VSB TECHNICAL UNIVERSITY OF OSTRAVA | FACULTY OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE | DEPARTMENT OF COMPUTER SCIENCE

Data Visualization

460-4120

Fall 2024

Last update 30. 10. 2024

Tensor Data Visualization

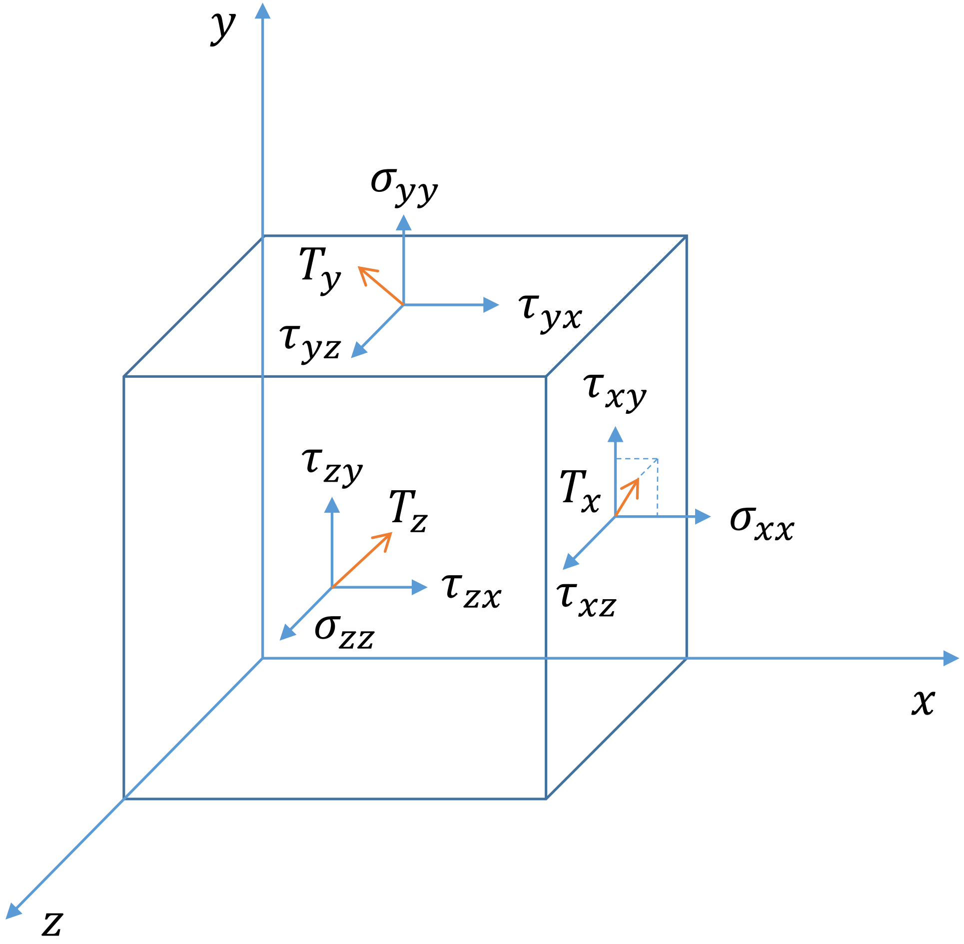
The total number of indices required to identify each component uniquely is equal to the dimension of the array, and is called the order, degree or *rank* of the tensor

- 0th order tensors – scalars (magnitude, 1 value)
- 1st order tensors – vectors (magnitude + direction, e.g. 3 values in 3D space)
- 2nd order tensors – dyads (quantities that have magnitude and two directions, represents variation of magnitude, e.g. 3×3 values)
- 3rd order tensors – triads (e.g. 3×3×3 values)
- 4rd order tensors – Einstein's general relativity required a tensor of rank 4 (x, y, z, t), i.e. 4×4×4×4 = 256 components
- and so on…
- Tensors (together with scalars and vectors) are important quantities in many fields: mechanics, electrodynamics, fluid mechanic, crystal structure, general relativity and others
- But the visualization of tensors is quite challenging

Graphical Representation of Dyad

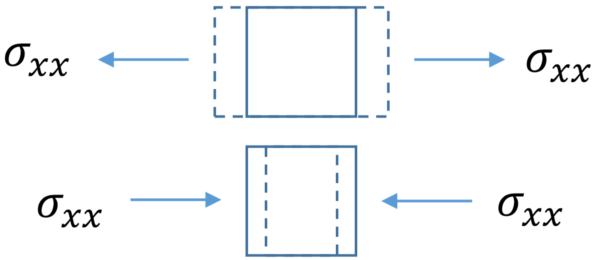
$$\begin{matrix} T_x \\ T_y \\ T_z \end{matrix} \begin{bmatrix} \sigma_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_{zz} \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

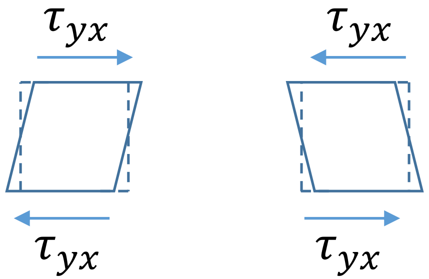


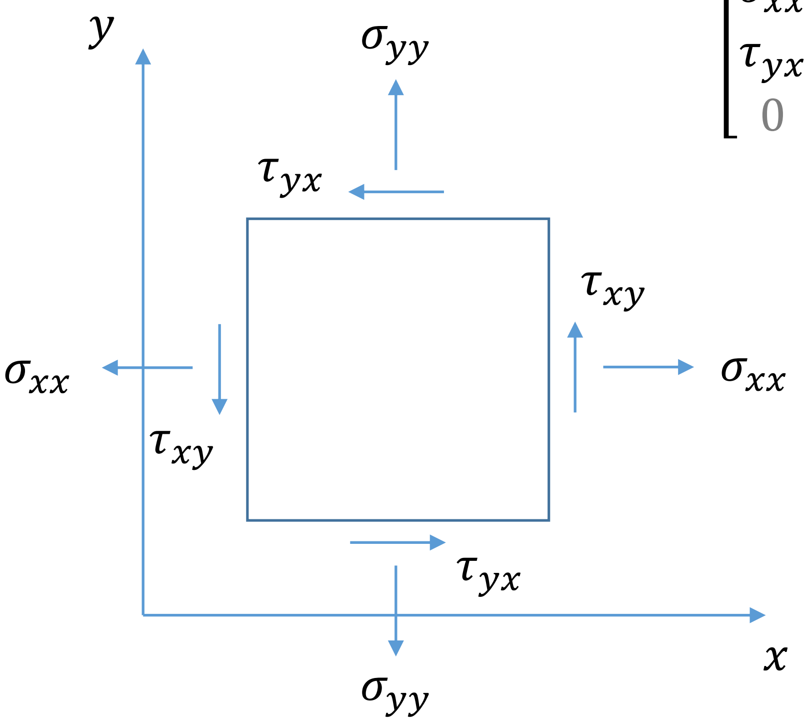
Stress Tensor in 2D

Force acting along the axis perpendiculary to the sides ($\sigma = \frac{F_{\text{normal}}}{A}$) causes stretching ($\sigma_{xx} > 0$ or $\sigma_{yy} > 0$) or compression ($\sigma_{xx} < 0$ or $\sigma_{yy} < 0$)



Force acting along the sides ($\tau = \frac{F_{\text{parallel}}}{A}$) causes shear stress ($\tau_{xy} \neq 0$ or $\tau_{yx} \neq 0$)





$$\begin{bmatrix} \sigma_{xx} & \tau_{xy} & 0 \\ \tau_{yx} & \sigma_{yy} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\sigma_{xx} = \frac{F_x}{A_x}$$

$$\sigma_{yy} = \frac{F_y}{A_y}$$

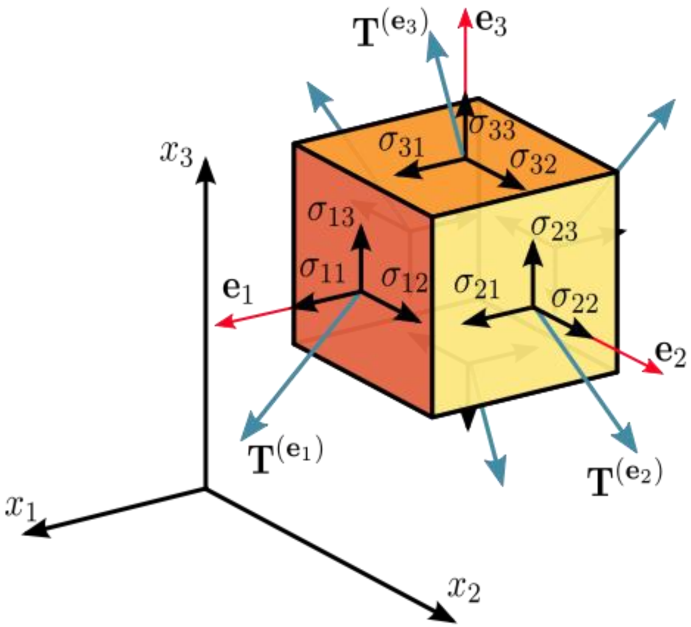
$$\tau_{xy} = \frac{F_y}{A_x}$$

$$\tau_{yx} = \frac{F_x}{A_y}$$

Stress Tensor in 3D

- Cauchy stress tensor – 2nd order tensor
- Completely define the state of stress at a point inside a material

$$\begin{bmatrix} T_1^{(\mathbf{n})} & T_2^{(\mathbf{n})} & T_3^{(\mathbf{n})} \end{bmatrix} = \begin{bmatrix} n_1 & n_2 & n_3 \end{bmatrix} \cdot \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{bmatrix}.$$



Second Order Tensors

- May be defined as an operator that acts on a vector $\boldsymbol{u}$ generating another vector $\boldsymbol{v}$, so that $\mathbf{T}(\boldsymbol{u}) = \boldsymbol{v}$ or $\mathbf{T} \cdot \boldsymbol{u} = \boldsymbol{v}$
- $\mathbf{T}$ is a mapping that takes one vector as input, and gives one vector as output

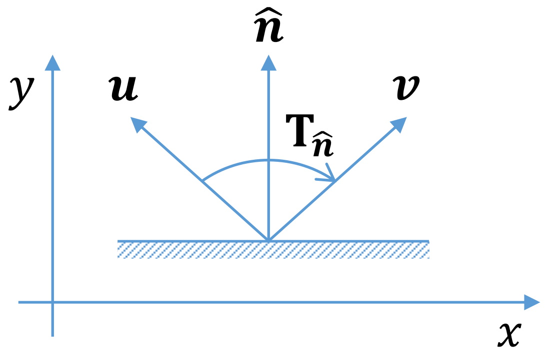
  $\mathbf{T}: V \rightarrow V$, where V is a vector space
- It is a linear operator (linear transformation) so it holds that $\mathbf{T}$ is
  - distributive $\mathbf{T}(\boldsymbol{a} + \boldsymbol{b}) = \mathbf{T}\boldsymbol{a} + \mathbf{T}\boldsymbol{b}$
  - associative $\mathbf{T}(k\boldsymbol{a}) = k\mathbf{T}(\boldsymbol{a})$

  for all $\boldsymbol{a}, \boldsymbol{b} \in V$ and $k \in \mathbb{R}$
- Tensors $\mathbf{S}$ and $\mathbf{T}$ are equal iff $\mathbf{S} \cdot \boldsymbol{v} = \mathbf{T} \cdot \boldsymbol{u}$ for any $\boldsymbol{u}, \boldsymbol{v} \in V$

Second Order Tensor Example

- Operator which transforms every vector into its mirror-image with respect to a given plane

$$\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} -1 \\ 1 \end{bmatrix} = \mathbf{T}_{\hat{\boldsymbol{n}}} \cdot \boldsymbol{u} = \mathbf{T}_{\hat{\boldsymbol{n}}}(\boldsymbol{u}) = \boldsymbol{v} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

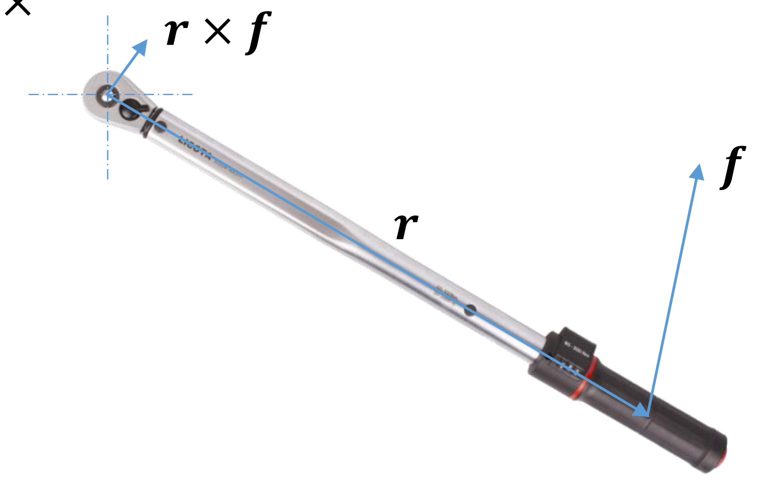


- Note that the situation will be the same in 3D

Second Order Tensor Example

- Operator which transforms force vector $\boldsymbol{f}$ into the moment/torque vector $\boldsymbol{r} \times \boldsymbol{f}$

$$\underbrace{\begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix}}_{\boldsymbol{r}\times} \cdot \boldsymbol{f} = \mathbf{T}_r \cdot \boldsymbol{f} = \mathbf{T}_r(\boldsymbol{f}) = \boldsymbol{r} \times \boldsymbol{f}$$



Tensor Product (Dyad)

- The tensor product of two vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ is written as $\boldsymbol{u} \otimes \boldsymbol{v}$
  - Dot product $\boldsymbol{u} \cdot \boldsymbol{v}$
  - Cross product $\boldsymbol{u} \times \boldsymbol{v}$
  - Direct sum $\boldsymbol{u} \oplus \boldsymbol{v}$
  - Tensor (outer) product $\boldsymbol{u} \otimes \boldsymbol{v}$

„making new vectors from old“

$$\boldsymbol{u} \oplus \boldsymbol{v} = \begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{bmatrix}$$

$$\boldsymbol{u} \otimes \boldsymbol{v} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} u_1 v_1 \\ u_1 v_2 \\ u_2 v_1 \\ u_2 v_2 \\ u_3 v_1 \\ u_3 v_2 \end{bmatrix} \overset{\text{reshape}}{\Leftrightarrow} \begin{bmatrix} u_1 v_1 & u_1 v_2 \\ u_2 v_1 & u_2 v_2 \\ u_3 v_1 & u_3 v_2 \end{bmatrix} = \boldsymbol{u}\boldsymbol{v}^{\mathrm{T}}$$

1st order tensor with dimension 3

1st order tensor with dimension 2

2nd order tensor with dimensions (3, 2)

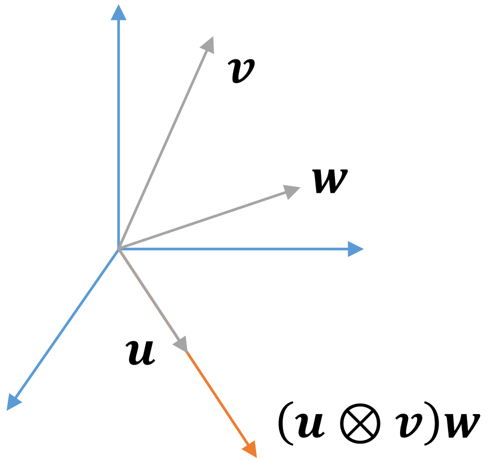
$\mathbb{R}^m \otimes \mathbb{R}^n$ is isomorphism to $\mathbb{R}^{mn}$

Tensor Product (Dyad)

- Tensor product of two vectors (dyad transformation) can be defined as follows

$$(\boldsymbol{u} \otimes \boldsymbol{v})\boldsymbol{w} = \boldsymbol{u}(\boldsymbol{v} \cdot \boldsymbol{w})$$

- It transforms a vector $\boldsymbol{w}$ into a new vector with the direction of $\boldsymbol{u}$ and length of $\|\boldsymbol{u}\|(\boldsymbol{v} \cdot \boldsymbol{w})$



Let us have two vectors $u := \begin{bmatrix} 1 \\ -2 \\ 3 \end{bmatrix}$ and $v := \begin{bmatrix} 7 \\ 6 \\ -5 \end{bmatrix}$

and if their tensor (outer) product is

$$u \cdot v^{\mathrm{T}} = \begin{bmatrix} 7 & 6 & -5 \\ -14 & -12 & 10 \\ 21 & 18 & -15 \end{bmatrix}$$

then for an arbitrary vector w, e.g. $w := \begin{bmatrix} -1 \\ 4 \\ 3 \end{bmatrix}$

the following equality must hold

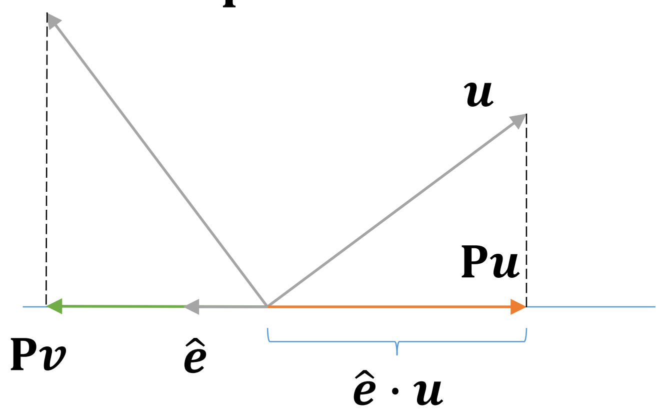
$$\left(u \cdot v^{\mathrm{T}}\right) \cdot w = \begin{bmatrix} 2 \\ -4 \\ 6 \end{bmatrix} \quad = \quad u \cdot (v \cdot w) = \begin{bmatrix} 2 \\ -4 \\ 6 \end{bmatrix}$$

- Note that the tensor product is not commutative

Projection Tensor

- Consider the dyad $\hat{\boldsymbol{e}} \otimes \hat{\boldsymbol{e}}$ then from the definition we get

$$\underbrace{(\hat{\boldsymbol{e}} \otimes \hat{\boldsymbol{e}})}_{\mathbf{P}}\boldsymbol{u} = \hat{\boldsymbol{e}}(\hat{\boldsymbol{e}} \cdot \boldsymbol{u})$$



- $\hat{\boldsymbol{e}} \otimes \hat{\boldsymbol{e}}$ is called projection tensor

Tensor Data Visualization

- Scalar field, e.g. $s\colon \mathbb{R}^3 \to \mathbb{R}$
- Vector field, e.g. $\boldsymbol{u}\colon \mathbb{R}^3 \to \mathbb{R}^3$
- Tensor field, e.g. $\mathbf{T}\colon \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}$

- With tensor fields, we measure some magnitude at some point and in some direction
- Tensors are independent of the coordinate systems

Tensor Data Visualization

- Tensor attributes are high-dimensional generalization of vectors and matrices
- In other words, tensor data encode some spatial property that varies as a function of position and direction
- A tensor with rank (order) r in a n-dimensional space has r indices and n^r components (r is a number representing simultaneous directions)
  - Scalars ($r = 0$) … n^0 component with no index (value)
  - Vectors ($r = 1$) … n^1 components with only one index (vector[i])
  - Matrices ($r = 2$) … n^2 components with two indices (matrix[i][j])
  - Tensors ($r \geq 3$)

Only where $\frac{\partial f}{\partial x}(x) = 0$, otherwise $k = \frac{y''}{(1+y'^2)^{3/2}}$

$x_0: f(x) = 0, k(x) = \frac{\partial^2 f}{\partial x^2}(x) = 0$

$x_1: f(x) = -x^2, k(0) = \frac{\partial^2 f}{\partial x^2}(0) = -2$

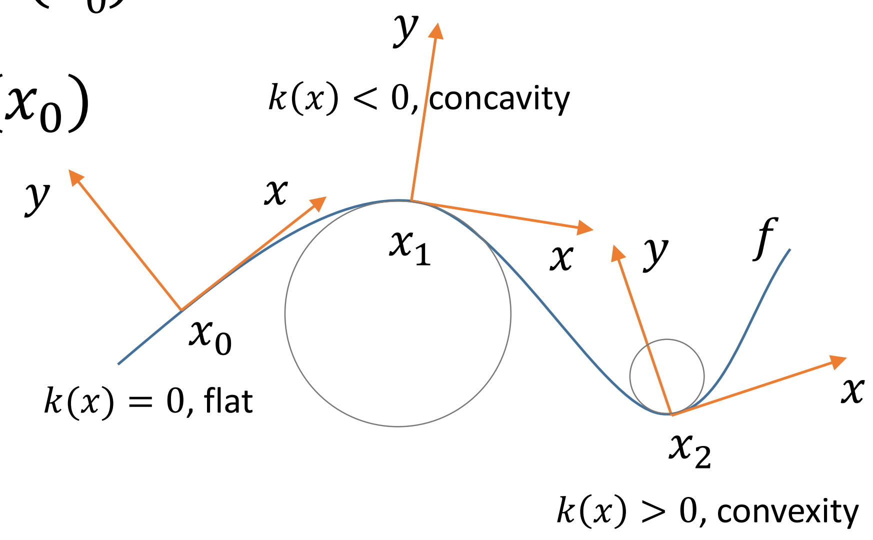
$x_2: f(x) = 2x^2, k(0) = \frac{\partial^2 f}{\partial x^2}(0) = 4$

Tensor Data Visualization

- Example: curvature of a planar curve
  - In local xy coordinate system, a curve can be described as $y = f(x)$ (explicit curve) in the neighborhood of a point x_0 where $f(x_0) = 0$
  - Signed curvature is then defined as $k(x) = \frac{\partial^2 f}{\partial x^2}(x_0)$

    a single number (pointing to $k(x)$)

  - Alternative definitions of curvature
    - $k(x) = 1/\text{radius of circle tangent to } f$ at x_0
    - How quickly normal $\boldsymbol{n}$ changes around x_0



Curvature of Plane Curves

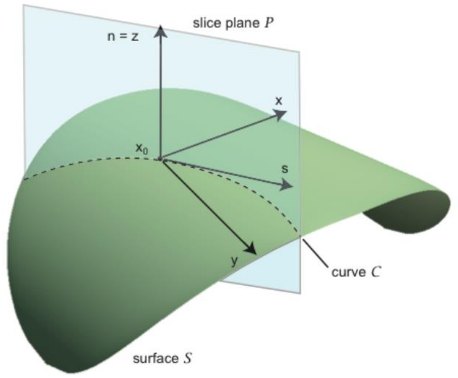
- Let $c(t) = (x(t), y(t))$ be a proper ($\mathrm{d}c/\mathrm{d}t$ is defined, differentiable and nowhere equal to the zero vector) parametric representation of a plane curve then the signed curvature is

$$k = \frac{x'y'' - y'x''}{\left(x'^2 + y'^2\right)^{3/2}}$$

- Example: $x(t) = r\cos(t) + \ x_0; \mathrm{y}(t) = r\sin(t)\ +\ x_0$

$$k = \frac{r\ \sin(t)\ r\ \sin(t) + r\ \cos(t)\ r\ \cos(t)}{(r^2\ \sin^2(t) + r^2\ \cos^2(t) + \ )^{3/2}} = \cdots = \frac{r^2}{r^3} = \frac{1}{r}$$

Tensor Data Visualization



- Example: curvature of a surface
  - In local xyz coordinate system, a surface can be described as $\mathrm{z} = f(x, y)$ (explicit surface) in the neighborhood of a point x_0 where $f(x_0) = 0$
  - Definition of a curvature is analogous with planar case but in which direction to look for changes? This implies that the curvature of a surface at some point cannot be descibed with a single number
  - We have to compute $C(x_0) = \frac{\partial^2 f}{\partial \boldsymbol{s}^2}(x_0) = \boldsymbol{s}^T H \boldsymbol{s}$ where H is called the Hessian of f

$$H = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix}$$

The problem is that we need to construct local coordinate systems at every point on surface and it is not obvious how to do that

and $\boldsymbol{s}$ is the direction in which we look for the surface (normal) curvature

Tensor Data Visualization

- Example: curvature of a surface
  - Solution is to let the surface be describe by an implicit function $f(x, y, z) = 0$
  - Then we can express the curvature of the surface f as

$$C(x_0, \boldsymbol{s}) = \frac{\partial^2 f}{\partial \boldsymbol{s}^2}(x_0) = \frac{\boldsymbol{s}^T H \boldsymbol{s}}{\|\nabla f(x_0)\|},$$

  where H is the 3×3 Hessian matrix in global coordinate system

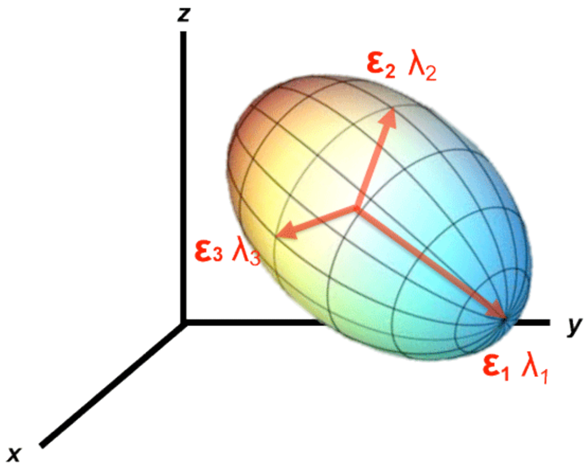
$$H = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial x \partial z} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} & \frac{\partial^2 f}{\partial y \partial z} \\ \frac{\partial^2 f}{\partial z \partial x} & \frac{\partial^2 f}{\partial z \partial y} & \frac{\partial^2 f}{\partial z^2} \end{bmatrix}$$

A curvature tensor of the given surface f is fully described by 3×3 matrix of 2nd order derivatives

Recall that $\mathbf{T}$: $\mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}$

Tensor Data Visualization

- Tensor fields are common quantity in engineering and physical sciences:
  - Stress, strain, diffusion, velocity gradients, etc.
- Mostly second-order tensors – interpreted as a linear transformation between vectors (represented in 3D by 3x3 matrices):
  - Stress to strain, force to deformation
- Special case – symmetric second-order tensors:
  - Can be viewed as anisotropic ellipsoids (eigenvectors and eigenvalues are principal axes of the diffusion ellipsoids)



Tensor Data Visualization

- Example: diffusion tensor
  - Consider an anisotropic material
  - We have to compute diffusivity at a point **x** in a direction **s**

$$D(\mathbf{x},\mathbf{s}) = \frac{\partial^2 f}{\partial \mathbf{s}^2}(\mathbf{x})$$

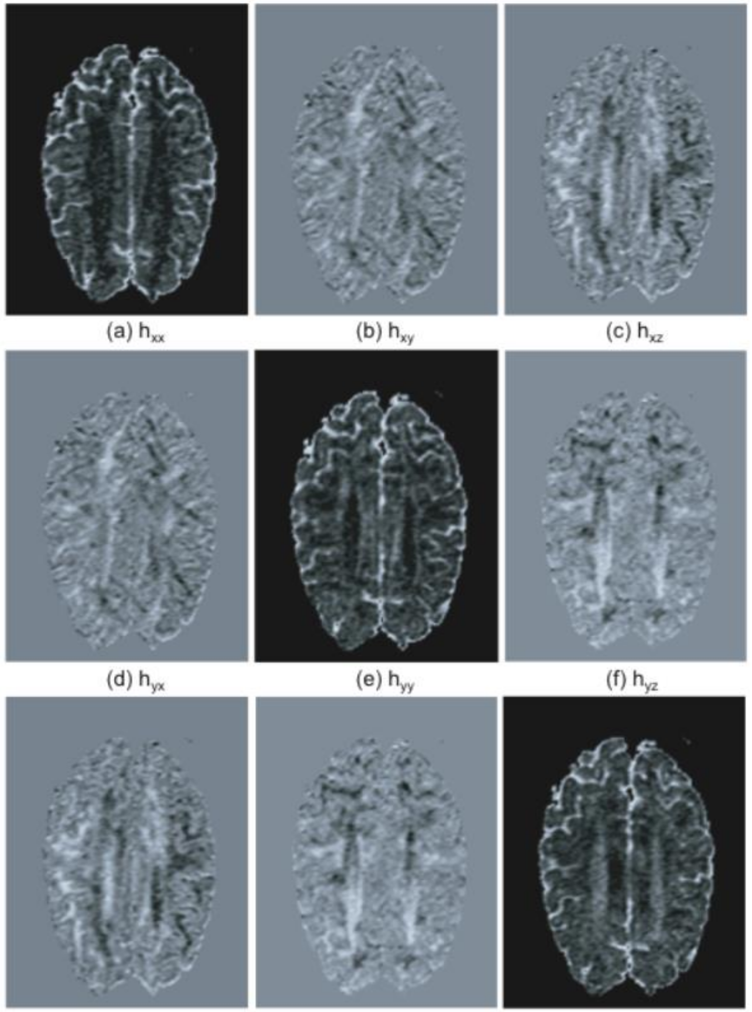
f … speed of water motion in tissue

  - Application: Diffusion of water in the human brain tissue
    - Strong along neural fibers
    - Weak across fibers

Tensor Data Visualization

- Example: diffusion tensor
  - Compute Hessian in R^3
  - Select some slice of interest
  - Visualize all components of H using color mapping

  - We get 9 images, some are same due to the symmetry
  - But we do not really care about diffusion along x, y, z axes



Tensor Data Visualization

- Principal component analysis

α … angle of ***s*** with local coordinate axis $\mathbf{x}_0$

$$\boldsymbol{s}^{\mathrm{T}} = (\cos(\alpha), \sin(\alpha))$$

$$C(\mathbf{x},\mathbf{s}) = \frac{\partial^2 f}{\partial \mathbf{s}^2}(\mathbf{x}) = \mathbf{s}^T \mathbf{H}\mathbf{s} = h_{11}\cos^2\alpha + (h_{12}+h_{21})\sin\alpha\cos\alpha + h_{22}\cos^2\alpha$$

We are looking for extremal curvature

$$\frac{\partial C}{\partial \alpha}(\mathbf{x},\mathbf{s}) = 0 \rightarrow \quad -h_{11}\cos\alpha\sin\alpha - \frac{h_{12}+h_{21}}{2}(\sin^2\alpha - \cos^2\alpha) + h_{22}\sin\alpha\cos\alpha = 0$$

$$\begin{aligned} h_{11}\cos\alpha + h_{12}\sin\alpha &= \lambda\cos\alpha \\ h_{21}\cos\alpha + h_{22}\sin\alpha &= \lambda\sin\alpha. \end{aligned}$$

- This is equivalent to a system of equations in matrix form

$$H\mathbf{s} = \lambda\mathbf{s}$$

Note that the matrix H only stretches (eigen) vector ***s***

$$(H - \lambda I)\mathbf{s} = 0$$

From linear algebra, this is equivalent to $\det(H - \lambda I) = 0$

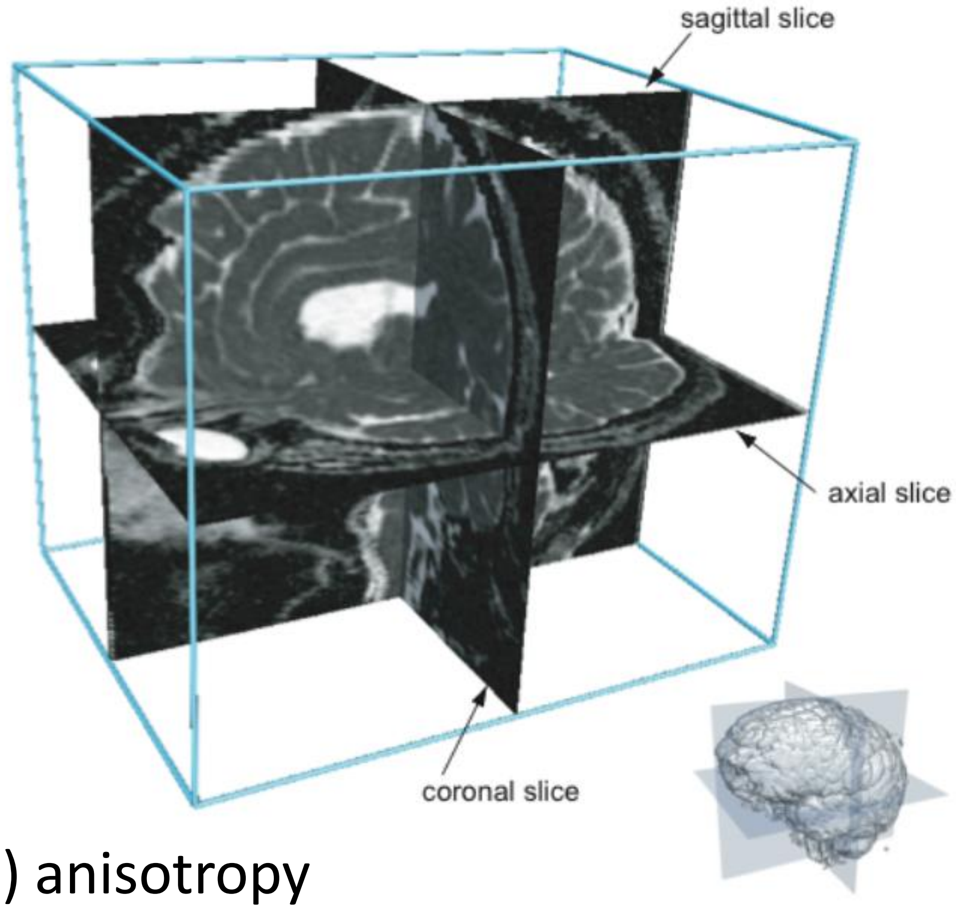
Tensor Data Visualization

- Principal component analysis
- Mean diffusivity

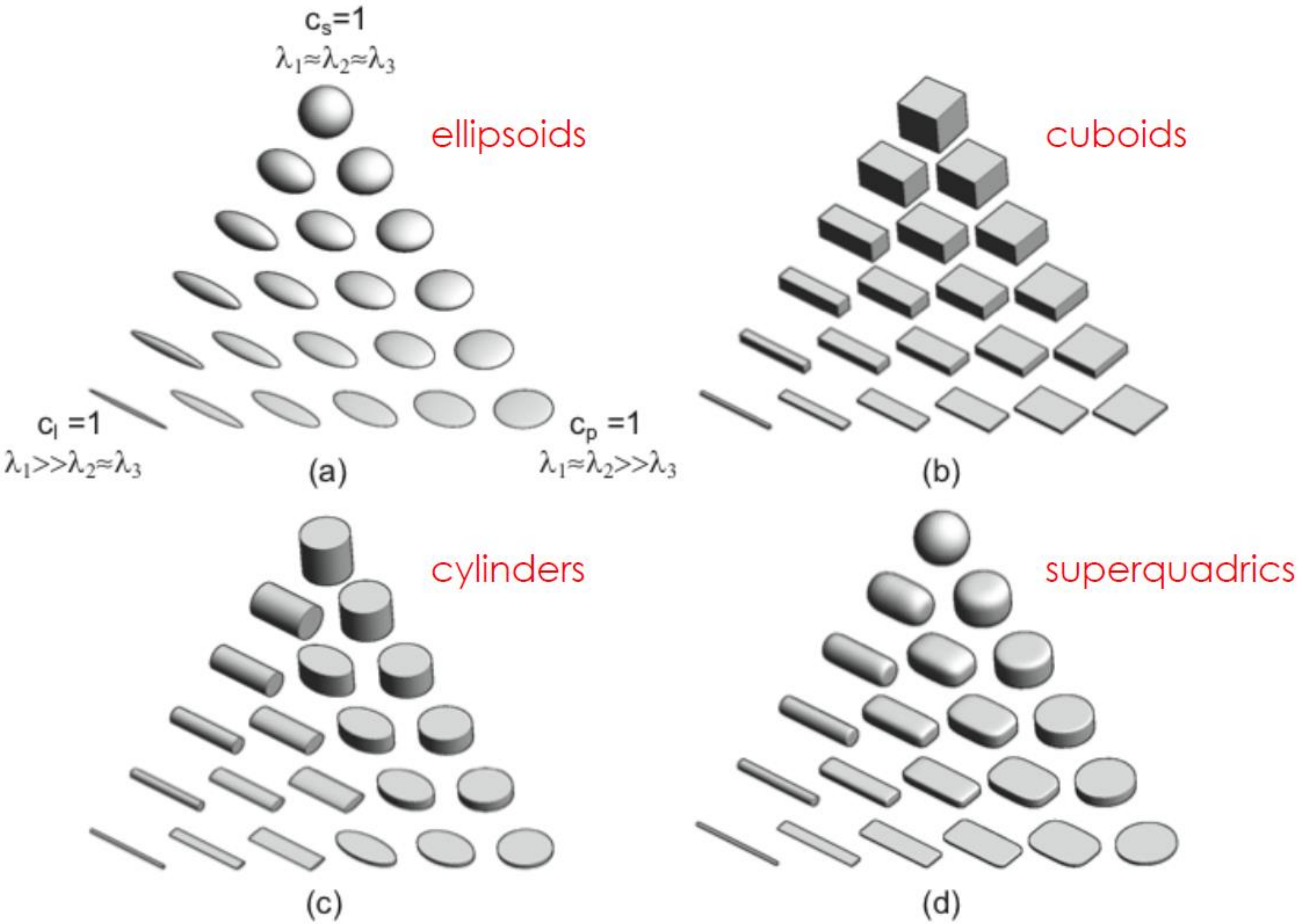
$$\mu = \frac{1}{3}(\lambda_1 + \lambda_2 + \lambda_3)$$

- White = strong mean diff.
- Black = weak mean diff.

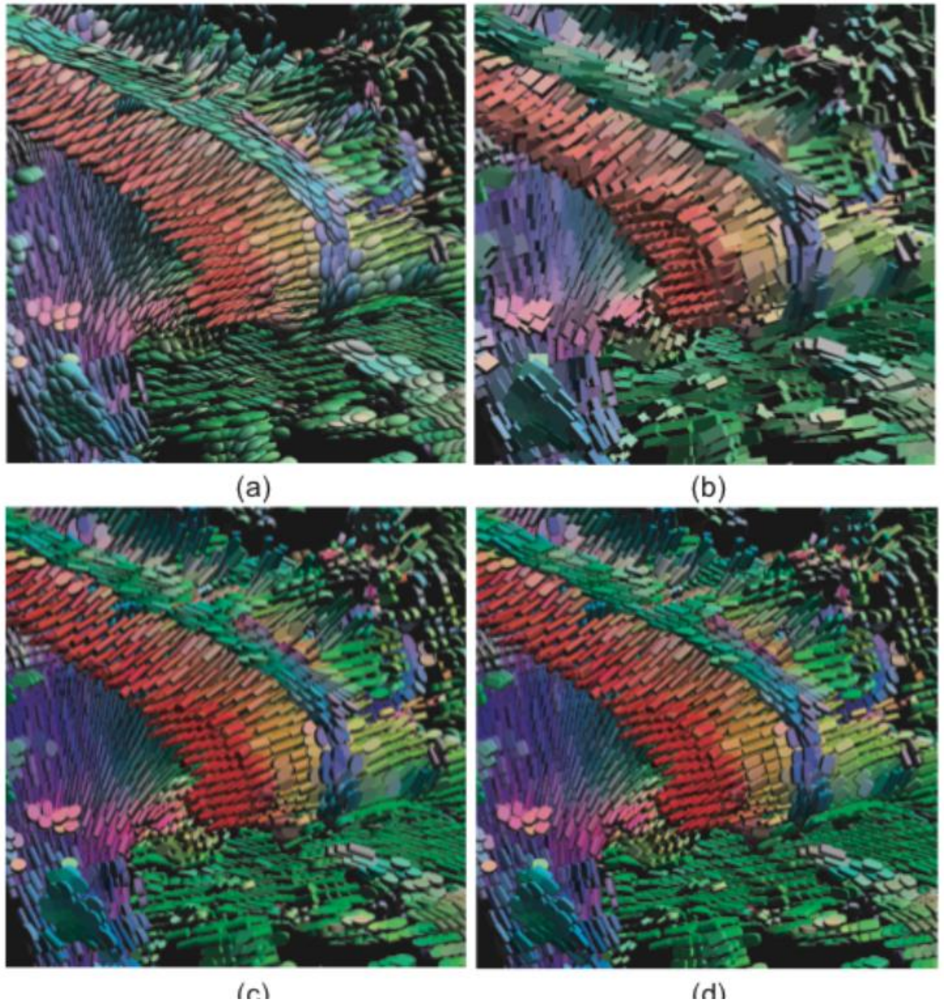
- Other measures: (fractional/relative) anisotropy



Tensor Data Visualization



Tensor Data Visualization



a) ellipsoids
b) cuboids
c) cylinders
d) superquadrics

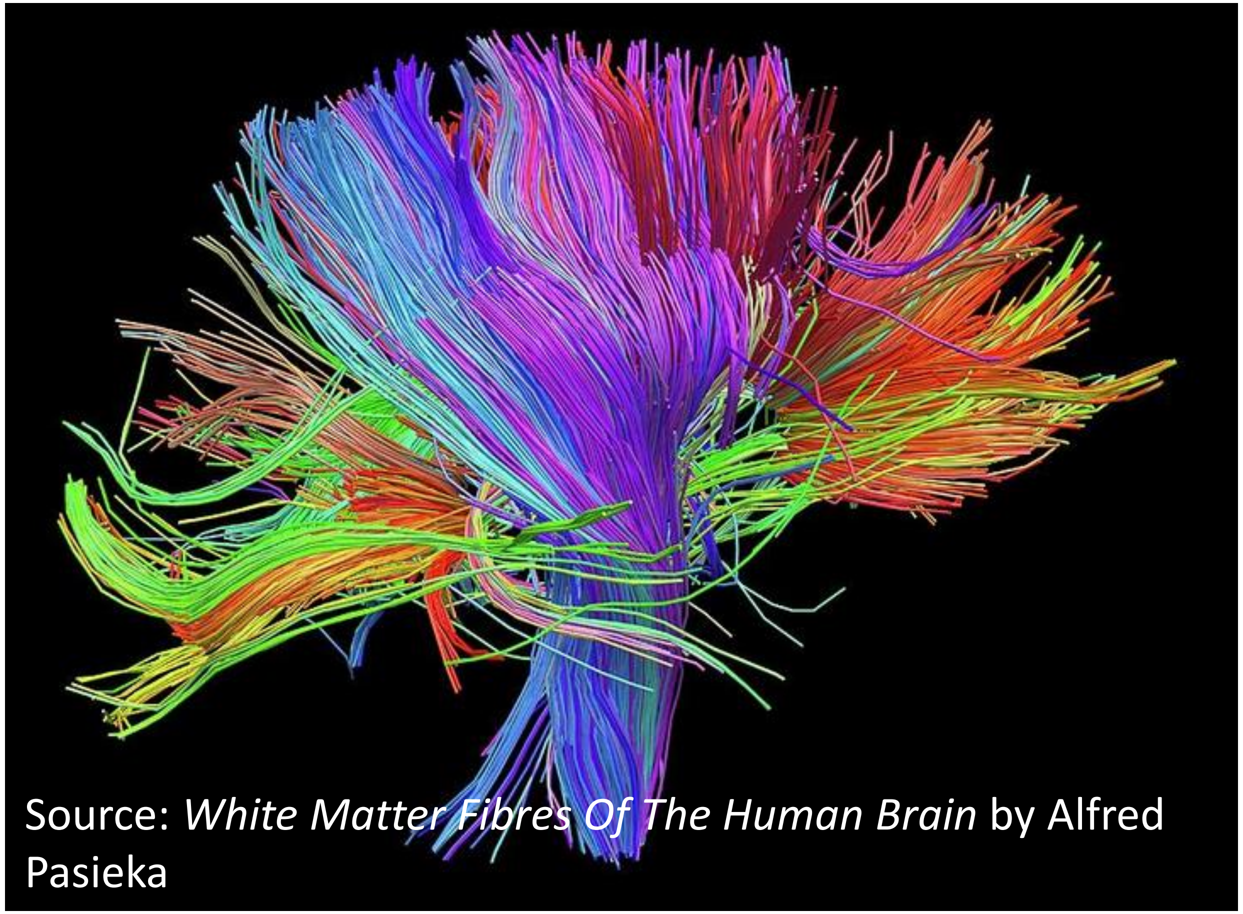
Superquadrics look arguably most ‘natural’

Tensor Data Visualization

- Tensor visualization:
  - Component visualization
  - Anisotropy visualization
  - Major eigenvector visualization
- Fiber tracking
  - Basic fiber tracking
  - Stream tubes
  - Hyperstreamlines

Fiber Tracking

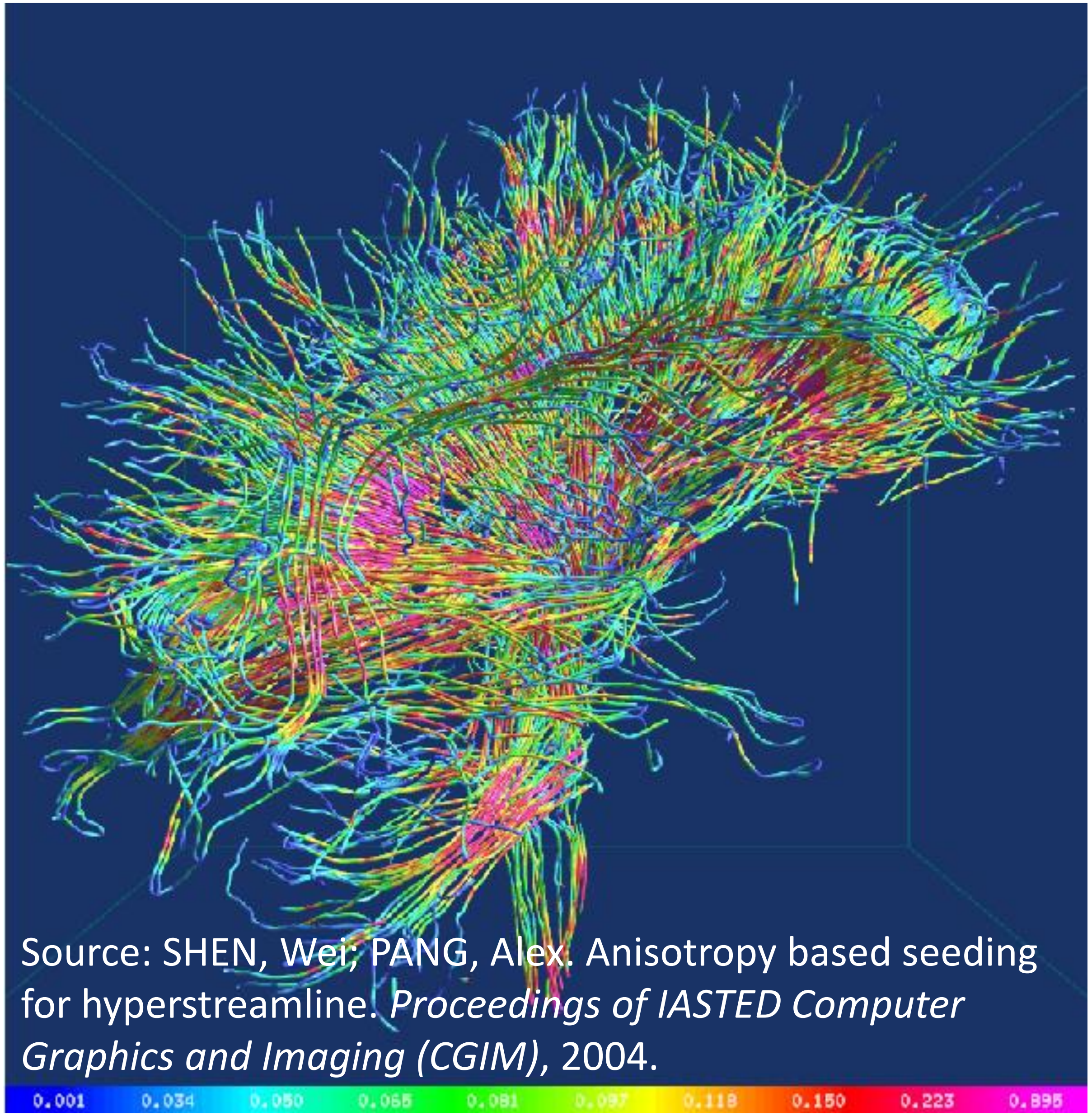
- Similar to streamlines in case of vector fields visualization
- Tracks the direction of the major eigenvectors
- Tubes have circular crossection
- Color indicates the local direction of the hyperstreamlines
- Mostly used for DT-MRI tensors



Source: *White Matter Fibres Of The Human Brain* by Alfred Pasieka

Hyperstreamlines

- Extension of fiber tracking that enables us to visualize direction information from the tensor field beyond the major eigenvector
- Construct stream tubes in the direction of major eigenvector
- Circular cross section is replaced by an elliptical cross section controlled by medium and minor eigenvector



Source: SHEN, Wei; PANG, Alex. Anisotropy based seeding for hyperstreamline. *Proceedings of IASTED Computer Graphics and Imaging (CGIM)*, 2004.