3D Interpolation Using Co-Helicity

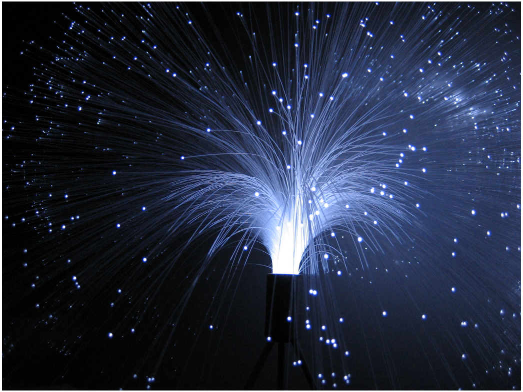
Olivier St-Martin Cormier

McGill University

April 8, 2010

Goal of the project

Represent a tree by a cluster of fibers emanating from a small region of space

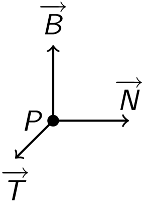


[src:http://www.flickr.com/]

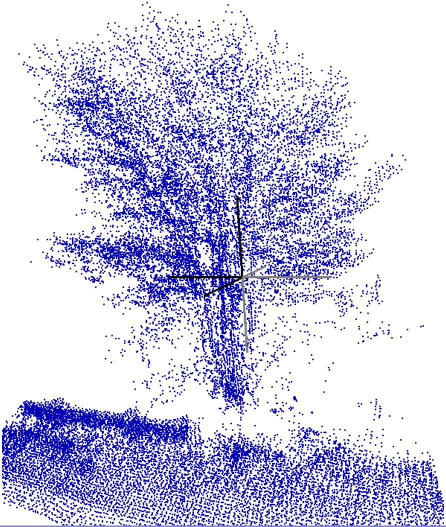
Input Data

[src:http://www.wikipedia.org/]

- LIDAR Scan of a Tree
- 30947 Points
- Preprocessed data
- Frenet-Serret Frame at each point

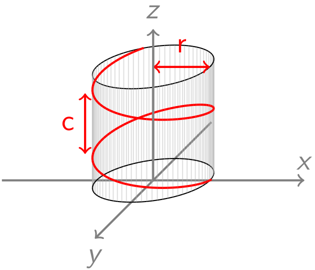


Inital Data Representation



Representation of a Helix

$$H(t) = \Big(x(\theta), y(x(\theta), z(x(\theta)\Big) = \Big(r\cos(\theta), r\sin(\theta), c\theta\Big)$$



Derivation of the Unit Tangent of an Helix

$$
\begin{aligned}
T(t) &= \frac{H'(t)}{\|H'(t)\|} \\
&= \frac{\big(-r\sin(\theta), r\cos(\theta), c\big)}{\big([-r\sin(\theta)]^2 + [r\cos(\theta)]^2 + c^2\big)^{1/2}} \\
&= \frac{\big(-r\sin(\theta), r\cos(\theta), c\big)}{\big(r^2[\sin^2(\theta) + \cos^2(\theta)] + c^2\big)^{1/2}} \\
&= \frac{\big(-r\sin(\theta), r\cos(\theta), c\big)}{\big(r^2 + c^2\big)^{1/2}}
\end{aligned}
$$

Derivation of the Unit Normal of an Helix

$$
\begin{aligned}
N(t) &= \frac{T'(t)}{\|T'(t)\|} \\
&= \frac{\big(-r\cos(\theta), -r\sin(\theta), 0\big)}{\big([-r\cos(\theta)]^2 + [-r\sin(\theta)]^2 + 0^2\big)^{1/2}} \\
&= \frac{\big(-r\cos(\theta), -r\sin(\theta), 0\big)}{\big(r^2[\sin^2(\theta) + \cos^2(\theta)]\big)^{1/2}} \\
&= \frac{\big(-r\cos(\theta), r\sin(\theta), 0\big)}{\big(r^2\big)^{1/2}} \\
&= \Big(-\cos(\theta), -\sin(\theta), 0\Big)
\end{aligned}
$$

Summary of Helix Equations

- Equation

$$H(t) = \Big(r\cos(\theta), r\sin(\theta), c\theta \Big)$$

- Unit Tangent

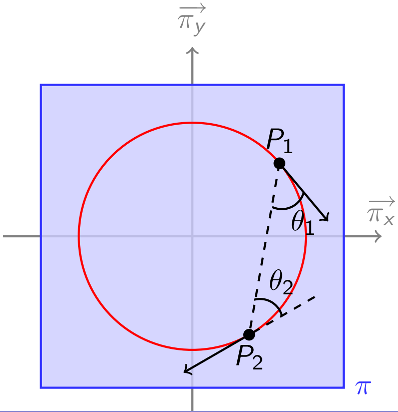
$$T(t) = \frac{1}{\sqrt{r^2 + c^2}} \cdot \Big(-r\sin(\theta), r\cos(\theta), c \Big)$$

- Unit Normal

$$N(t) = \Big(-\cos(\theta), -\sin(\theta), 0 \Big)$$

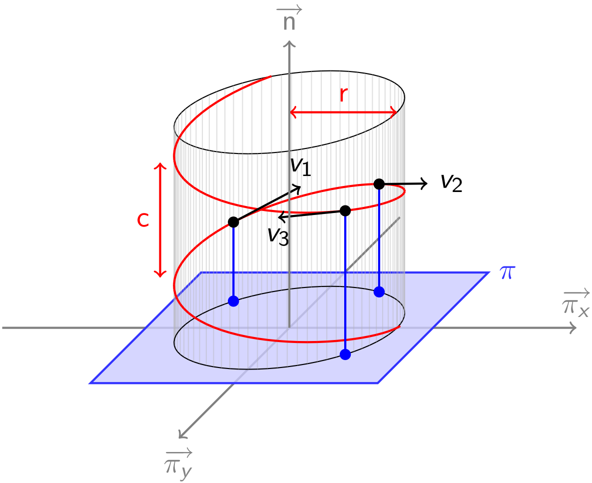
Review of Co-Circularity [1]

Two points on a circle are co-circular if the magnitude of the angle between their respective tangent and the line passing through the two points are equal. That is, we need $|\theta_1| = |\theta_2|$



Co-Helicity

Extension of co-circularity in the third dimension

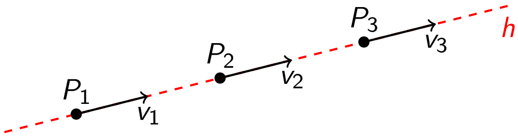


Algorithm to Determine Co-Helicity [2]

Step 1 - Check for collinearity

Two conditions need to be checked:

- P_1,P_2, and P_3 on the same line
- v_1,v_2, and v_3 tangent to the line formed by P_1,P_2, and P_3



Algorithm to Determine Co-Helicity [2]

Step 2 - Find the main axis of the possible helix

$$\overrightarrow{n} = (v_3 - v_2) \times (v_2 - v_1)$$

Step 3 - Find the projections of the points and the tangent estimates on the plane $\pi \perp \overrightarrow{n}$

$$proj_\pi(v) = v - \frac{\langle v, \overrightarrow{n} \rangle}{|\overrightarrow{n}|^2} \cdot \overrightarrow{n}$$

Algorithm to Determine Co-Helicity [2]

Step 4 - Check for pairwise co-circularity of the projected points and tangent estimates

Step 5 - Find the radius of the circumcircle of the projected points forming a triangle with sides a, b, and c

$$\text{Heron's Formula: } A_{\triangle} = \frac{\sqrt{(a^2+b^2+c^2)^2 - 2(a^4+b^4+c^4)}}{4}$$

$$R = \frac{a \cdot b \cdot c}{4 \cdot A_{\triangle}}$$

Algorithm to Determine Co-Helicity [2]

Step 6 - Assume $\theta_1 = 0$ and find θ_2

$$\theta_2 = \arccos\left(\langle \frac{\text{proj}_\pi(v_1)}{\|\text{proj}_\pi(v_1)\|}, \frac{\text{proj}_\pi(v_2)}{\|\text{proj}_\pi(v_2)\|} \rangle\right)$$

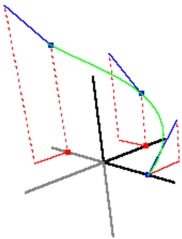
Step 7 - Find c from

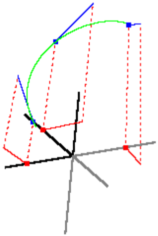
$$\langle v_1, v_2 \rangle = \frac{r^2 \cos(t_2 - t_1) + c^2}{r^2 + c^2}$$

Step 8 - Check that v_n are tangent to the helix

What's Done?

Finding co-helicity support and drawing helices





References

1 Pierre Parent and Steven and W. Zucker, *Trace inference, curvature consistency, and curve detection*. IEEE Transactions on Pattern Analysis and Machine Intelligence 1989

2 Peter Savadjiev and Jennifer S. W. Campbell and G. Bruce Pike and Kaleem Siddiqi, *3D Curve Inference for Diffusion MRI Regularization*. 2005

3 Peter Savadjiev and Steven W. Zucker and Kaleem Siddiqi, *On the Differential Geometry of 3D Flow Patterns: Generalized Helicoids and Diffusion MRI Analysis*. International Conference on Computer Vision (ICCV) 2007

Questions?