Fiber Orientation Assessment via Symmetric Tensor Decomposition

Yaniv Gur, Fangxiang Jiao, Chris R. Johnson

SCI Institute, University of Utah

Stella Xinghua Zhu

The University of Hong Kong

HARDI reconstruction workshop

ISBI’12, Barcelona

From higher-order tensors to fiber orientations

- An ODF may be represented as an even-order homogenous polynomial (symmetric tensor).

$$F(\mathbf{v}) = \sum_{a=0}^{n} \sum_{b=0}^{n-a} c_{ab} v_1^a v_2^b v_3^{n-a-b}, \quad \mathbf{v} \in S^2$$

From higher-order tensors to fiber orientations

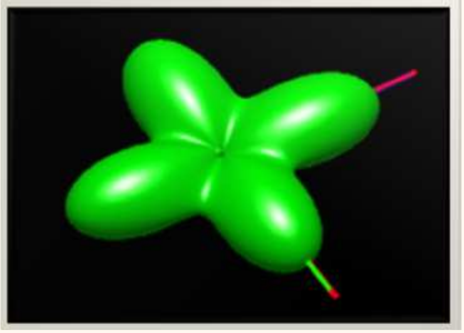
- An ODF may be represented as an even-order homogenous polynomial (symmetric tensor).

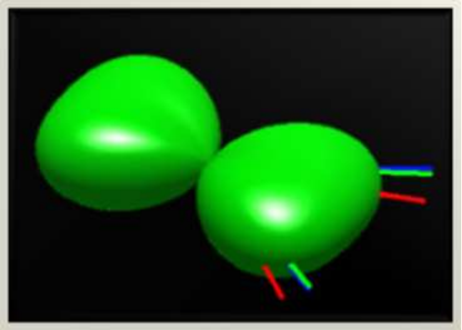
$$F(\mathbf{v}) = \sum_{a=0}^{n} \sum_{b=0}^{n-a} c_{ab} v_1^a v_2^b v_3^{n-a-b}, \quad \mathbf{v} \in S^2$$

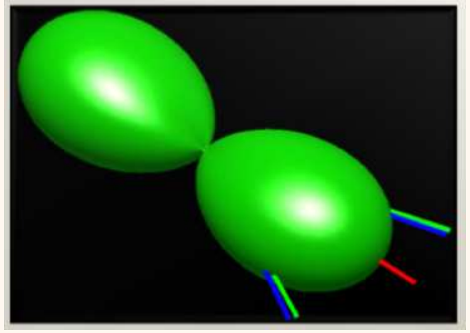
- A low-rank tensor approximation provide accurate estimates of the fiber orientations and enables to reach the full angular resolution of the estimation technique [Schultz et al. IEEE TVCG (2008), Jiao et al., IPMI (2011)]

Higher-order tensor decompositions

$$\mathcal{D} \approx \sum_{r=1}^{k} \lambda_r (\underbrace{\mathbf{v}_r \otimes \mathbf{v}_r \otimes \cdots \otimes \mathbf{v}_r}_{\text{n times}}), \quad ||\mathbf{v}_r|| = 1.$$







The accuracy of the orientations estimation depends on the ODF quality of reconstruction.

The accuracy of the orientations estimation depends on the ODF quality of reconstruction.

Can we use a symmetric tensor decomposition to estimate the orientations directly from the DWI data ?

Higher-order homogenous polynomials

- Homogenous polynomials may be decomposed as a sum of powers of linear-forms [Common et al., 2010].

$$F(x_1, x_2, \ldots, x_l) = \sum_{i=1}^{r} \lambda_i (\boldsymbol{\alpha}_i \cdot \mathbf{x})^d$$

- We approximate a fODF by powers of linear-forms that represent single fibers

$$F(\mathbf{x}) \approx \sum_{i=1}^{k} (\boldsymbol{\alpha}_i \cdot \mathbf{x})^d, \quad \boldsymbol{\alpha}_i \in \mathbb{R}^3,\ \mathbf{x} \in S^2,\ k < r$$

Spherical Deconvolution

- To estimate the fiber parameters directly from the DWI data we convolve the fODF approximation

$$\min_{\boldsymbol{\alpha}_j} \frac{1}{2} \sum_{i=1}^{n} \left\| S(\mathbf{g}_i, b) - S_0 \int_{S^2} \sum_{j=1}^{k} (\boldsymbol{\alpha}_j \cdot \mathbf{v})^d K(\mathbf{g}_i, \mathbf{v}) d\mathbf{v} \right\|^2$$

- Single fiber response kernel

$$K(\mathbf{g}_i, \mathbf{v}, \delta) = e^{-\delta(\mathbf{g}_i^T \cdot \mathbf{v})^2}$$

Spherical Deconvolution

- To estimate the fiber parameters directly from the DWI data we convolve the fODF approximation

$$\min_{\boldsymbol{\alpha}_j} \frac{1}{2} \sum_{i=1}^{n} \left\| S(\mathbf{g}_i, b) - S_0 \int_{S^2} \sum_{j=1}^{k} (\boldsymbol{\alpha}_j \cdot \mathbf{v})^d K(\mathbf{g}_i, \mathbf{v}) d\mathbf{v} \right\|^2$$

- Single fiber response kernel

$$K(\mathbf{g}_i, \mathbf{v}, \delta) = e^{-\delta(\mathbf{g}_i^T \cdot \mathbf{v})^2}$$

- The problem is solved via an alternating iterative Levenberg-Marquardt scheme.

Spherical Deconvolution

- To estimate the fiber parameters directly from the DWI data we convolve the fODF approximation

$$\min_{\boldsymbol{\alpha}_j} \frac{1}{2} \sum_{i=1}^{n} \left\| S(\mathbf{g}_i, b) - S_0 \int_{S^2} \sum_{j=1}^{k} (\boldsymbol{\alpha}_j \cdot \mathbf{v})^d K(\mathbf{g}_i, \mathbf{v}) d\mathbf{v} \right\|^2$$

- Single fiber response kernel

$$K(\mathbf{g}_i, \mathbf{v}, \delta) = e^{-\delta(\mathbf{g}_i^T \cdot \mathbf{v})^2}$$

- The problem is solved via an alternating iterative Levenberg-Marquardt scheme.
- $\mathbf{u}_j = \frac{\tilde{\boldsymbol{\alpha}}_j}{\|\tilde{\boldsymbol{\alpha}}_j\|}$
- $w_j = \frac{\|\tilde{\boldsymbol{\alpha}}_j\|^d}{\sum_{j=1}^{k} \|\tilde{\boldsymbol{\alpha}}_j\|^d}$

Model selection

- Set $k = 1$ and estimate $\tilde{\boldsymbol{\alpha}}_1^1$.

Model selection

- Set $k = 1$ and estimate $\tilde{\boldsymbol{\alpha}}_1^1$.
- Set $k = 2$, use $\tilde{\boldsymbol{\alpha}}_1^1$ for initialization and estimate $\tilde{\boldsymbol{\alpha}}_i^2$, $i = 1, 2$.

Model selection

- Set $k = 1$ and estimate $\tilde{\boldsymbol{\alpha}}_1^1$.
- Set $k = 2$, use $\tilde{\boldsymbol{\alpha}}_1^1$ for initialization and estimate $\tilde{\boldsymbol{\alpha}}_i^2$, $i = 1, 2$.
- If $w_2 > t_1$ set $k = 3$, use the previous estimates for initialization and estimate $\tilde{\boldsymbol{\alpha}}_i^3$, $i = 1, 2, 3$, otherwise accept $\tilde{\boldsymbol{\alpha}}_1^1$.

Model selection

- Set $k = 1$ and estimate $\tilde{\boldsymbol{\alpha}}_1^1$.
- Set $k = 2$, use $\tilde{\boldsymbol{\alpha}}_1^1$ for initialization and estimate $\tilde{\boldsymbol{\alpha}}_i^2$, $i = 1, 2$.
- If $w_2 > t_1$ set $k = 3$, use the previous estimates for initialization and estimate $\tilde{\boldsymbol{\alpha}}_i^3$, $i = 1, 2, 3$, otherwise accept $\tilde{\boldsymbol{\alpha}}_1^1$.
- If $w_3 > t_2$ accept $\tilde{\boldsymbol{\alpha}}_i^3$, $i = 1, 2, 3$, otherwise accept $\tilde{\boldsymbol{\alpha}}_i^2$, $i = 1, 2$.

Model selection

- Set $k = 1$ and estimate $\tilde{\boldsymbol{\alpha}}_1^1$.
- Set $k = 2$, use $\tilde{\boldsymbol{\alpha}}_1^1$ for initialization and estimate $\tilde{\boldsymbol{\alpha}}_i^2,\ i = 1, 2$.
- If $w_2 > t_1$ set $k = 3$, use the previous estimates for initialization and estimate $\tilde{\boldsymbol{\alpha}}_i^3,\ i = 1, 2, 3$, otherwise accept $\tilde{\boldsymbol{\alpha}}_1^1$.
- If $w_3 > t_2$ accept $\tilde{\boldsymbol{\alpha}}_i^3,\ i = 1, 2, 3$, otherwise accept $\tilde{\boldsymbol{\alpha}}_i^2,\ i = 1, 2$.
- The thresholds t_1 and t_2 are adjusted to the data (here we used the training data).

Model selection

- Set $k = 1$ and estimate $\tilde{\boldsymbol{\alpha}}_1^1$.
- Set $k = 2$, use $\tilde{\boldsymbol{\alpha}}_1^1$ for initialization and estimate $\tilde{\boldsymbol{\alpha}}_i^2,\ i = 1, 2$.
- If $w_2 > t_1$ set $k = 3$, use the previous estimates for initialization and estimate $\tilde{\boldsymbol{\alpha}}_i^3,\ i = 1, 2, 3$, otherwise accept $\tilde{\boldsymbol{\alpha}}_1^1$.
- If $w_3 > t_2$ accept $\tilde{\boldsymbol{\alpha}}_i^3,\ i = 1, 2, 3$, otherwise accept $\tilde{\boldsymbol{\alpha}}_i^2,\ i = 1, 2$.
- The thresholds t_1 and t_2 are adjusted to the data (here we used the training data).
- Sampling scheme: b=3000, 64 directions

Thank you !