1 How to estimate the radius of curvature at a vertex?

What is the radius of curvature, \( r(c) \), at \( C \)?

Method 1: Fit a circle to \((B, C, D)\)

\[
    r_1(c) = \frac{(L_1)^2}{8h}, \quad L_1 = \|BC\| + \|CD\| \tag{1}
\]

Method 2: Fit a parabola.

\[
    r_2(c) = \frac{(L_2)^2}{8h}, \quad L_2 = \|BD\| \tag{2}
\]

Notice that \( L_1 \) and \( L_2 \) are converging so that \( L_1 = L_2 \). When \( h \) is strong \( L_2 \) is better. \( L_1 \) also has no concept of the order of the vertices. Method 2 is Jarek recommended.

2 How to estimate the normal at vertex \( C \)?

Tangent at \( C \) is simply:

\[
    T_C = \frac{T_{BC} + T_{CD}}{2} \tag{3}
\]

\( T_C \parallel BD \) (they are parallel)
3 "Visual" fidelity

Not so much about geometry...more about the way it looks.
Maps (roads)
- topological consistency between real world and map.
- left, straight, winding.

We need to preserve the qualitative attributes of a curve.
Examples of things to preserve:

- Extrema
  - Number of extrema
  - Relative size, position.

- Orientation (with respect to each other)

- Features
  - junction
  - endings
  - max/min
  - inflection point
  - local curvature extrema

The human is good at detecting vertical and horizontal lines. It’s also good at recognizing similar shapes and finding patterns.

4 Measuring the error between curves

Remember the poor man’s Hausdorff distance algorithm?

Problems: trade off between accuracy and computation time. Is there a bound to our Hausdorff approximation?

\[ H \leq h, \text{ where } H \text{ is the real Hausdorff and } h \text{ is the approximation.} \]

5 Hierarchical simplification

a.k.a. Multiresolution. This is the "please the biggest offender" idea.