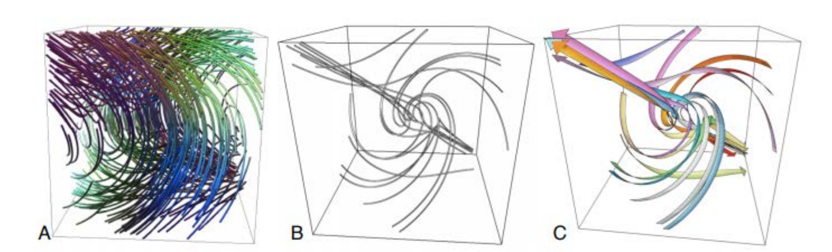


An Illustrative Visualization Framework for 3D Vector Fields

Cheng-Kai Chen, Shi Yan, Hongfeng Yu, Nelson Max, Kwan-Liu Ma

By: Michael Tran

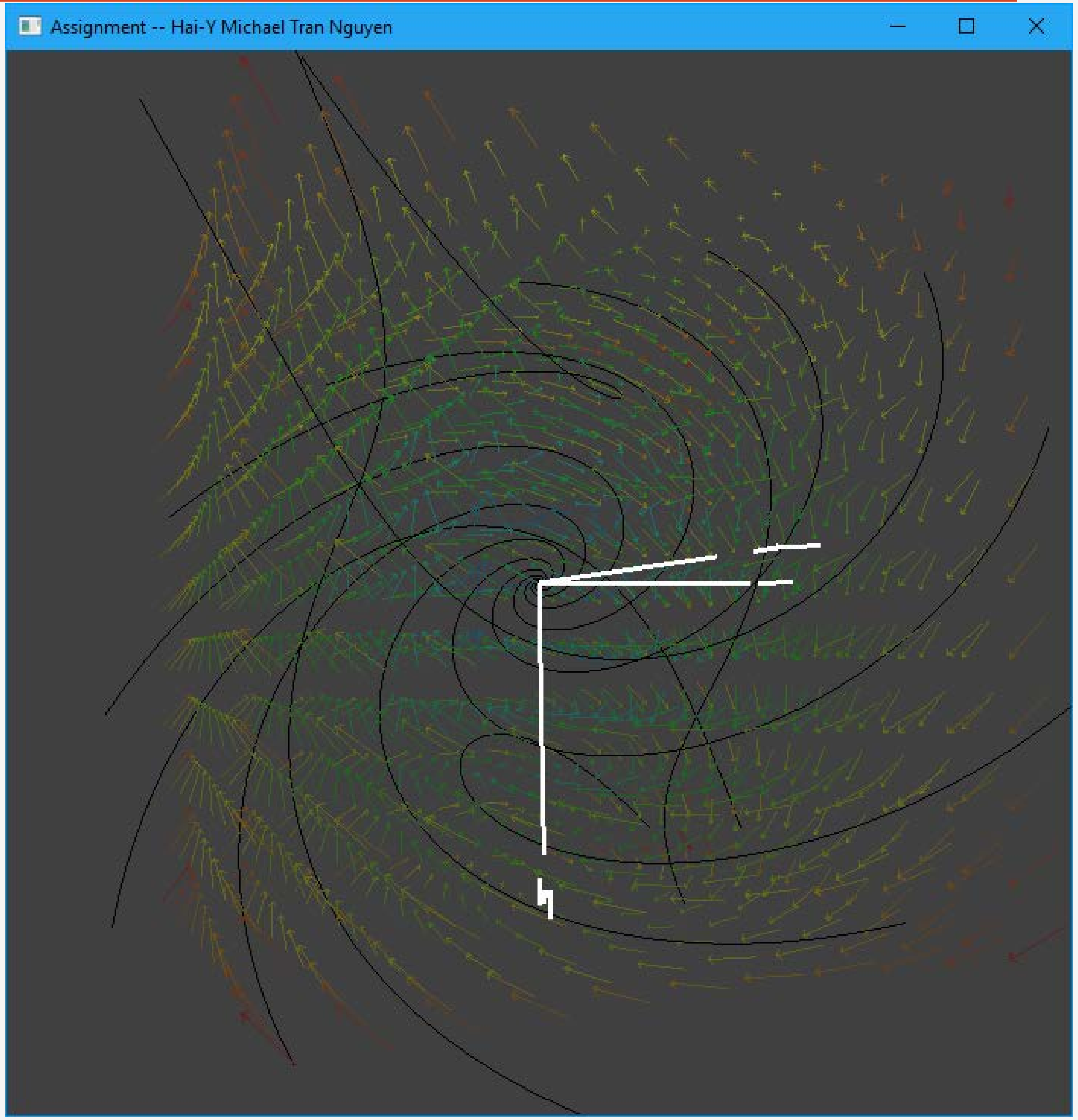
COSC 6344 – Visualization

11/30/2018

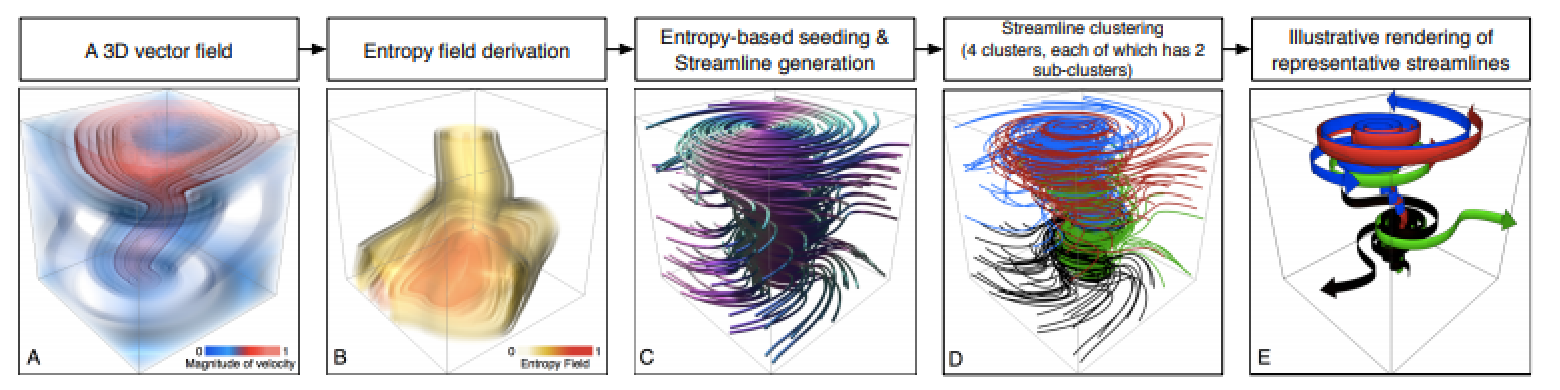
Goals

Recall: Assignment 7…

Instead of manually choosing seed locations or using stream/stream ribbon probes, we want to seed and visualize automatically.



Overview



1. Entropy-Based Seeding

- Histogram/Binning, Entropy Calculations, Seeding

2. Two-Stage Streamline Clustering

- K-means Algorithm on Vector Spatial Properties
- K-means Algorithm on Vector Shape Properties

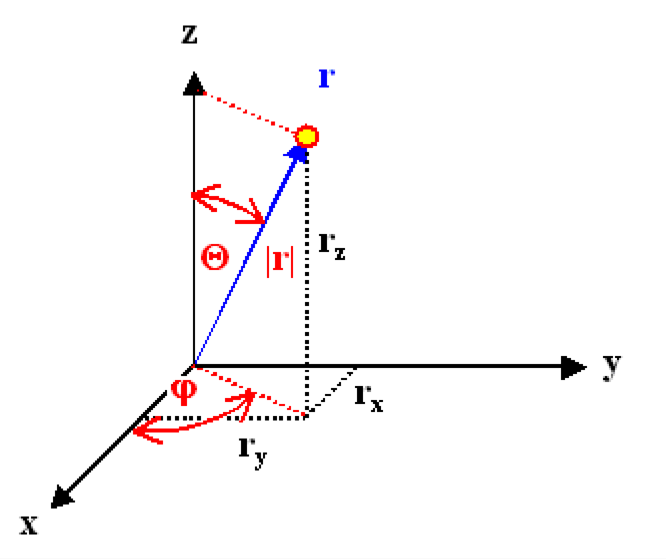
3. Streamtape Generation

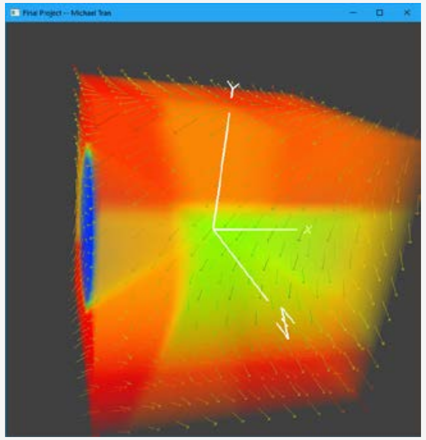
- Binormal Vector, Tangent Vectors, Normal Vectors, Torsion

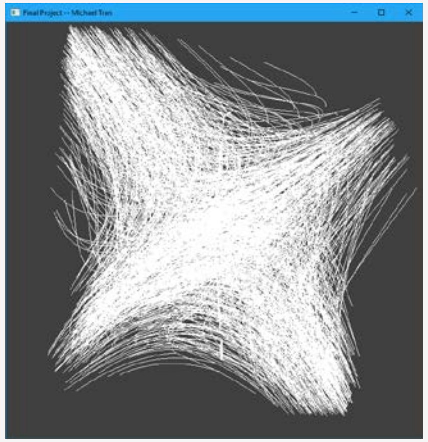
Entropy-Based Seeding

To calculate Entropy:

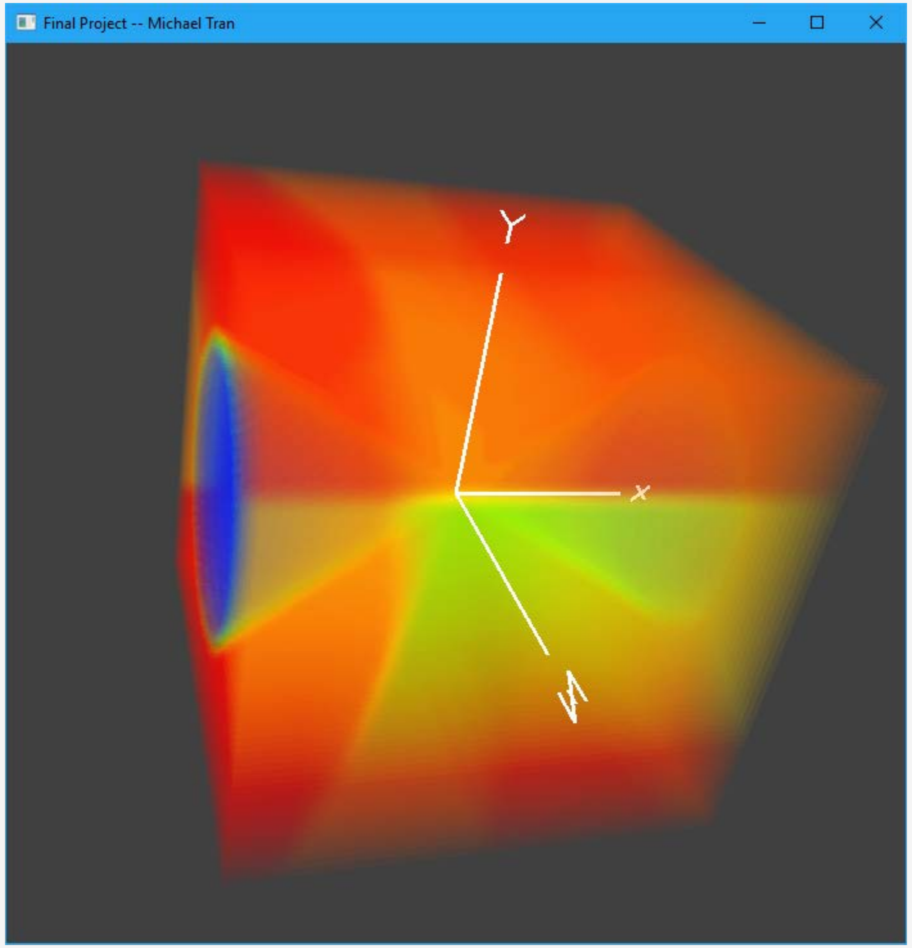
- Bin each voxel in the vector field depending on **Phi** (φ) and **Theta** (θ).
- Calculate **Probability** in each voxel: $$p(x_i) = \frac{C(x_i)}{\sum_{i=1}^{n} C(x_i)},$$
- Calculate **Entropy** in each voxel using a 3x3x3 neighborhood as reference: $$H(x) = -\sum_{x_i \in X} p(x_i) log_2 p(x_i),$$
- Shuffle all points and choose 1000 random points
  - Give a point with higher entropy a higher chance of being chosen.



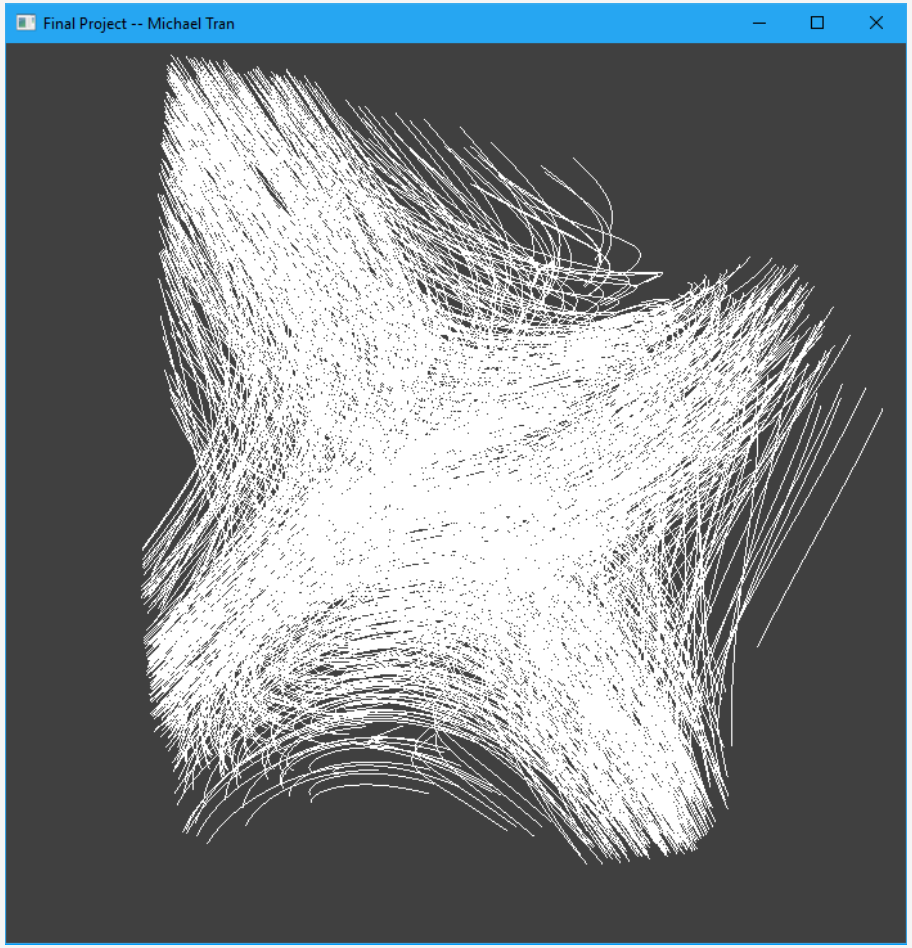




Entropy-Based Seeding - Results



Entropy Field

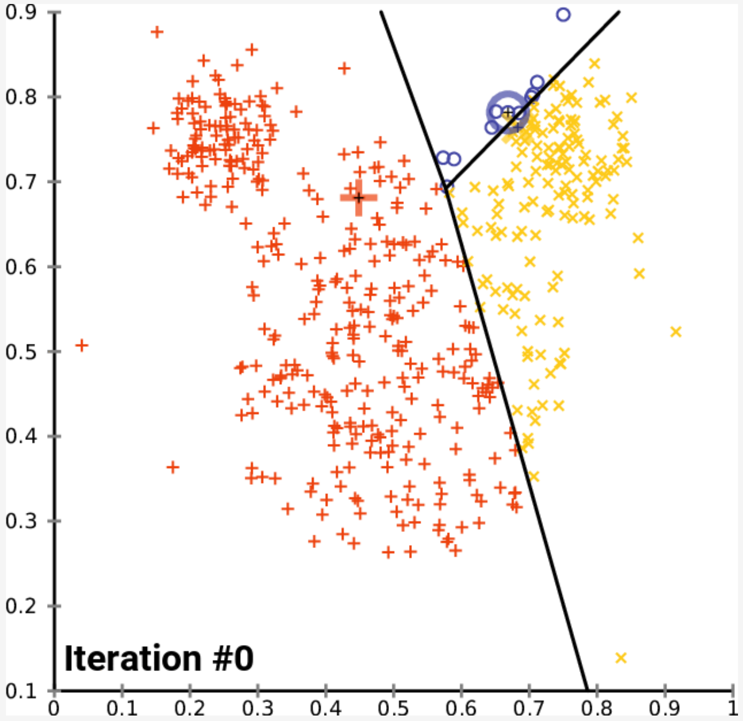


1000 Seeds Picked Based on Entropy Field

Streamline Clustering

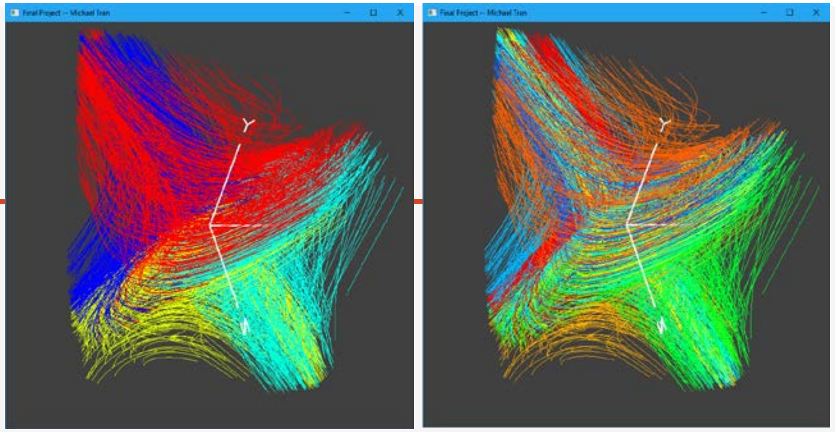
To cluster properly, we cluster **two times** – cluster all streamlines, and then cluster again on the clusters from the previous clustering.

To cluster, we employ the **k-means clustering algorithm** based on certain vector properties.



https://en.wikipedia.org/wiki/K-means_clustering

K-means Algorithm



First Stage: K-means Algorithm on Vector Spatial Properties

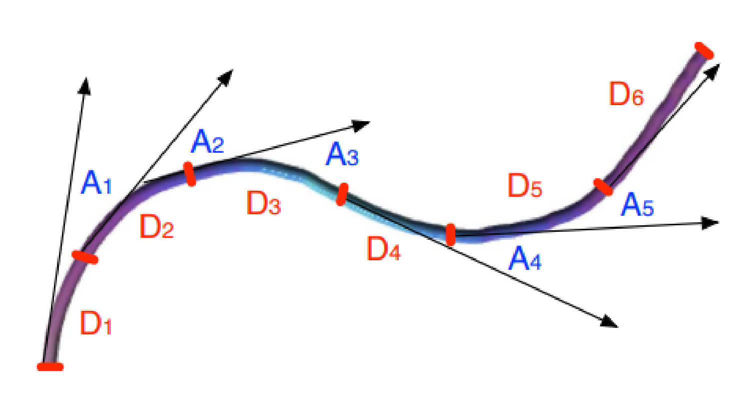
- For the first clustering we take the streamline **start/mid/end point** as our clustering property.

Second Stage: K-means Algorithm on Vector Shape Properties

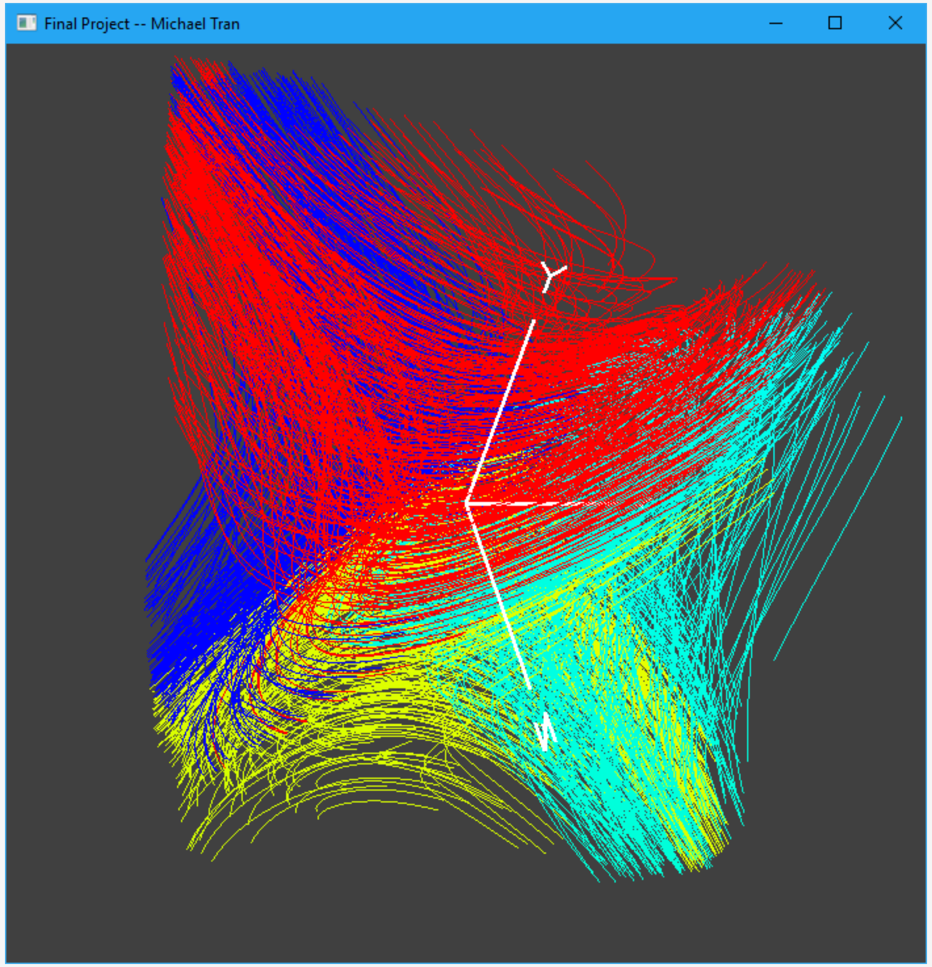
- We take the clusters from the first stage and apply K-means algorithm again on each cluster. We use **Linear Entropy (E_L)** and **Angular Entropy (E_A)** as our clustering property.

$$E_L = -\frac{1}{\log_2(m+1)} \sum_{j=0}^{m} \frac{D_j}{L_S} \log_2 \frac{D_j}{L_S}$$

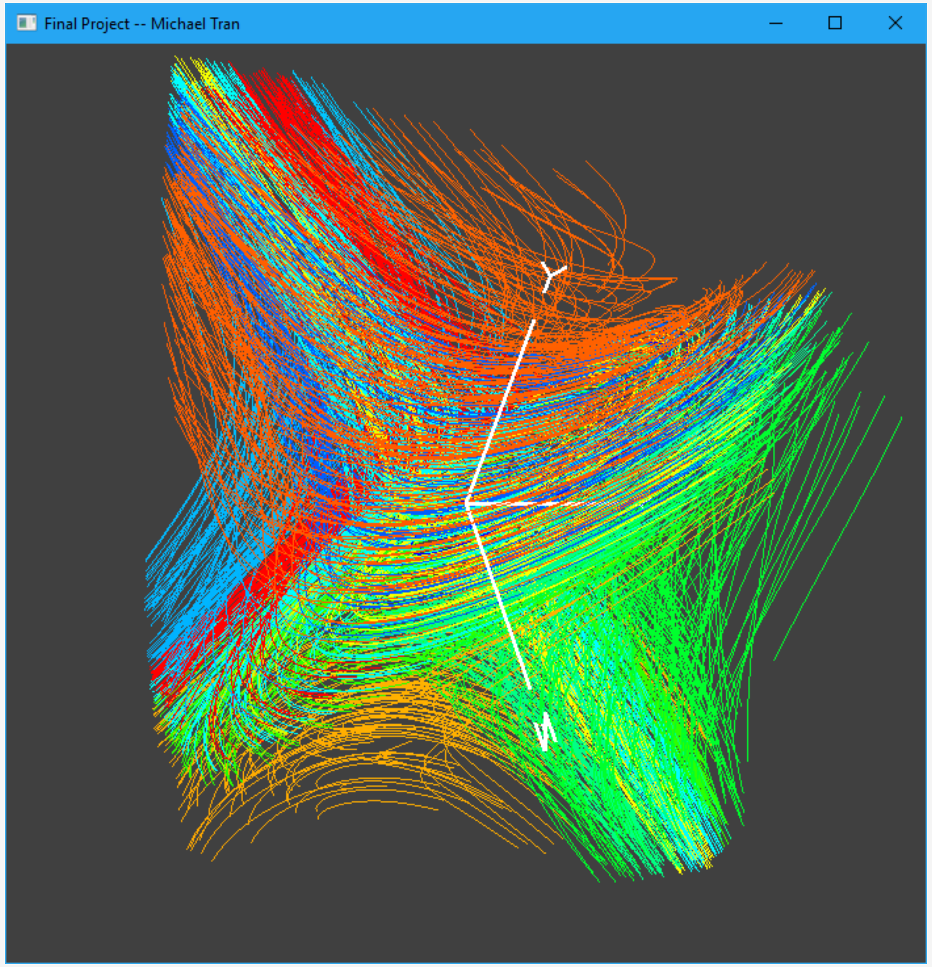
$$E_A = -\frac{1}{\log_2(m)} \sum_{j=0}^{m-1} \frac{A_j}{L_A} \log_2 \frac{A_j}{L_A}$$



Streamline Clustering - Results



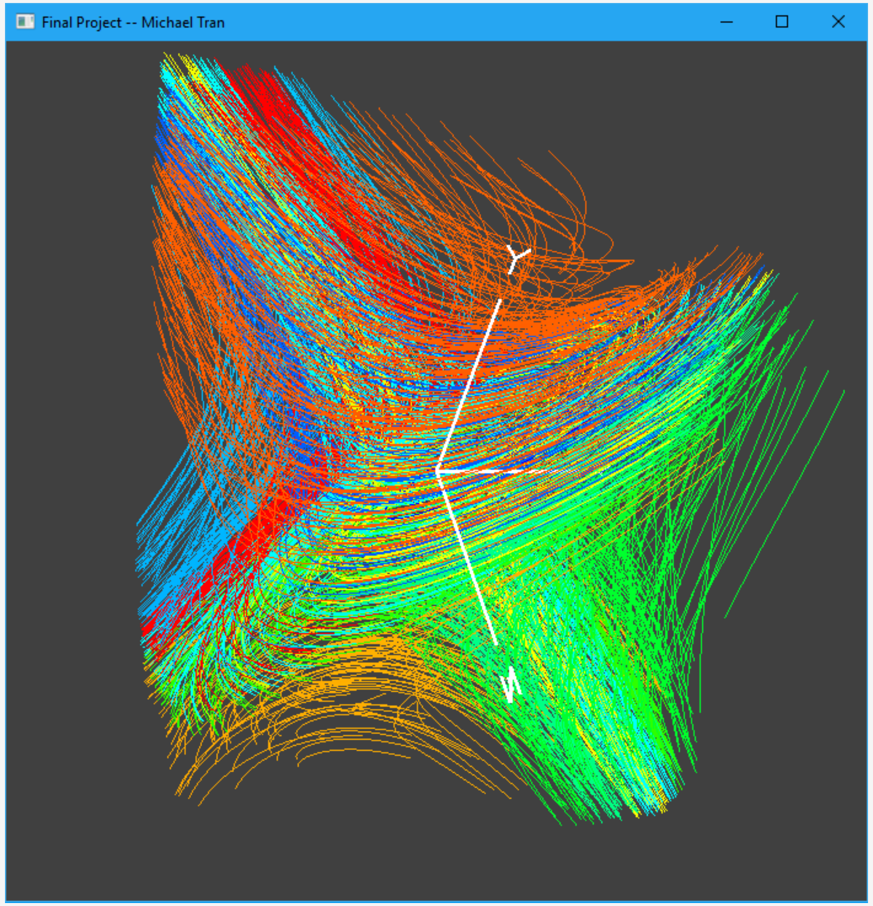
Stage 1 Clustering



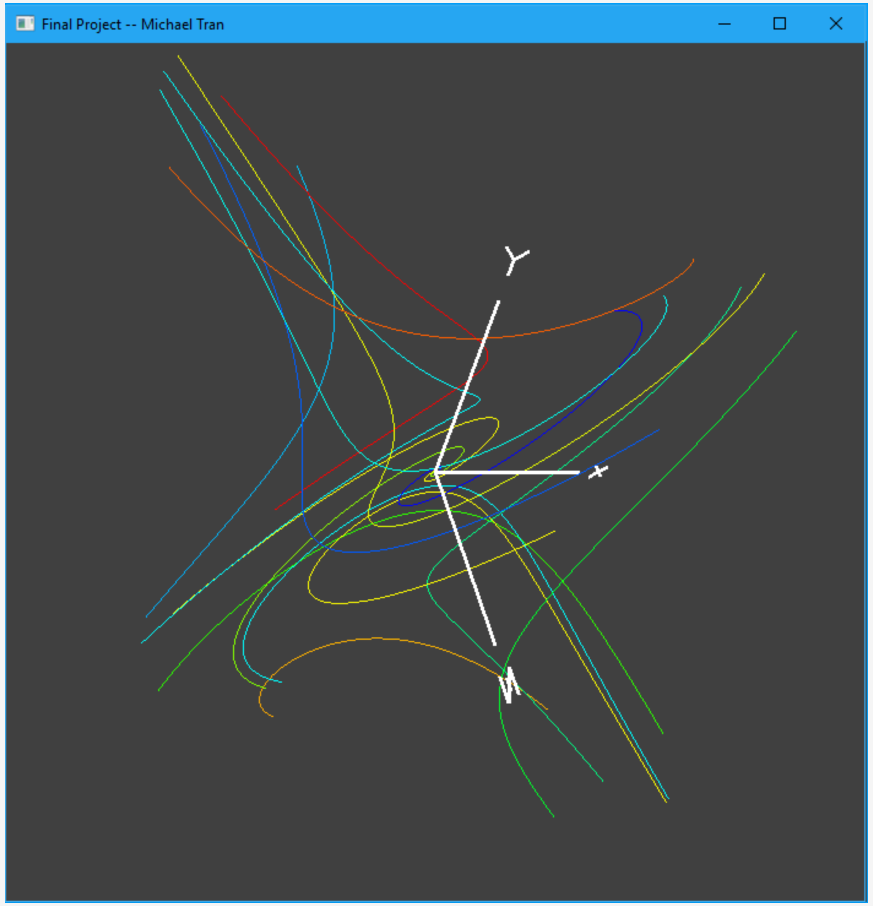
Stage 2 Clustering

Streamline Clustering – Bundling

After all streamlines are clustered, we pick the streamline closest to each centroid as the candidate streamline.



Stage 2 - Clustering



Bundled Streamlines

Streamtape Generation

At each point on a streamline, we calculate the following variables:

Binormal Vector **B**, Tangent Vector **V**, Normal Vector **N**, Torsion **τ**, Width **ω**

$$\begin{aligned} B &= T \times N \\ T &= dr/ds \\ N &= dT/ds \end{aligned}$$

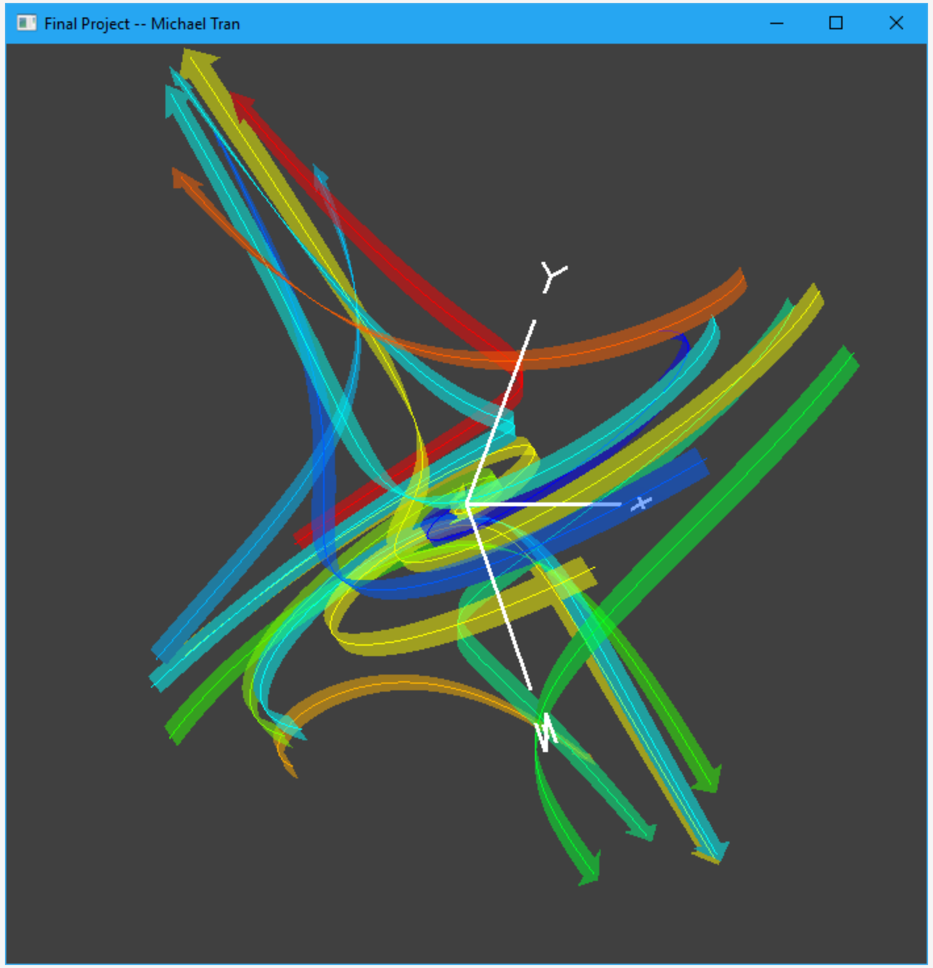
$$\tau = \frac{det(r', r'', r''')}{||r' \times r''||^2}$$

$$\omega = 1.0 - \tau.$$

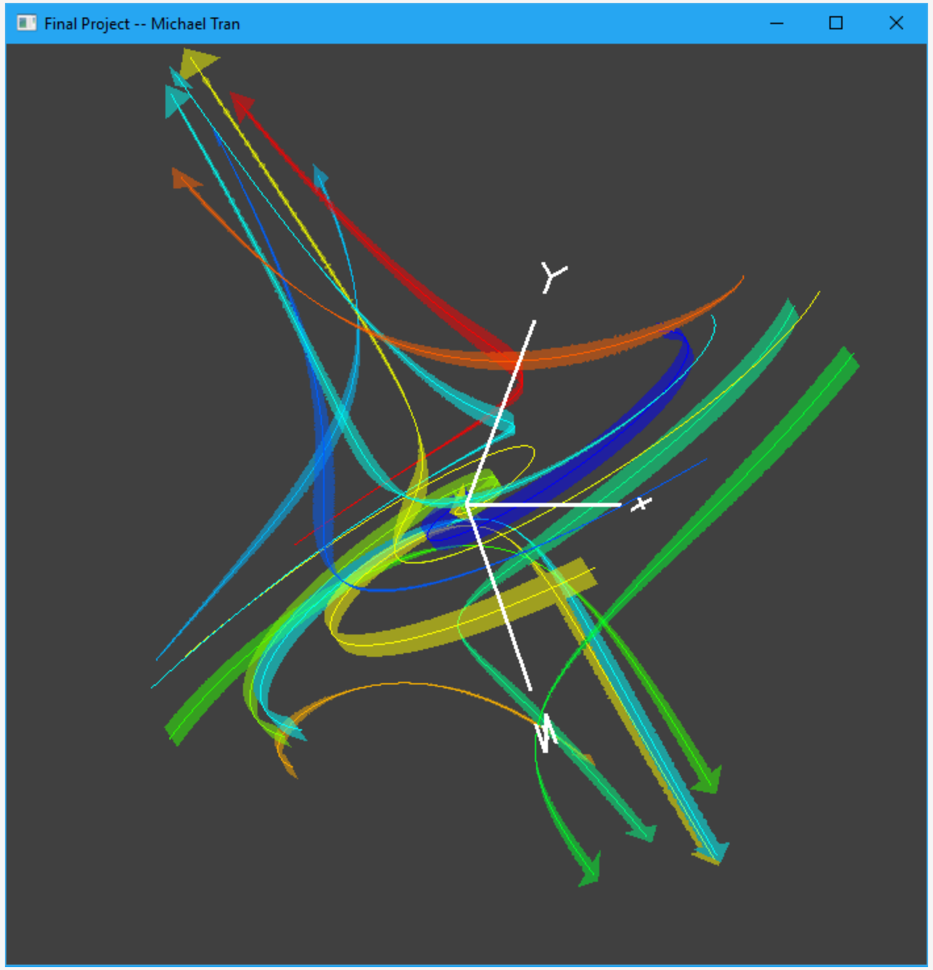
r′, r″, r‴ are the first, second, and third derivatives of r(s) (the streamline).

Once we have the values, we can render the stream tape.

Streamtape Generation - Results



Streamtapes (Without Factoring Torsion)



Streamtapes (With Torsion)

Demo