1 Szenario

Gegeben sind 2 Vektoren $\vec{v}$ und $\vec{w}$ im selben Koordinatensystem. Als anschauliches Beispiel kann man sich vorstellen man kennt 2 Positionen eines Flugzeugs und sucht nach dem Quaternion, welches Position P1 nach Position P2 rotiert.

2 Idee

Wie hoffentlich bekannt ist, repräsentiert ein Quaternion eine Drehachse $\vec{u}$ und einen Drehwinkel α . Ein Quaternion q kann mit beiden Angaben wie folgt dargestellt werden:

$$q = \cos(\alpha/2) \cdot \vec{u} \sin(\alpha/2)$$

Wenn man also den Winkel α und die Drehachse $\vec{u}$ bestimmen kann, so kann man leicht nach obiger Formel das Quaternion erzeugen.
Der Winkel α ist wie schon angesprochen der Drehwinkel von $\vec{v}$ nach $\vec{w}$, um diesen zu bestimmen berechnet man das Skalarprodukt von $\vec{v}$ und $\vec{w}$:

$$\alpha = \arccos\left(\frac{\vec{v} \cdot \vec{w}}{|\vec{v}| \cdot |\vec{w}|}\right)$$

Um die Drehachse zu bestimmen, hilft folgende Überlegung. 2 Vektoren spannen stets eine Ebene in $\mathbb{R}^3$ auf. Damit die Ebene bei der Rotation nciht verlassen wird, wird ein auf die Ebene senkrechter Vektor benötigt. Diesen erzeugen wir durch das Kreuzprodukt von $\vec{v}$ und $\vec{w}$:

$$\vec{u} = \frac{\vec{v} \times \vec{w}}{|\vec{v} \times \vec{w}|}$$

Damit haben wir α und $\vec{u}$ bestimmt und können unser Quaternion direkt aufstellen.

3 Implementierung

Wir werden wie oben angegeben Verfahren:

```
Vector3D v(1,2,3);
Vector3D w(0,-1,-4);
double alpha;
Vector3D u;

alpha = v.getAngle(w); // liefert winkel in rad
u = v.cross(w).normalize();

Quaternion q(alpha,u);
```

und letztendlich können über das Quaternion die Yaw,-Pitch und Roll Winkel bestimmt werden:

```
YPR Y(q);
```