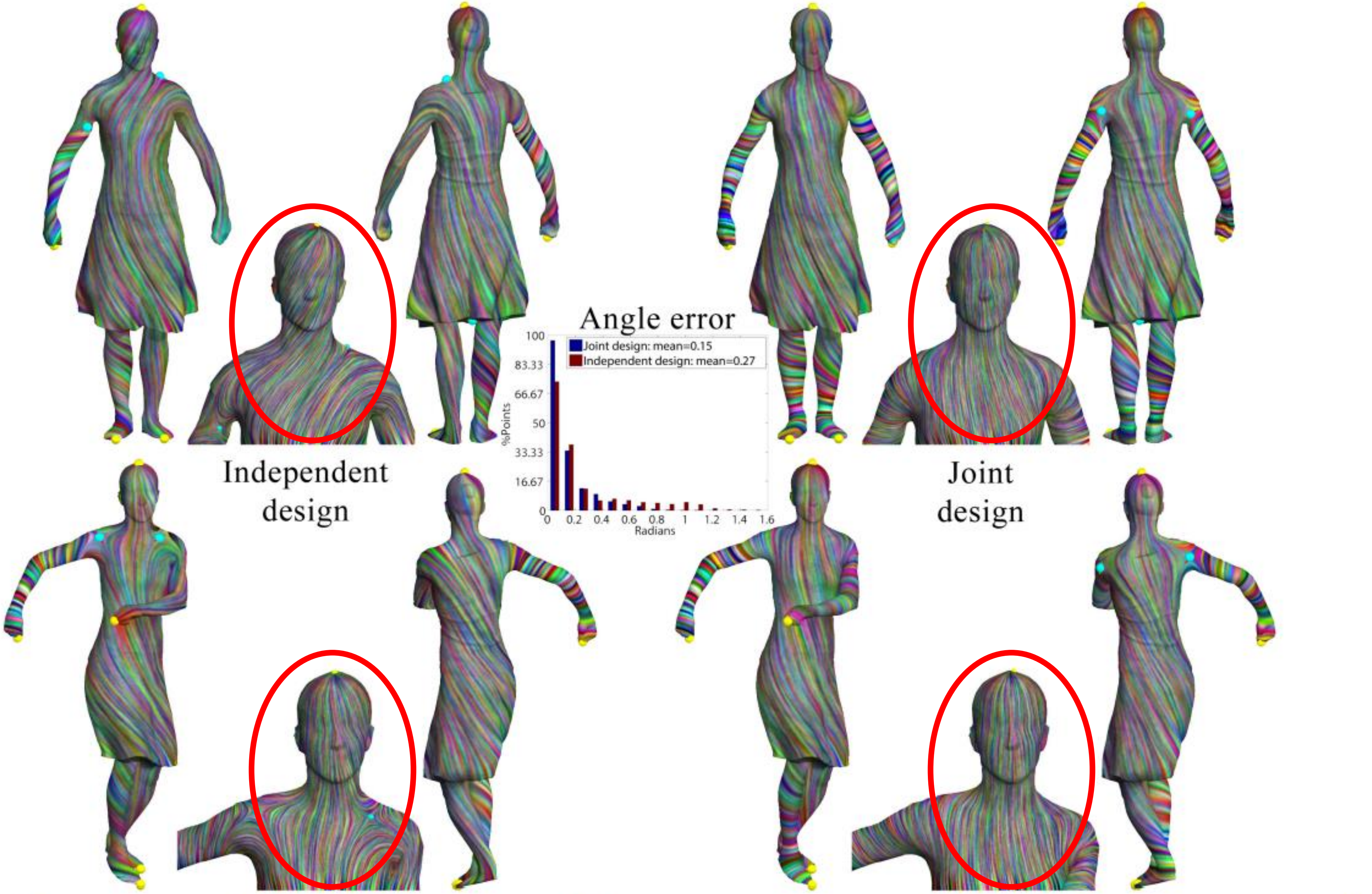


Figure 12: *(left) Independent design on two shapes which are in correspondence does not yield a consistent vector field, even if compatible constraints are used. (right) Solving jointly using our framework yields consistent vector fields (note the corresponding locations of the singularities on the back of the shape). See the text for details.*

Integrable field

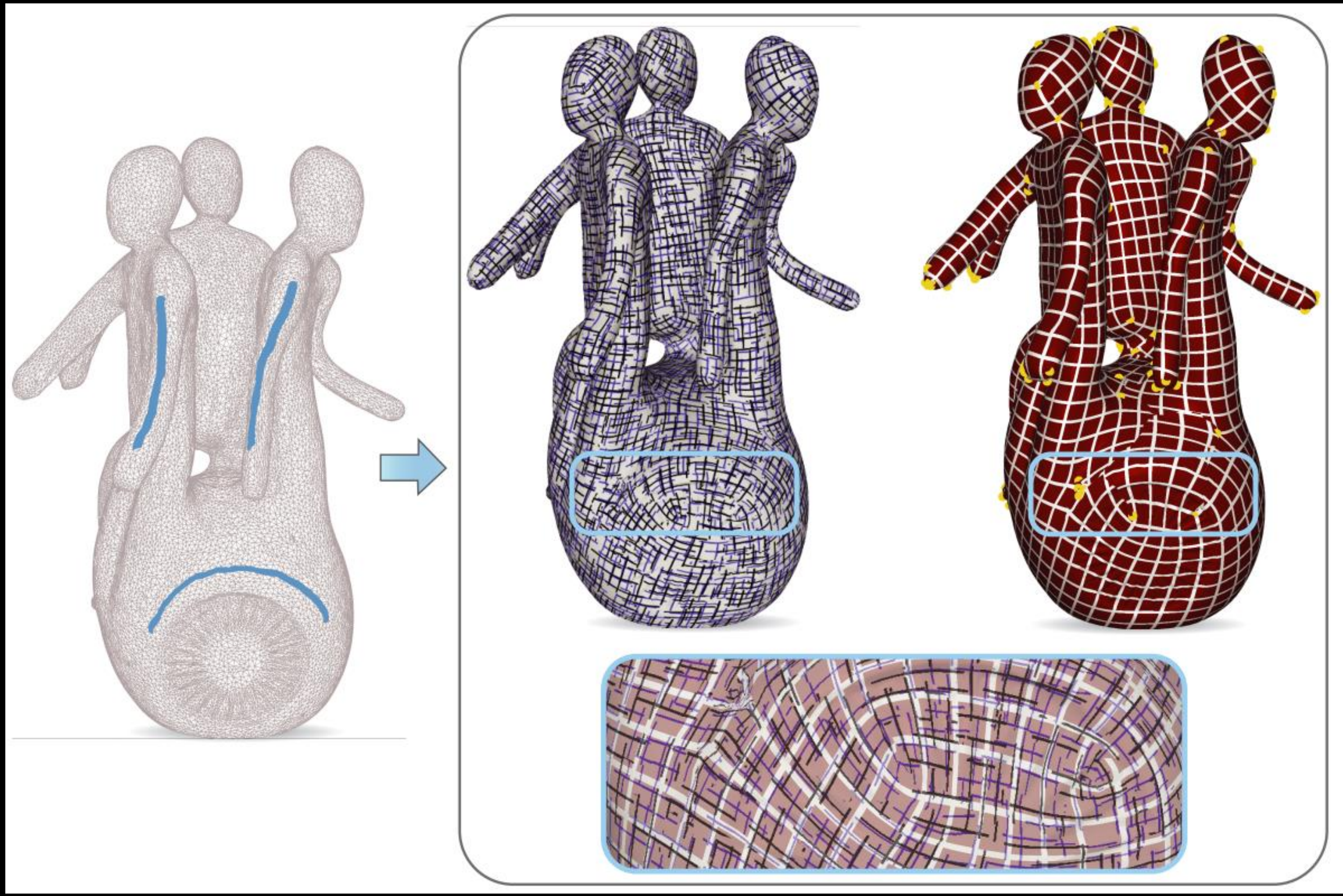
- In most of these applications, vector fields are computed to serve as a guiding basis for the construction of global parameterizations.

- Parameterization coordinates:
  - Defined on vertices
  - Two scalar variables

- Gradients of parameterization coordinates are two separate vector fields.
  - Integrable
  - Curl-free

Integrable field

- Thinking from a opposite way:
  - If the vector fields are curl-free, it can be integrated to be parameterization coordinates.
  - Minimize the difference between the tangent field and the gradient of the function in the least-squares sense.
  - Thus, if the curl-free field is foldover-free and with low distortion, the parameterization is also foldover-free and with low distortion.



Integrability

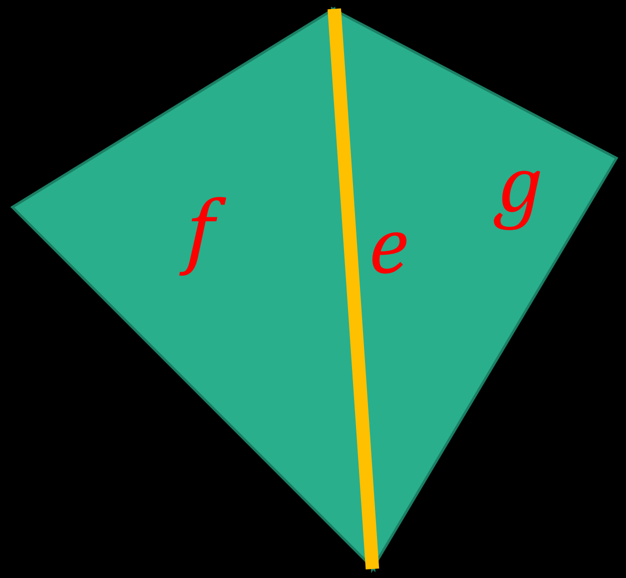
Circulation: $\oint_L \; v \cdot dr$

- Scalar function $h: M \rightarrow R$

$$\langle \nabla h_f, e \rangle = \langle \nabla h_g, e \rangle$$

- It trivially follows that $\langle \nabla h_f, e \rangle - \langle \nabla h_g, e \rangle$ is zero for any function h.

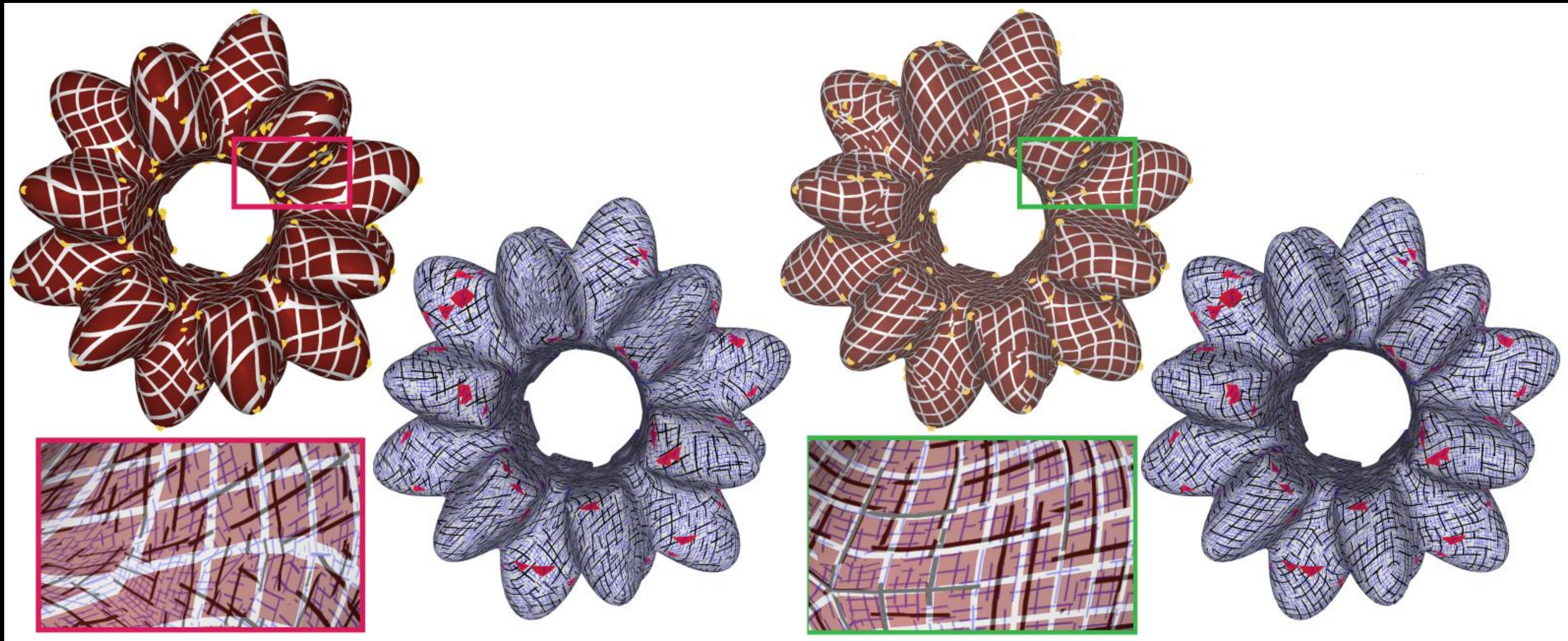
- Discrete curl for any vector field α:
    - $\langle \alpha_f, e \rangle - \langle \alpha_g, e \rangle$
    - α is curl-free if and only if $\langle \alpha_f, e \rangle = \langle \alpha_g, e \rangle$.



Vector field design

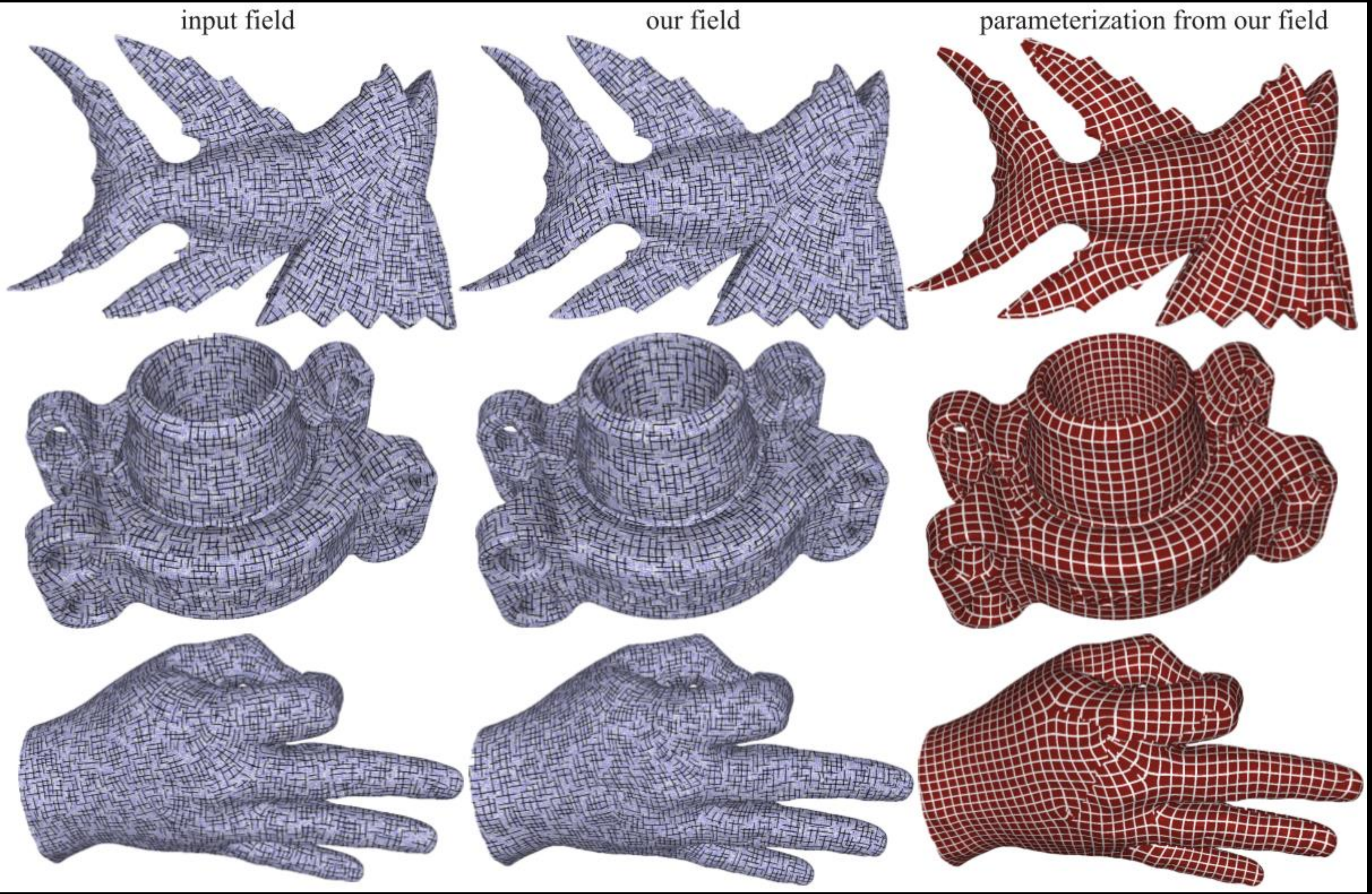
$$J = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} \\ \frac{\partial v}{\partial x} & \frac{\partial v}{\partial y} \end{bmatrix}, \alpha = \left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}\right), \beta = (\frac{\partial v}{\partial x}, \frac{\partial v}{\partial y})$$

- Objective:
  - Fairness
- Constraints:
  - Curl-free
  - Foldover-free
  - Low distortion



Poisson integration

$$\min_h \sum_f \left\| \nabla h_f - \alpha_f \right\|_2^2$$



Extension

- Another question:
  - **Given a cross field, how to modify it to be curl-free?**

- Paper: *Computing inversion-free mappings by simplex assembly*

$$E = E_C + \lambda E_{field}$$

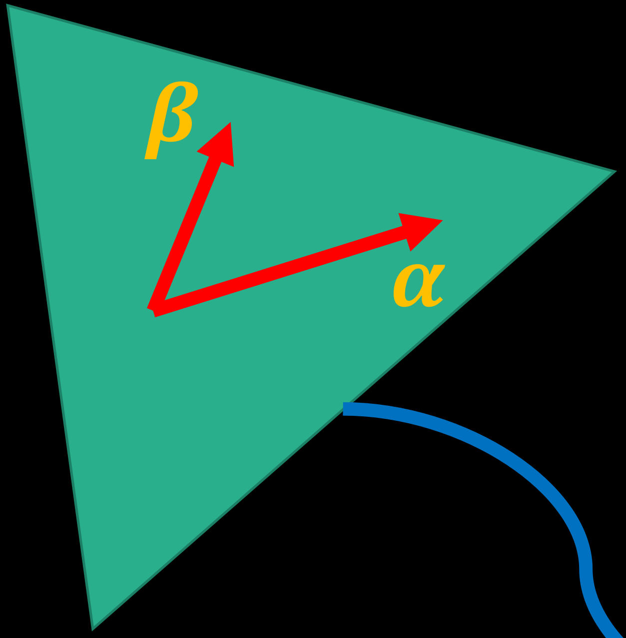
$$E_{field} = \sum_e \left(\langle \alpha_f, e \rangle - \langle \alpha_g, e \rangle\right)^2 + \left(\langle \beta_f, e \rangle - \langle \beta_g, e \rangle\right)^2$$

E_C: distortion energy

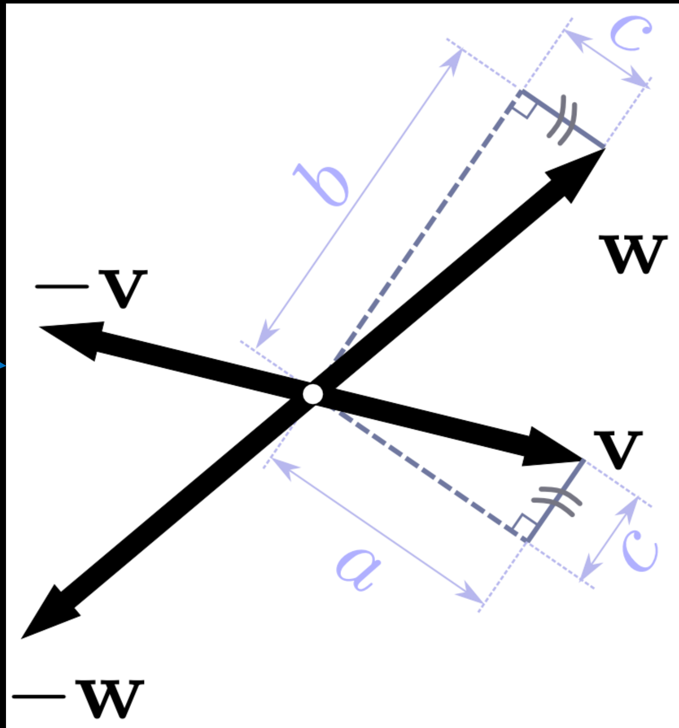
Increase λ to make E_{field} approach zero

Properties of resulting field

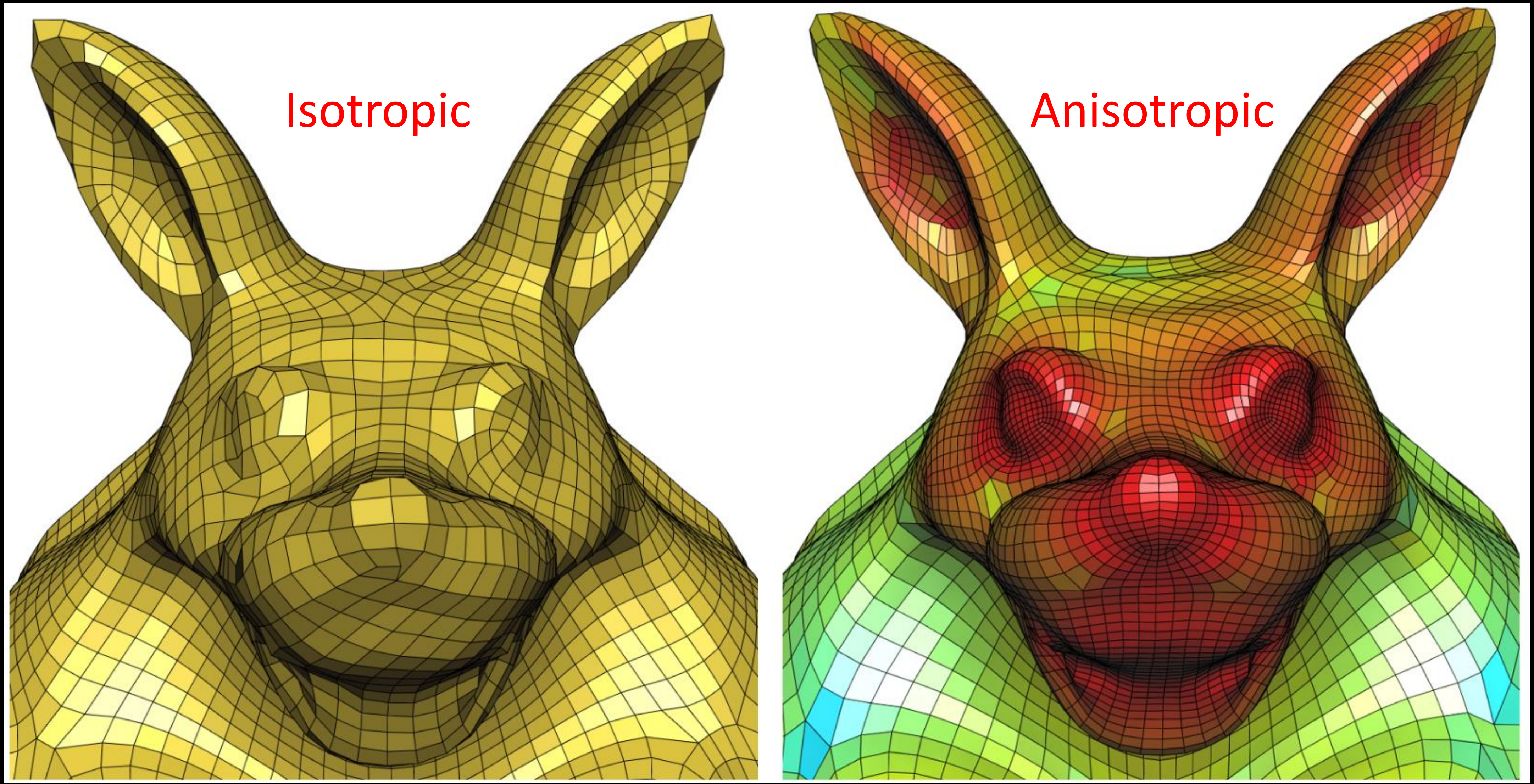
- Non-Orthogonal
- Different lengths



- Frame field

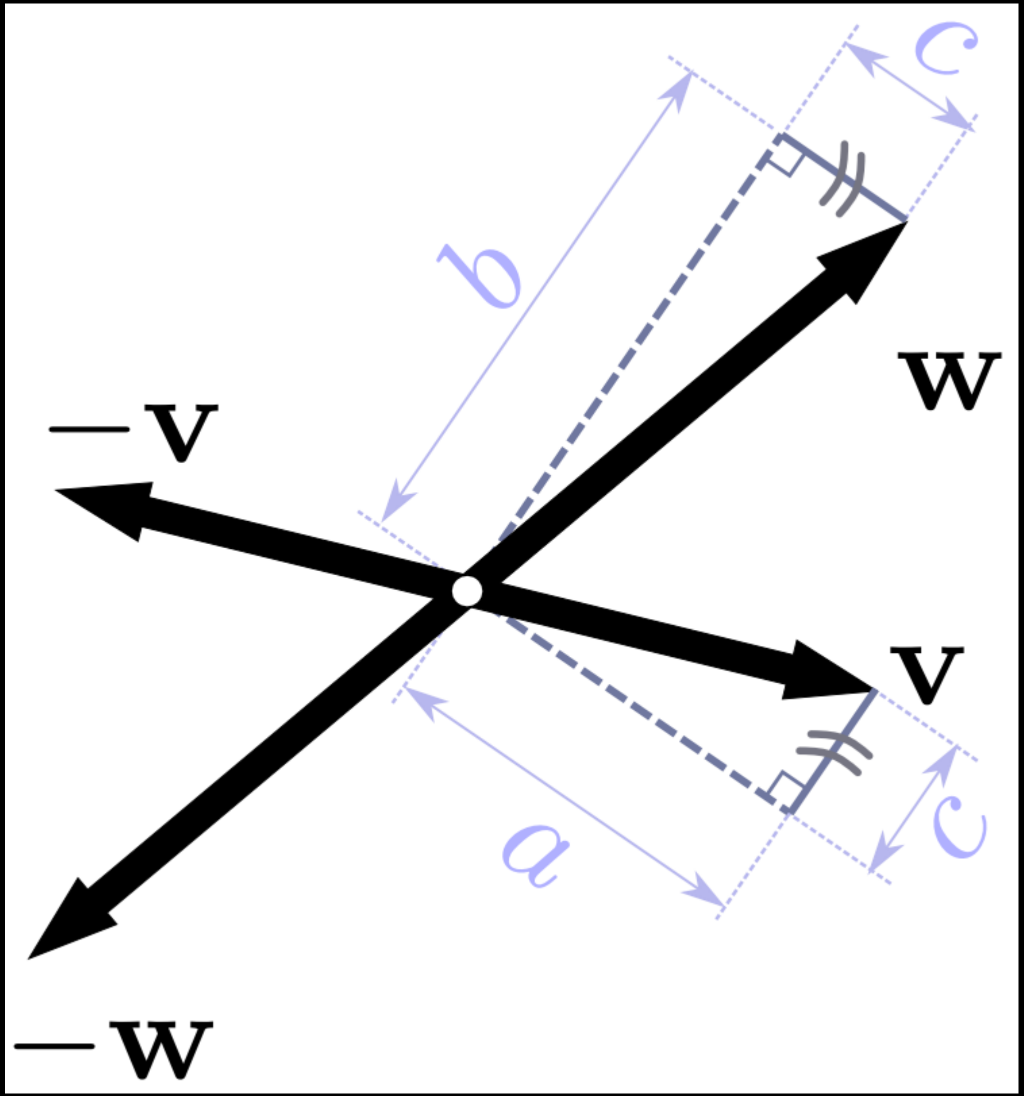


Frame Fields



Frame Fields

- Cross filed (4-RoSy field)
  - $X = \langle \boldsymbol{u}, \boldsymbol{u}^{\perp}, -\boldsymbol{u}, -\boldsymbol{u}^{\perp} \rangle$
- Frame field (2^2 vector field)
  - $F = \langle \boldsymbol{v}, \boldsymbol{w}, -\boldsymbol{v}, -\boldsymbol{w} \rangle$
- Canonical decomposition
  - $F = WX$ (polar decomposition)
  - W: symmetric positive definite matrix

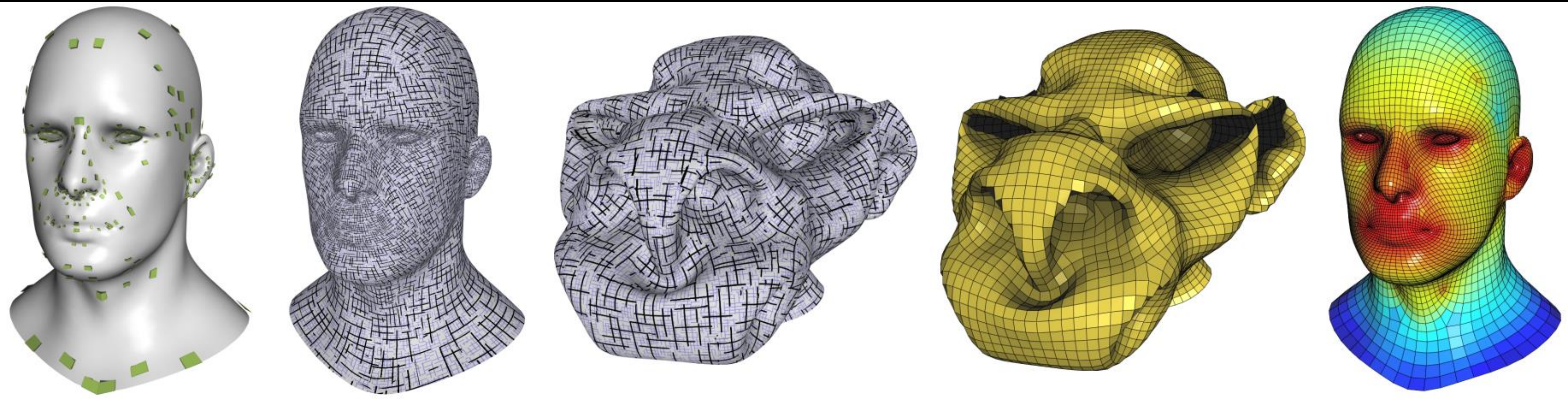


Paper: *Frame Fields: Anisotropic and Non-Orthogonal Cross Fields*

- A frame field is said to be continuous/smooth if both X and W are continuous/smooth.

- Synthesis of frame field:
  - Separately design X and W
  - X: formerly method, e.g., $E_{fair-4} = \frac{4}{2}\sum_e w_e \left(\phi_j - (\phi_i + X_{ij} + \frac{2\pi}{N} k_{ij})\right)^2$
  - W: Laplacian smoothing, guarantee that the resulting W are SPD.

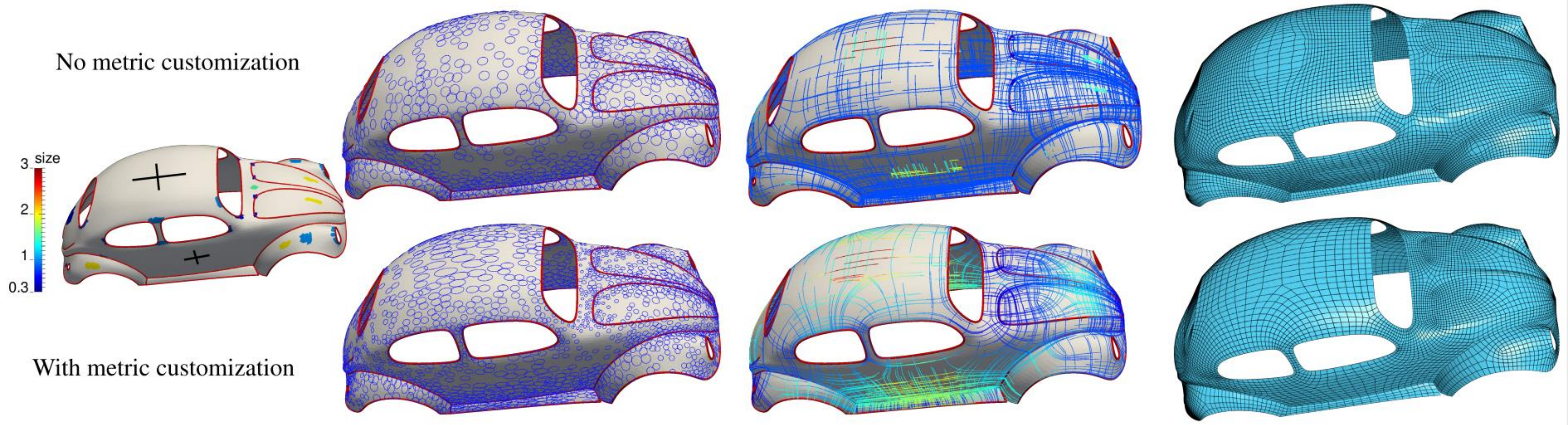
An example



Constraints Frame field Deformation Isotropic Anisotropic

Paper: *Frame Field Generation through Metric Customization*

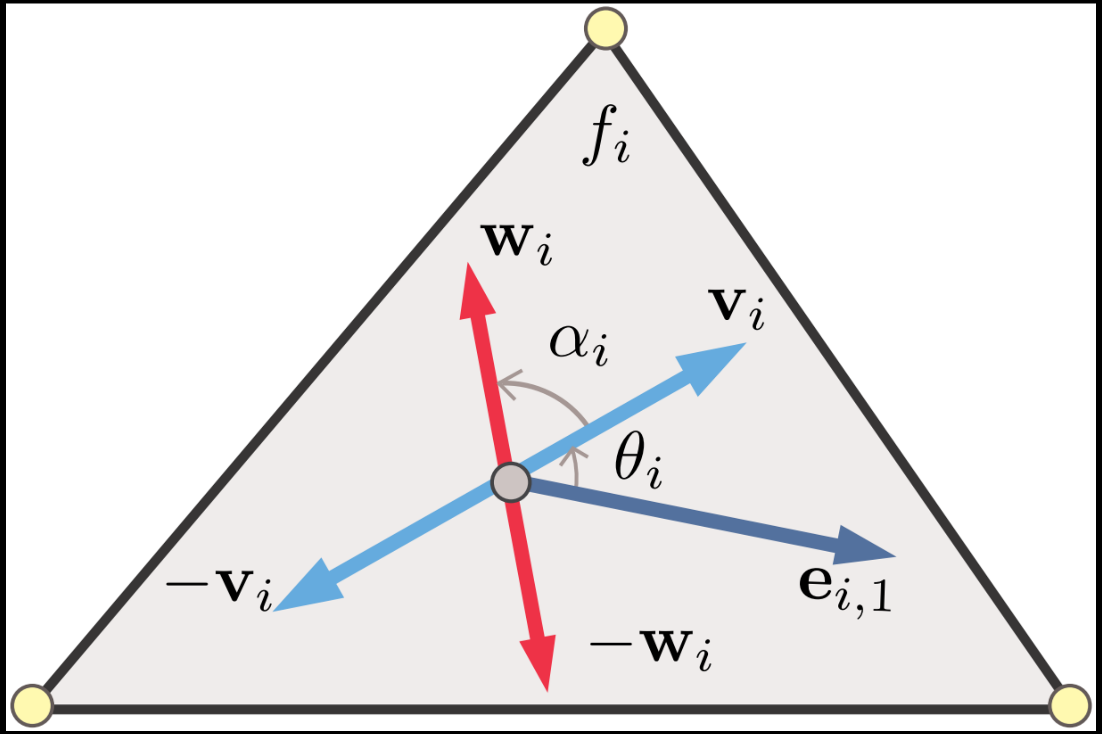
- Generic frame fields (with arbitrary anisotropy, orientation, and sizing) can be regarded as cross fields in a specific Riemannian metric.
  - First compute a discrete metric on the input surface.



2^2 directional field

Paper: General Planar Quadrilateral Mesh Design Using Conjugate Direction Field

- $F = \langle \boldsymbol{v}, \boldsymbol{w}, -\boldsymbol{v}, -\boldsymbol{w} \rangle$
  - $\|\boldsymbol{v}\| = \|\boldsymbol{w}\|$
- Equivalent classes using permutation
  - $\langle \boldsymbol{v}, \boldsymbol{w}, -\boldsymbol{v}, -\boldsymbol{w} \rangle$
  - $\langle \boldsymbol{w}, -\boldsymbol{v}, -\boldsymbol{w}, \boldsymbol{v} \rangle$
  - $\langle -\boldsymbol{v}, -\boldsymbol{w}, \boldsymbol{v}, \boldsymbol{w} \rangle$
  - $\langle -\boldsymbol{w}, \boldsymbol{v}, \boldsymbol{w}, -\boldsymbol{v} \rangle$
- Signed-permutation matrix group G
  - $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}, \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$

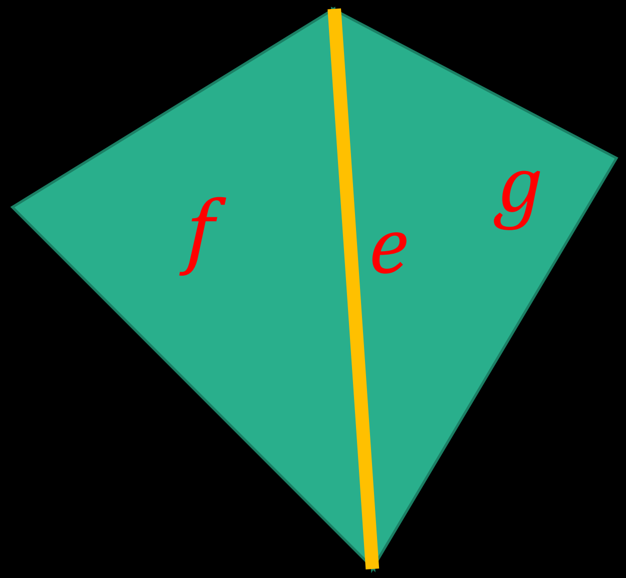


Smoothness of 2^2 directional field

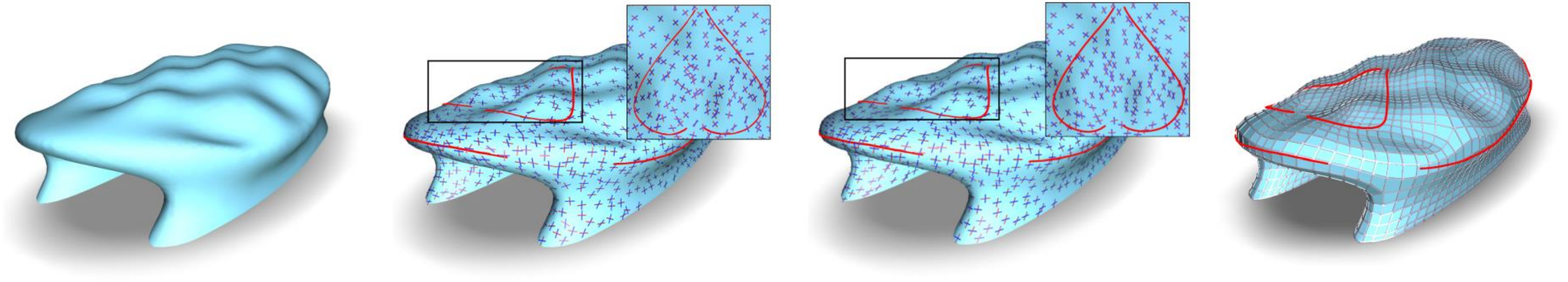
- Transformation between adjacent faces

$$[v_f|w_f] = [v_g|w_g]P_{fg}$$

- Smoothness measure
  - closeness from P_{fg} to G
  - $E_{fg} = \sum_i \left(H\left(P_{fg}[\cdot,i]\right) + H(P_{fg}[i,\cdot])\right) + \sum_i \left(\left(P_{fg}[\cdot,i]^2 - 1\right)^2 + \left(P_{fg}[i,\cdot]^2 - 1\right)^2\right) + \left(\det P_{fg} - 1\right)^2$
  - $\mathrm{H}(\eta) = \eta_x^2\eta_y^2 + \eta_y^2\eta_z^2 + \eta_z^2\eta_x^2$



Example



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

Figure 3: *CDF design on an airport terminal model. (a) The original model. (b) An initial CDF from user-specified strokes (red lines). (c) The optimized CDF. (d) The resulting PQ mesh.*

Extension to 3D field

Paper: All-Hex Meshing using Singularity-Restricted Field

- $F = \{U, V, W\}$
  - Right-handled: $U \times V \cdot W > 0$

- 24 equivalent classes since the permutations form the chiral cubical symmetry group G.
  - First vector has six options
  - Second one has four options
  - Third one only has one options
  - $6 \times 4 \times 1 = 24$

- Smoothness measure:
  - closeness from P_{fg} to G
  - Similar to former method

Efficiency

- Former methods:
  - Large-scale sparse linear system or nonlinear energy
  - Expensive: too many variables
  - Global view

- Is there any other ways?
  - **Reducing the variable number**
  - **Starting from local view**

Paper: *Instant Field-Aligned Meshes*

Figure 1: *Remeshing a scanned dragon with 13 million vertices into feature-aligned isotropic triangle and quad meshes with ∼80k vertices. From left to right, for both cases: visualizations of the orientation field, position field, and the output mesh (computed in 71.1 and 67.2 seconds, respectively). For the quad case, we optimize for a quad-dominant mesh at quarter resolution and subdivide once to obtain a pure quad mesh.*

Key techniques

- Gauss-Seidel method

$$v_i \leftarrow \frac{1}{\sum_{j\in\Omega(i)} w_{ij}} \sum_{j\in\Omega(i)} w_{ij} v_j$$

- Multiresolution hierarchy
  - improve convergence
  - allow the algorithm to move out of local minima

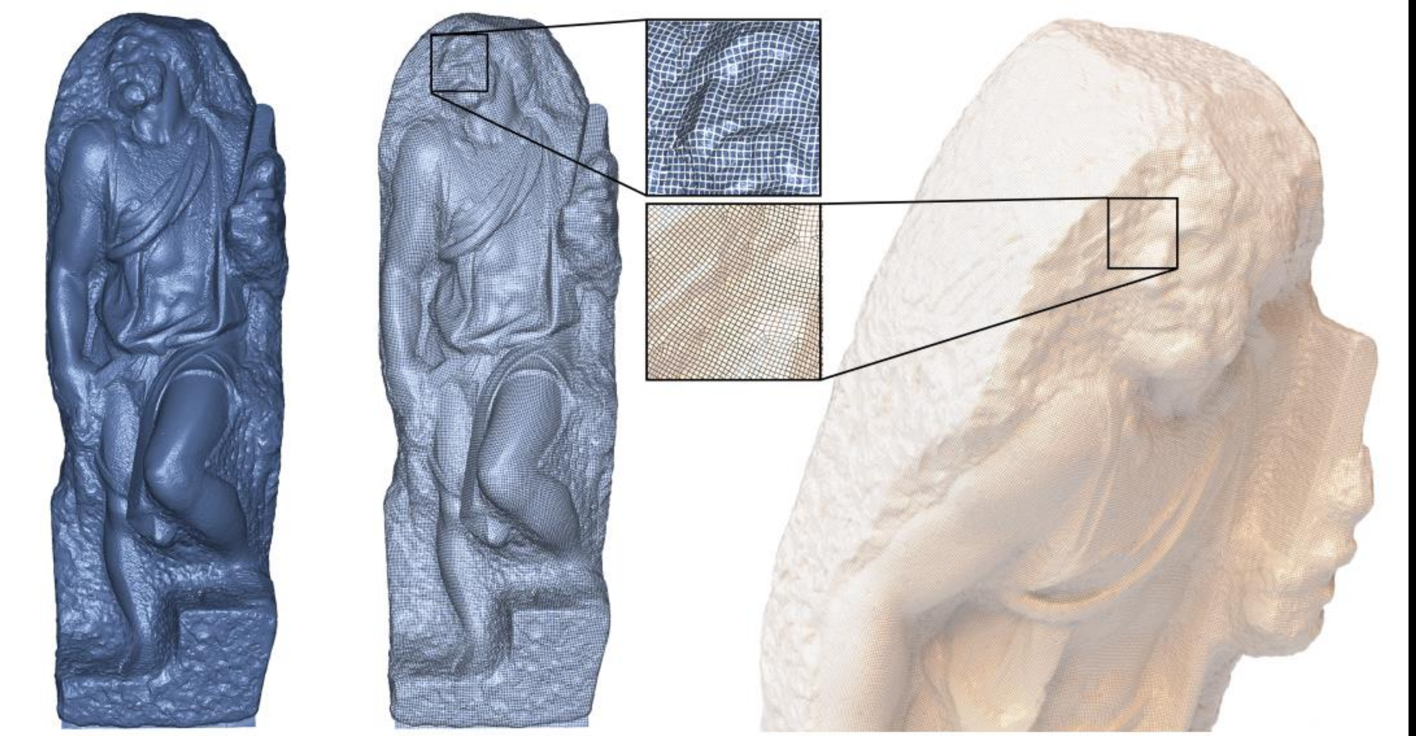


Figure 24: *Our method scales to extremely large datasets, such as the 372M triangle St. Matthew statue acquired by the Digital Michelangelo project [Levoy et al. 2000]. The middle column shows a visualization of the position field, and the right is the final quadrilateral mesh. The entire process takes 9 minutes and 18 seconds.*