■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks:... In the following we use quaternions to re-write the function to be optimized, i.e., $\sum_{i=1}^{n} \langle \mathbb{x}_i, R\mathbb{y}_i \rangle$. In quaternion notation, we represent a point $\mathbb{x}_i$ as $x_i = 0 + i x_1^i + j x_2^i + k x_3^i$ and a point $\mathbb{y}_i$ as $y_i = 0 + i y_1^i + j y_2^i + k y_3^i$, where $\mathbb{x}_i = (x_1^i, x_2^i, x_3^i)^T$ and $\mathbb{y}_i = (y_1^i, y_2^i, y_3^i)^T$, $i = 1 \ldots n$. We write the quaternion defining a rotation in the compact notation $q = q_{rot} = (c + \langle \mathbb{i}, \mathbb{d} \rangle s) = (c\frac{\alpha}{2} + \langle \mathbb{i}, \mathbb{d} \rangle s\frac{\alpha}{2}) =$ $= (c\frac{\alpha}{2} + (i d_x + j d_y + k d_z) s\frac{\alpha}{2})$, where α is the unknown rotation angle and $\mathbb{d} = (d_x, d_y, d_z)^T$ the unknown unit axis direction vector. Further, we must use $\overline{q_{rot}} = (c - \langle \mathbb{i}, \mathbb{d} \rangle s)$ to solve the problem.

Since $\sum_{i=1}^{n} \langle \mathbb{x}_i, R\mathbb{y}_i \rangle = \sum_{i=1}^{n} \langle R\mathbb{y}_i, \mathbb{x}_i \rangle$, we use the representation $\sum_{i=1}^{n} \langle R\mathbb{y}_i, \mathbb{x}_i \rangle$ — which offers some advantages — in the following. The image of $\mathbb{y}_i$ is called $\hat{\mathbb{y}}_i = R\mathbb{y}_i = R \cdot (y_1^i, y_2^i, y_3^i)^T$, which has the quaternion representation $\hat{y}_i = 0 + i\hat{y}_1^i + j\hat{y}_2^i + k\hat{y}_3^i =$ $= \hat{y}_0^i + i y_1^i + j y_2^i + k y_3^i$, $i = 1 \ldots n$. Using these conventions, we obtain the following "standard" inner product:

$$\langle R\mathbb{y}_i, \mathbb{x}_i \rangle = \langle \hat{\mathbb{y}}_i, \mathbb{x}_i \rangle = [\hat{y}_1^i, \hat{y}_2^i, \hat{y}_3^i] \begin{bmatrix} x_1^i \\ x_2^i \\ x_3^i \end{bmatrix} = \hat{y}_1^i x_1^i + \hat{y}_2^i x_2^i + \hat{y}_3^i x_3^i$$

$$= [0, \hat{y}_1^i, \hat{y}_2^i, \hat{y}_3^i] \begin{bmatrix} 0 \\ x_1^i \\ x_2^i \\ x_3^i \end{bmatrix} = \hat{y}_0^i x_0^i + \hat{y}_1^i x_1^i + \hat{y}_2^i x_2^i + y_3^i x_3^i,$$

$i = 1 \ldots n$.

...

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• **Laplacian eigenfunctions and neural networks:...** In order to simplify the notation for the unknown axis direction vector $\mathbf{d} = (d_x, d_y, d_z)^T$, $\|\mathbf{d}\| = 1$, this vector is called $\mathbf{u} = (u, v, w)^T$, $\|\mathbf{u}\| = 1$, in the subsequent derivations. We can now write $\langle R\mathbf{y}_i, \mathbf{x}_i \rangle$ in the equivalent quaternion representation:

• $$R\mathbf{y}_i \;\hat{=}\; q_{rot} \cdot y_i \cdot \overline{q_{rot}}$$

$$= \left(c_{\frac{\alpha}{2}} + s_{\frac{\alpha}{2}}(iu + jv + kw)\right) \cdot \left(0 + iy_1^i + jy_2^i + ky_3^i\right) \cdot \left(c_{\frac{\alpha}{2}} - s_{\frac{\alpha}{2}}(iu + jv + kw)\right)$$

$$\left(= \left(\hat{y}_0^i + i\hat{y}_1^i + j\hat{y}_2^i + k\hat{y}_3^i\right)\right)$$

$$= \left[c_{\frac{\alpha}{2}}, i s_{\frac{\alpha}{2}} u, j s_{\frac{\alpha}{2}} v, k s_{\frac{\alpha}{2}} w\right] \begin{bmatrix} 0 \\ i y_1^i \\ j y_2^i \\ k y_3^i \end{bmatrix} \left[c_{\frac{\alpha}{2}}, -i s_{\frac{\alpha}{2}} u, -j s_{\frac{\alpha}{2}} v, -k s_{\frac{\alpha}{2}} w\right] =$$

$$= \left[c_{\frac{\alpha}{2}}, i s_{\frac{\alpha}{2}} u, j s_{\frac{\alpha}{2}} v, k s_{\frac{\alpha}{2}} w\right] \cdot$$

$$\cdot \begin{bmatrix} 0 & 0 & 0 & 0 \\ i y_1^i c_{\frac{\alpha}{2}} & -i^2 y_1^i s_{\frac{\alpha}{2}} u & -ij y_1^i s_{\frac{\alpha}{2}} v & -ik y_1^i s_{\frac{\alpha}{2}} w \\ j y_2^i c_{\frac{\alpha}{2}} & -ji y_2^i s_{\frac{\alpha}{2}} u & -j^2 y_2^i s_{\frac{\alpha}{2}} v & -jk y_2^i s_{\frac{\alpha}{2}} w \\ k y_3^i c_{\frac{\alpha}{2}} & -ki y_3^i s_{\frac{\alpha}{2}} u & -kj y_3^i s_{\frac{\alpha}{2}} v & -k^2 y_3^i s_{\frac{\alpha}{2}} w \end{bmatrix} =$$

$$= [\,0 + i^2 y_1^i s_{\frac{\alpha}{2}} c_{\frac{\alpha}{2}} u + j^2 y_2^i s_{\frac{\alpha}{2}} c_{\frac{\alpha}{2}} v + k^2 y_3^i s_{\frac{\alpha}{2}} c_{\frac{\alpha}{2}} w,$$
$$0 - i^3 y_1^i s^2_{\frac{\alpha}{2}} u^2 - j^2 i y_2^i s^2_{\frac{\alpha}{2}} vu - k^2 i y_3^i s^2_{\frac{\alpha}{2}} wu,$$
$$0 - i^2 j y_1^i s^2_{\frac{\alpha}{2}} uv - j^3 y_2^i s^2_{\frac{\alpha}{2}} v^2 - k^2 j y_3^i s^2_{\frac{\alpha}{2}} wv,$$
$$0 - i^2 k y_1^i s^2_{\frac{\alpha}{2}} uw - j^2 k y_2^i s^2_{\frac{\alpha}{2}} vw - k^3 y_3^i s^2_{\frac{\alpha}{2}} w^2\,] =$$

$$= [\,0 - y_1^i s_{\frac{\alpha}{2}} c_{\frac{\alpha}{2}} u - y_2^i s_{\frac{\alpha}{2}} c_{\frac{\alpha}{2}} v - y_3^i s_{\frac{\alpha}{2}} c_{\frac{\alpha}{2}} w,$$
$$0 + i y_1^i s^2_{\frac{\alpha}{2}} u^2 + i y_2^i s^2_{\frac{\alpha}{2}} vu + i y_3^i s^2_{\frac{\alpha}{2}} wu,$$
$$0 + j y_1^i s^2_{\frac{\alpha}{2}} uv + j y_2^i s^2_{\frac{\alpha}{2}} v^2 + j y_3^i s^2_{\frac{\alpha}{2}} wv,$$
$$0 + k y_1^i s^2_{\frac{\alpha}{2}} uw + k y_2^i s^2_{\frac{\alpha}{2}} vw + k y_3^i s^2_{\frac{\alpha}{2}} w^2\,] =$$

•••

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• **Laplacian eigenfunctions and neural networks:...** $= \hat{y}_0^i + i\hat{y}_1^i + j\hat{y}_2^i + k\hat{y}_3^i =$

$$= -s\tfrac{\alpha}{2}c\tfrac{\alpha}{2}\left(y_1^i u + y_2^i v + y_3^i w\right)$$
$$+\, i\left(s^2\tfrac{\alpha}{2}\left(y_1^i u^2 + y_2^i vu + y_3^i wu\right)\right)$$
$$+\, j\left(s^2\tfrac{\alpha}{2}\left(y_1^i uv + y_2^i v^2 + y_3^i wv\right)\right)$$
$$+\, k\left(s^2\tfrac{\alpha}{2}\left(y_1^i uw + y_2^i vw + y_3^i w^2\right)\right).$$

• Next, one can compute the inner product $\langle R\mathbf{y}_i, \mathbf{x}_i\rangle$:

$$\langle R\mathbf{y}_i, \mathbf{x}_i\rangle = \Big[-s\tfrac{\alpha}{2}c\tfrac{\alpha}{2}\left(y_1^i u + y_2^i v + y_3^i w\right),\; s^2\tfrac{\alpha}{2}\left(y_1^i u^2 + y_2^i vu + y_3^i wu\right),\; s^2\tfrac{\alpha}{2}\left(y_1^i uv + y_2^i v^2 + y_3^i wv\right),\; s^2\tfrac{\alpha}{2}\left(y_1^i uw + y_2^i vw + y_3^i w^2\right)\Big] \cdot \begin{bmatrix} x_0^i \\ x_1^i \\ x_2^i \\ x_3^i \end{bmatrix} =$$

$$\left(s\tfrac{\alpha}{2}c\tfrac{\alpha}{2} = \tfrac{1}{2}s\alpha,\quad s^2\tfrac{\alpha}{2} = s\tfrac{\alpha}{2}s\tfrac{\alpha}{2} = \tfrac{1}{2} - \tfrac{1}{2}c\alpha \right)$$

$$= -\tfrac{1}{2}s\alpha\left(x_0^i y_1^i u + x_0^i y_2^i v + x_0^i y_3^i w\right)$$
$$+\left(\tfrac{1}{2} - \tfrac{1}{2}c\alpha\right)\Big(x_1^i y_1^i u^2 + x_1^i y_2^i vu + x_1^i y_3^i wu$$
$$+ x_2^i y_1^i uv + x_2^i y_2^i v^2 + x_2^i y_3^i wv$$
$$+ x_3^i y_1^i uw + x_3^i y_2^i vw + x_3^i y_3^i w^2\Big).$$

These inner products $(i = 1 \ldots n)$ define the individual summands of the function to be maximized, called $f = f(\alpha, u, v, w) = \sum_{i=1}^{n} \langle R\mathbf{y}_i, \mathbf{x}_i\rangle$.

• **Note.** A necessary condition for the function f to be maximal (locally) is the requirement that its partial derivatives must be zero at a maximum, i.e., $\frac{\partial}{\partial\alpha} f = f_\alpha = 0$, $\frac{\partial}{\partial u} f = f_u = 0$, $\frac{\partial}{\partial v} f = f_v = 0$ and $\frac{\partial}{\partial w} = f_w = 0$. Further, one must keep in mind the geometrical meaning of the variables: The values of α and $\mathbf{u}$ define optimal angle and **axis** direction.

...

OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks:... First, since the unit vector $\mathbf{u} = (u, v, w)^T$, $\|\mathbf{u}\| = 1$, defines the direction of the optimal axis of rotation, both $\mathbf{u}$ and $-\mathbf{u}$ are optimal axis direction vectors (affecting only the orientation of the axis). Second, as a consequence of periodicity, an infinite number of rotation angles must be kept in mind: If $\alpha \in [0, 2\pi]$ is an optimal rotation angle, then $\alpha - 2\pi, \alpha - 4\pi, \alpha - 6\pi, \ldots$ and $\alpha + 2\pi, \alpha + 4\pi, \alpha + 6\pi, \ldots$ will also be optimal. Third, since an optimal axis direction vector $\mathbf{u}$ must satisfy $\|\mathbf{u}\| = 1$, the maximization problem must consider the constraint that $\mathbf{u}$ is a positional vector of a point on the unit sphere. Thus, one can use the method of **LAGRANGE MULTIPLIERS** to maximize $f(\alpha, u, v, w)$.

When computing the partial derivatives of $f(\alpha, u, v, w)$, we can use the rule that holds for the differentiation of a general multivariate function $F = F(x_1, \ldots, x_m) = \sum_{i=1}^{n} F_i(x_1, \ldots, x_m)$: $\frac{\partial}{\partial x_j}\left(\sum_{i=1}^{n} F_i(x_1, \ldots, x_m)\right) = \sum_{i=1}^{n}\left(\frac{\partial}{\partial x_j} F_i(x_1, \ldots, x_m)\right)$.
We apply this rule to compute the four partial derivatives of $f(\alpha, u, v, w) = \sum_{i=1}^{n} f_i(\alpha, u, v, w) = \sum_{i=1}^{n} \langle R\mathbf{y}_i, \mathbf{x}_i \rangle$.
In the following, we provide the needed partial derivatives $\frac{\partial}{\partial \alpha} f_i = \frac{\partial}{\partial \alpha}\langle R\mathbf{y}_i, \mathbf{x}_i \rangle$, $\frac{\partial}{\partial u} f_i = \frac{\partial}{\partial u}\langle R\mathbf{y}_i, \mathbf{x}_i \rangle$, $\frac{\partial}{\partial v} f_i = \frac{\partial}{\partial v}\langle R\mathbf{y}_i, \mathbf{x}_i \rangle$ and $\frac{\partial}{\partial w} f_i = \frac{\partial}{\partial w}\langle R\mathbf{y}_i, \mathbf{x}_i \rangle$.

...

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks: ...

$$\frac{\partial}{\partial \alpha}\langle R y_i, x_i\rangle = -\tfrac{1}{2} c\alpha \left(x_0^i y_1^i u + x_0^i y_2^i v + x_0^i y_3^i w \right)$$
$$+\tfrac{1}{2} s\alpha \left(x_1^i y_1^i u^2 + x_1^i y_2^i vu + x_1^i y_3^i wu + x_2^i y_1^i uv + x_2^i y_2^i v^2 + x_2^i y_3^i wv + x_3^i y_1^i uw + x_3^i y_2^i vw + x_3^i y_3^i w^2 \right).$$

$(= f_\alpha^i)$

$$\frac{\partial}{\partial u}\langle R y_i, x_i\rangle = -\tfrac{1}{2} s\alpha \left(x_0^i y_1^i \right)$$
$$+\left(\tfrac{1}{2}-\tfrac{1}{2}c\alpha\right)\big(2 x_1^i y_1^i u + x_1^i y_2^i v + x_1^i y_3^i w$$
$$+ x_2^i y_1^i v + 0 + 0$$
$$+ x_3^i y_1^i w + 0 + 0 \big).$$

$(= f_v^i)$

$$\frac{\partial}{\partial v}\langle R y_i, x_i\rangle = -\tfrac{1}{2} s\alpha \left(x_0^i y_2^i \right)$$
$$+\left(\tfrac{1}{2}-\tfrac{1}{2}c\alpha\right)\big(0 + x_1^i y_2^i u + 0$$
$$+ x_2^i y_1^i u + 2 x_2^i y_2^i v + x_2^i y_3^i w$$
$$+ 0 + x_3^i y_2^i w + 0 \big).$$

$(= f_v^i)$

$$\frac{\partial}{\partial w}\langle R y_i, x_i\rangle = -\tfrac{1}{2} s\alpha \left(x_0^i y_3^i \right)$$
$$+\left(\tfrac{1}{2}-\tfrac{1}{2}c\alpha\right)\big(0 + 0 + x_1^i y_3^i u$$
$$+ 0 + 0 + x_2^i y_3^i v$$
$$+ x_1^i y_3^i u + x_3^i y_2^i v + 2 x_3^i y_3^i w \big).$$

$(= f_w^i)$

We will call the i^{th} of these four partial derivatives f_α^i, f_u^i, f_v^i and f_w^i. We write f_u^i, f_v^i and f_w^i more compactly:

$$f_u^i = -\tfrac{1}{2} s\alpha \left(x_0^i y_1^i \right) + \left(\tfrac{1}{2}-\tfrac{1}{2}c\alpha\right)\Big(2 x_1^i y_1^i u + \left(x_1^i y_2^i + x_2^i y_1^i\right) v + \left(x_1^i y_3^i + x_3^i y_1^i\right) w \Big).$$

$$f_v^i = -\tfrac{1}{2} s\alpha \left(x_0^i y_2^i \right) + \left(\tfrac{1}{2}-\tfrac{1}{2}c\alpha\right)\Big(\left(x_1^i y_2^i + x_2^i y_1^i\right) u + 2 x_2^i y_2^i v + \left(x_2^i y_3^i + x_3^i y_2^i\right) w \Big).$$

$$f_w^i = -\tfrac{1}{2} s\alpha \left(x_0^i y_3^i \right) + \left(\tfrac{1}{2}-\tfrac{1}{2}c\alpha\right)\Big(\left(x_1^i y_3^i + x_3^i y_1^i\right) u + \left(x_2^i y_3^i + x_3^i y_2^i\right) v + 2 x_3^i y_3^i w \Big).$$

...