Geometry in Computer Vision

Spring 2010

Lecture 7A

Representations of 3D rotations

Orthogonal transformations

- From linear algebra we know that for a vector space *V* there is a special set of transformations **A** known as *orthogonal transformations* (or *self-adjoint* transf.)

$(\mathbf{A}\,\mathbf{x}) \cdot \mathbf{y} = \mathbf{x} \cdot (\mathbf{A}\,\mathbf{y})$ for all $\mathbf{x}, \mathbf{y} \in V$

$\mathbf{A}^T\mathbf{A} = \mathbf{A}\,\mathbf{A}^T = \mathbf{I}$

These two definitions are equivalent (why?)

Orthogonal transformations

- For $V=R^3$, the set of orthogonal transformations is denoted O(3)
- O(3) are represented by 3×3 matrices that satisfy $\mathbf{A}^T\mathbf{A} = \mathbf{I}$ (or $\mathbf{A}^T = \mathbf{A}^{-1}$)
- From $\mathbf{A}^T\mathbf{A} = \mathbf{I}$ follows that det $\mathbf{A} = \pm 1$ (why?)
- O(3) consists of two *disconnected parts* in the space of 3×3 matrices:
  - one with det **A** = 1
  - one with det **A** = -1
- O(3) forms a *group* under matrix multiplication

3D Rotations

- The set of O(3) with det **A** = 1 are *3D rotations*
- Also known as the *special orthogonal transformations*
- Denoted SO(3)
- Forms a group under matrix multiplication

- The set of O(3) with det **A** = -1 do not form a group (why?)
- This set includes mirroring operations

Representations

- In many applications we want to determine a rotation:
  - External camera parameters include a rotation
  - **E** is determined by a rotation and a translation
  - Find the rigid transformation between 2 point sets; it includes a rotation
  - Bundle adjustment …
- To solve such problems, we often need to parameterize the set of rotations: SO(3)

3D Rotations

- A 3D rotation **R** is characterized by a
  - normalized vector **n** (2 d.o.f.)
  - rotation angle α (1 d.o.f.)
  - α is well-defined, e.g., using the right-hand-rule
- **R** rotates around the vector **n** with the angle α
- Note: (**n**, α) is equivalent to (-**n**, -α)
- In total: 3 degrees of freedom

Euler angles

- We can decompose any **R** $\in$ SO(3) into a product of 3 rotations around *fixed axes*
- For example:

  $\mathbf{R} = \mathrm{Rot}_z(\alpha_1)\ \mathrm{Rot}_x(\alpha_2)\ \mathrm{Rot}_z(\alpha_3)$

- (α_1, α_2, α_3) are the *Euler angles* of **R**

Euler angles

- (α_1, α_2, α_3) are unique (modulo 2π, π, 2π)
- Non-trivial relation between (α_1, α_2, α_3) and (**n**, α)
- Non-trivial to combine two rotations
- Non-trivial mapping **R** $\rightarrow$ (α_1, α_2, α_3)
- Not very interesting for practical applications

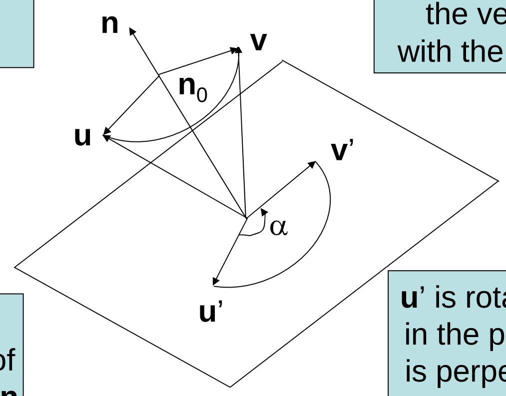
Vector **n** and angle α

- (**n**, α) is a convenient representation
  - With |**n**|=1
  - Explicitly describes the rotation axis and angle
- But
  - Not unique unless we impose restrictions on (**n**, α)
  - Not trivial to combine two rotations
- How do we map **R** $\leftrightarrow$ (**n**, α) ?

The anatomy of a 3D rotation

v = **R u**

R rotates **u** around the vector **n** with the angle α



$\mathbf{n}_0$ is the projection of **v** and **u** on **n**

u' is rotated to **v**' in the plane that is perpendicular to **n**

The anatomy of a 3D rotation

- **v** can be decomposed as $\mathbf{v} = \mathbf{n}_0 + \mathbf{v}'$

  where $\mathbf{n}_0$ is the projection of **u** onto **n**

  $$\mathbf{n}_0 = \mathbf{n}\,\mathbf{n}^T\mathbf{u}$$

  and **v**' is the rotation of **u**' in the plane perpendicular to **n**

The anatomy of a 3D rotation

- Let (**p**, **q**) be an ON-basis for the plane that is perpendicular to **n**
- The coordinates of **u**' in this basis is ($\mathbf{p}^T\mathbf{u}$, $\mathbf{q}^T\mathbf{u}$)
- The coordinates of **v**' in this basis is ($\mathbf{p}^T\mathbf{v}'$, $\mathbf{q}^T\mathbf{v}'$)

and
$$\begin{pmatrix} \mathbf{p}^T\mathbf{v}' \\ \mathbf{q}^T\mathbf{v}' \end{pmatrix} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} \mathbf{p}^T\mathbf{u} \\ \mathbf{q}^T\mathbf{u} \end{pmatrix}$$

The anatomy of a 3D rotation

- From this we get

$$\mathbf{v}' = (\mathbf{p} \quad \mathbf{q}) \begin{pmatrix} \mathbf{p}^T\mathbf{v}' \\ \mathbf{q}^T\mathbf{v}' \end{pmatrix} = (\mathbf{p} \quad \mathbf{q}) \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} \mathbf{p}^T \\ \mathbf{q}^T \end{pmatrix} \mathbf{u}$$

$$\mathbf{v}' = \left(\cos\alpha(\mathbf{p}\mathbf{p}^T + \mathbf{q}\mathbf{q}^T) + \sin\alpha[\mathbf{q}\mathbf{p}^T - \mathbf{p}\mathbf{q}^T]\right)\mathbf{u}$$

$$\mathbf{v} = \left(\mathbf{n}\mathbf{n}^T + \cos\alpha(\mathbf{I} - \mathbf{n}\mathbf{n}^T) + \sin\alpha[\mathbf{n}]_\times\right)\mathbf{u}$$

$$\mathbf{v} = \left(\mathbf{I} + (1-\cos\alpha)\,[\mathbf{n}]_\times^2 + \sin\alpha[\mathbf{n}]_\times\right)\mathbf{u}$$

Rodrigues' formula

- This gives *Rodrigues' formula* for **R**:

$$\mathbf{R} = \mathbf{I} + (1-\cos\alpha)\,[\mathbf{n}]_\times^2 + \sin\alpha[\mathbf{n}]_\times$$

- This gives us a mapping (**n**, α) → **R**
- How do we map **R** → (**n**, α) ?

Rodrigues' formula (II)

From this formula follows directly:

$$\operatorname{tr}\mathbf{R} = \operatorname{tr}\left(\mathbf{I} + (1-\cos\alpha)\,[\mathbf{n}]_\times^2 + \sin\alpha[\mathbf{n}]_\times\right)$$

$$\operatorname{tr}\mathbf{R} = 3 + (1-\cos\alpha)\,(-2) + \sin\alpha\,0$$

and we get

$$\frac{\operatorname{tr}\mathbf{R} - 1}{2} = \cos\alpha$$

Rodrigues' formula (II)

Rodrigues' formula also gives

$$\frac{\mathbf{R} - \mathbf{R}^T}{2} = \sin\alpha[\mathbf{n}]_\times$$

- From these last two relations we can solve for (**n**, α) (how?)

Eigensystem of **R**

- Clearly: **R n** = **n** ⇒
  n is an eigenvector of **R** with eigenvalue 1

- Maybe less clear:

  R (**p** + i **q**) = $e^{i\alpha}$ (**p** + i **q**)

  R (**p** - i **q**) = $e^{-i\alpha}$ (**p** - i **q**)

$i^2 = -1$

(**p** + i **q**) is an eigenvector of **R** with eigenvalue $e^{i\alpha}$

(**p** - i **q**) is an eigenvector of **R** with eigenvalue $e^{-i\alpha}$

(why?)

Eigensystem of **R**

- The eigenvalues of **R** are (1, $e^{i\alpha}$, $e^{-i\alpha}$)
- They are the solutions to det(**R** - λ**I**) = 0
- The corresponding *normalized* eigenvectors are

$$(\mathbf{n}, \frac{\mathbf{p}+i\,\mathbf{q}}{\sqrt{2}}, \frac{\mathbf{p}-i\,\mathbf{q}}{\sqrt{2}})$$

p and **q** are not uniquely defined (why?)

- (**n**, α) are given by an EVD of **R**

Eigensystem of **R**

- In summary we can write

Complex conjugation and transpose

$$\mathbf{R} = \begin{pmatrix}\mathbf{n} & \frac{\mathbf{p}+i\,\mathbf{q}}{\sqrt{2}} & \frac{\mathbf{p}-i\,\mathbf{q}}{\sqrt{2}}\end{pmatrix} \begin{pmatrix}1 & 0 & 0\\ 0 & e^{i\alpha} & 0\\ 0 & 0 & e^{-i\alpha}\end{pmatrix} \begin{pmatrix}\mathbf{n} & \frac{\mathbf{p}+i\,\mathbf{q}}{\sqrt{2}} & \frac{\mathbf{p}-i\,\mathbf{q}}{\sqrt{2}}\end{pmatrix}^{*}$$

- **R** = **E D E***
- **E** is a unitary basis: **E*****E** = **I**
- Can we connect this to Rodrigues' formula?

Matrix exponentials

- For a vector space *V* and a linear transformation **T**: *V* → *V* we define the matrix exponential of **T** as

$$e^{\mathbf{T}} = \sum_{k=0}^{\infty} \frac{1}{k!}\mathbf{T}^k = \mathbf{I} + \mathbf{T} + \tfrac{1}{2}\mathbf{T}^2 + \ldots$$

- This series is absolute convergent for any **T**, with $\mathbf{T}^0 = \mathbf{I}$
- $e^{\mathbf{T}}$ is linear transformation: *V* → *V*

Matrix exponentials

General properties:

- $e^{\mathbf{0}} = \mathbf{I}$
- $e^{a\mathbf{T}} e^{b\mathbf{T}} = e^{(a+b)\mathbf{T}}$ (why?)
- $e^{\mathbf{T}^T} = (e^{\mathbf{T}})^T$ (why?)
- $e^{-\mathbf{T}} = (e^{\mathbf{T}})^{-1}$ (why?)
- $e^{\mathbf{EDE}^*} = \mathbf{E}\, e^{\mathbf{D}}\, \mathbf{E}^*$ for unitary $\mathbf{E}$ ($\mathbf{E}^*\mathbf{E} = \mathbf{I}$) (why?)
- $e^{\mathbf{D}} = \text{diag}(e^{d_1}, e^{d_2}, \ldots)$ for $\mathbf{D} = \text{diag}(d_1, d_2, \ldots)$ (why?)

so(3)

- The set of skew-symmetric matrices is denoted so(3)
- $[\mathbf{m}]_\times \in so(3), \Rightarrow [\mathbf{m}]_\times = -[\mathbf{m}]_\times^T$
- $e^{[\mathbf{m}]_\times} = e^{-[\mathbf{m}]_\times^T} = (e^{-[\mathbf{m}]_\times})^T = ((e^{[\mathbf{m}]_\times})^{-1})^T$
  $\Rightarrow e^{[\mathbf{m}]_\times} \in SO(3)$
- The matrix exponential maps $so(3) \rightarrow SO(3)$

Eigensystem of $\alpha[\mathbf{n}]_\times$

- Clearly: $\alpha[\mathbf{n}]_\times \mathbf{n} = \mathbf{0} \Rightarrow$
  $\mathbf{n}$ is an eigenvector of $\alpha[\mathbf{n}]_\times$ with eigenvalue 0
- Furthermore:

$\alpha[\mathbf{n}]_\times (\mathbf{p} + i\,\mathbf{q}) = i\alpha\, (\mathbf{p} + i\,\mathbf{q})$

$\alpha[\mathbf{n}]_\times (\mathbf{p} - i\,\mathbf{q}) = -i\alpha\, (\mathbf{p} - i\,\mathbf{q})$

$i^2 = -1$

($\mathbf{p} + i\,\mathbf{q}$) is an eigenvector of $\alpha[\mathbf{n}]_\times$ with eigenvalue $i\alpha$

($\mathbf{p} - i\,\mathbf{q}$) is an eigenvector of $\alpha[\mathbf{n}]_\times$ with eigenvalue $-i\alpha$

(why?)

Eigensystem of $\alpha[\mathbf{n}]_\times$

- The eigenvalues of $\alpha[\mathbf{n}]_\times$ are $(0, i\alpha, -i\alpha)$
- The corresponding *normalized* eigenvectors are

$$(\mathbf{n}, \frac{\mathbf{p} + i\,\mathbf{q}}{\sqrt{2}}, \frac{\mathbf{p} - i\,\mathbf{q}}{\sqrt{2}})$$

- Same eigenvectors as $\mathbf{R}$!
- $\alpha[\mathbf{n}]_\times = \mathbf{E}\,\mathbf{D}'\mathbf{E}^*$ with $\mathbf{D}' = \text{diag}(0, i\alpha, -i\alpha)$
- Note: $\mathbf{D} = e^{\mathbf{D}'}$

so(3) $\rightarrow$ SO(3)

For **m** = α**n** we get:

$$e^{\alpha[\mathbf{n}]_\times} = e^{\mathbf{E}\mathbf{D}'\mathbf{E}^\star} = \mathbf{E}\, e^{\mathbf{D}'}\, \mathbf{E}^\star = \mathbf{E}\,\mathbf{D}\,\mathbf{E}^\star = \mathbf{R}$$

so(3) $\rightarrow$ SO(3)

Summary:

- The matrix exponential maps $\alpha[\mathbf{n}]_\times$ to **R**
- We can represent any **R** as the skew-symmetric matrix $\alpha[\mathbf{n}]_\times$ which has 3 parameters
- We can represent any **R** as the 3-dim vector **m**=α**n**
- If we restrict **m** to |**m**| < π, this representation is, in principle, one-to-one

Quaternions

- Quaternions are an extension of complex numbers, with 4 components instead of 2
- Quaternions form an associative division algebra
  - They can be added, subtracted, multiplied, and divided
  - Are non-commutative
- Can represented as a 4-dim vector
- Alternatively as a scalar + a vector

Quaternion algebra

- Using the scalar+vector notation:

$q_1 = (s_1, \mathbf{v}_1), \quad q_2 = (s_2, \mathbf{v}_2)$

$q_1 + q_2 = (s_1 + s_2, \mathbf{v}_1 + \mathbf{v}_2)$

$q_1 \cdot q_2 = (s_1 s_2 - \mathbf{v}_1 \cdot \mathbf{v}_2, s_1 \mathbf{v}_2 + s_2 \mathbf{v}_1 + v_1 \times v_2)$

$q_1^{-1} = (s_1, -\mathbf{v}_1)/(s_1^2 + |\mathbf{v}_1|^2) \Rightarrow q_1 q_1^{-1} = (1, \mathbf{0})$

Unit quaternions

q=(s, v)

- $|q|^2 = s^2+|v|^2$
- Unit quaternions satisfy $|q|^2 = 1$
- Represents the unit sphere in R^4, denoted S^3
- Any unit quaternion can be written

  q = (cos α/2, sin α/2 **n**) for some angle α and vector |n|=1 (why?)

- In this case q^{-1} = (cos α/2, -sin α/2 **n**)

Quaternion representation of rotations

- Let **u** $\in$ R^3 and represent it by the quaternion p = (0, **u**)
- Let q = (cos α/2, sin α/2 **n**) be a unit quaternion
- Gives q^{-1} = (cos α/2, -sin α/2 **n**)
- Consider the quaternion product qpq^{-1}

$$p\,q^{-1} = (\sin\tfrac{\alpha}{2}(\mathbf{n}\cdot\mathbf{u}), \cos\tfrac{\alpha}{2}\mathbf{u} + \sin\tfrac{\alpha}{2}(\mathbf{n}\times\mathbf{u}))$$

Quaternion representation of rotations

- Finally, we get

$$qpq^{-1} = (\cos\tfrac{\alpha}{2}, \sin\tfrac{\alpha}{2}\mathbf{n}) \cdot (\sin\tfrac{\alpha}{2}(\mathbf{n}\cdot\mathbf{u}), \cos\tfrac{\alpha}{2}\mathbf{u} + \sin\tfrac{\alpha}{2}(\mathbf{n}\times\mathbf{u}))$$

$$qpq^{-1} = \left(0, \cos^2\tfrac{\alpha}{2}\mathbf{u} + 2\cos\tfrac{\alpha}{2}\sin\tfrac{\alpha}{2}(\mathbf{n}\times\mathbf{u}) + \sin^2\tfrac{\alpha}{2}\mathbf{n}\mathbf{n}^T\mathbf{u} + \sin^2\tfrac{\alpha}{2}\mathbf{n}\times(\mathbf{n}\times\mathbf{u})\right)$$

$$qpq^{-1} = \left(0, \cos^2\tfrac{\alpha}{2}\mathbf{u} + \sin\alpha[\mathbf{n}]_\times\mathbf{u} + \sin^2\tfrac{\alpha}{2}(\mathbf{I} + [\mathbf{n}]_\times^2)\mathbf{u} + \sin^2\tfrac{\alpha}{2}[\mathbf{n}]_\times^2\mathbf{u}\right)$$

$$qpq^{-1} = \left(0, \mathbf{u} + \sin\alpha[\mathbf{n}]_\times\mathbf{u} + (1-\cos\alpha)[\mathbf{n}]_\times^2\mathbf{u}\right)$$

$$qpq^{-1} = (0, \mathbf{R}\mathbf{u})$$

Quaternion representation of rotations

Summary:

- We can represent points in R^3 as "imaginary" quaternions p
- The rotation (α, **n**) is represented as the unit quaternion q=(cos α/2, sin α/2 **n**)
- These consists of the set S^3
- The rotated point is computed as the *sandwich product* qpq^{-1}

Quaternion representation of rotations

- Composition of two rotations in standard 3×3 matrix algebra:
  - 27 mult
  - 18 add
- Composition of two rotations in quaternion algebra:
  - 16 mult
  - 12 add

The orthogonal Procrustes problem

- Given n known vectors $\mathbf{a}_k$ and $\mathbf{b}_k$, which orthogonal $\mathbf{R}$ minimizes

$$\sum_{k=1}^{n} \|\mathbf{a}_k - \mathbf{R}\mathbf{b}_k\|^2$$

The orthogonal Procrustes problem

Cookbook solution:

- See [Golub & Van Loan, *Matrix Computations*]
- Let $\mathbf{A}$ be a matrix with all $\mathbf{a}_k$ in its columns
- Let $\mathbf{B}$ be a matrix with all $\mathbf{b}_k$ in its columns
- $[\mathbf{U}\ \mathbf{S}\ \mathbf{V}] = \mathrm{svd}(\mathbf{A}\ \mathbf{B}^T)$
- $\mathbf{R} = \mathbf{U}\ \mathbf{V}^T$
- Note: $\mathbf{R}$ is in O(3) but may not be in SO(3)!

Estimation of absolute orientation

- Given two set of n corresponding 3D points $\mathbf{a}_k$ and $\mathbf{b}_k$ that are related by a rigid transformation:

$$\mathbf{a}_k = \mathbf{R}\ \mathbf{b}_k + \mathbf{t}$$

How can we determine $\mathbf{R}$ and $\mathbf{t}$?

In particular when there is noise present?

Estimation of absolute orientation

- Let $\mathbf{a}'$ and $\mathbf{b}'$ denote the centroids of the set $\mathbf{a}_k$ and the set $\mathbf{b}_k$, respectively:

  $\mathbf{a}' = \mathbf{R}\,\mathbf{b}' + \mathbf{t} \;\Rightarrow\; \mathbf{t} = \mathbf{a}' - \mathbf{R}\,\mathbf{b}'$

- We need to find $\mathbf{R}$ such that

  $\mathbf{a}_k - \mathbf{a}' = \mathbf{R}\,(\mathbf{b}_k - \mathbf{b}')$

- $\mathbf{R}$ can be found using the orthogonal Procrustes method
- Once $\mathbf{R}$ is determined, $\mathbf{t}$ is given by $\mathbf{a}' - \mathbf{R}\,\mathbf{b}'$
- See [Horn, *Closed-form solution of absolute orientation using unit Quaternions,* JOSA, 1987]