An Algorithm to Follow Arbitrarily Curved Paths

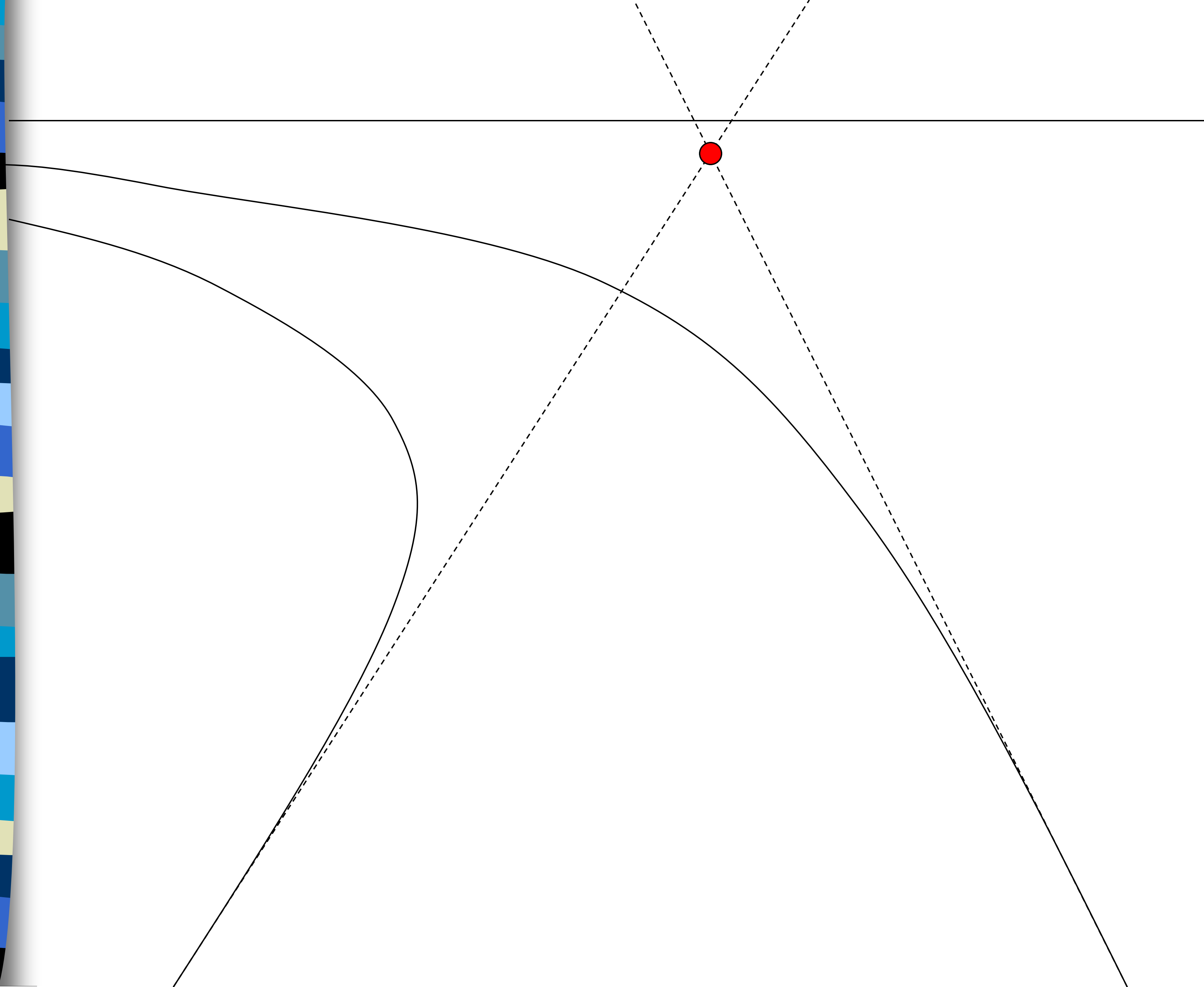
Steven Kapturowski

The Problem

- Given a curved path, find the appropriate movement so as to advance along the path’s center
- Must be able to moderate one’s speed along path so as to minimize acceleration endured by the body
- Should be robust to noise and missing data

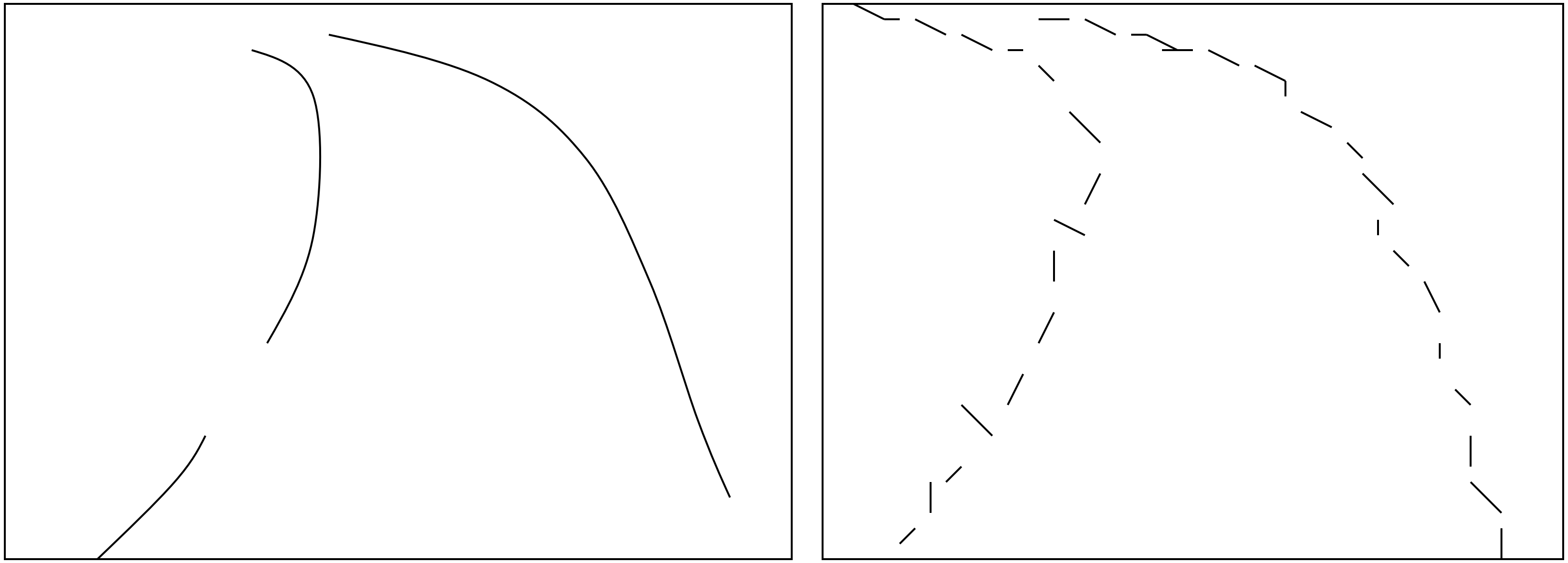
Solution Concept

- We assume image has been preprocessed to find image points of road edges
- Find a best fit curve for each road side
- Use slope of curves to estimate vanishing point
- Correlate radius of curvature to maximum safe velocity



Curve Fitting

- Use cubic spline curve to model road edge data
- Reduces noise compared to working with raw data
- Allows data interpolation when edge regions are missing



…Curve Fitting

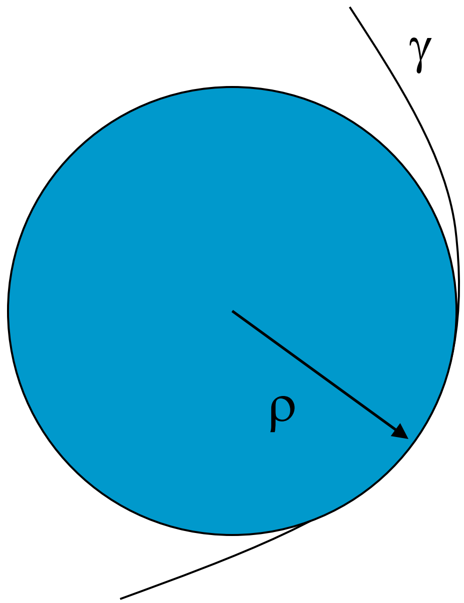
$$\mathbf{c}(\lambda) = \mathbf{v}_1(1-\lambda)^3 + \mathbf{v}_2\lambda(1-\lambda)^2 + \mathbf{v}_3\lambda^2(1-\lambda) + \mathbf{v}_4\lambda^3$$

- Minimize objective function:

  $E(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4) = \Sigma_i |c(\lambda_{min,I}) - p_i|^2$
- Currently using mean squared error (may change to weighted mean squared error as appropriate)
- For fixed **v**, $|c(\lambda) - p_i|$ one can find $\lambda_{min,I}$ by numerically solving $D_\lambda |c(\lambda_{min,I}) - p_i| = 0$ (Quintic Polynomial)
- Nonlinear optimization problem

Why look at curvature?

$$\rho = |\gamma'|^3 / \{ |\gamma'|^2|\gamma''|^2 - (\gamma' \cdot \gamma'')^2\}^{1/2}$$



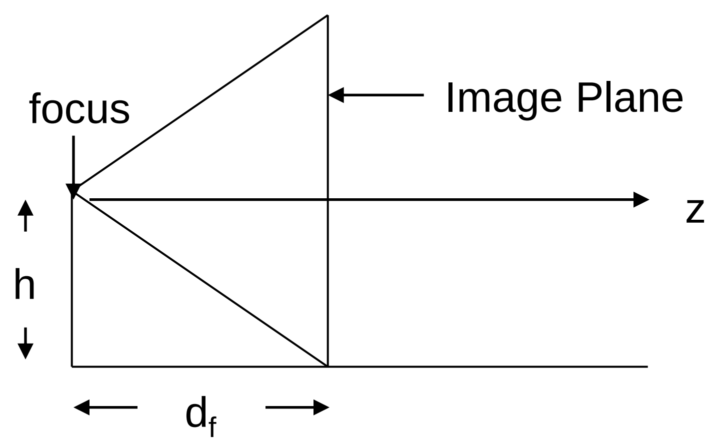
$$a_{Transverse} = v^2 / \rho$$

Need $a_{Transverse} < a_{max}$

$$a = a_{Transverse}^2 + a_{parallel}^2$$

…Curvature Continued

- Should be able to assume ground is level with optical axis for small z
- This allows us to find world curvature near the camera
- Not completely clear what to make of image curvature when z is not small



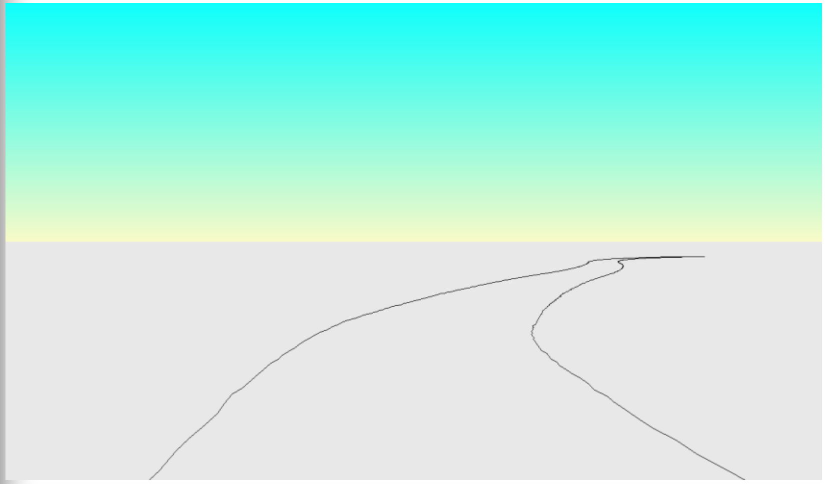
Literature Survey

- Significant work done on edge detection: Ref. [R] focuses on straight roads but detects boundaries in highly non-ideal conditions
- Several papers explore curve fitting to determine vanishing points:
- Ref. [MDMT] explores a road model consisting of concentric circular arcs
- Ref. [WTS] Considers formation of cubic and quartic spline models
- Little work found on velocity control and curvature measurements

Approach Paradigm

- Task-oriented
- 3D *and* motion based vision
- Recovery only of select details

Completed Work



- Developed road simulation model
- Data structures to facilitate algorithms
- ~75% finished with curve fitting routine

Goals (certain to complete)

- Program vanishing point detection
- Continue investigating relationship between image curvature and world curvature
- Test algorithms with incomplete and noisy data

Goals (if time permits)

- Relax idealized parameters
- Test on real world data
- Add lane constraints
- Compare results with different road models
- Explore hybrid road models

References

- [WTS] *Wang, Y., Teoh, E.K., & Shen, D."Lane detection and tracking using B-Snake" Image and Vision Computing 22 (2004)*
- [MDMT] *Morgan, A.D., Dagless, E.L., Milford, D.J., Thomas, B.T. "Road Edge Tracking for Robot Road Following" Image and Vision Computing 8(3) (1990)*
- [PS] *Plass, M., Stone, M. "Curve-fitting with Piecewise Parametric Cubics" Computer Graphics 17(3) (1983)*
- [R] *Rasmussen, C. "Grouping Dominant Orientations for Ill-Structured Road Following" Computer Vision and Pattern Recognition, Proceedings of the 2004 IEEE Computer Society Conference on*

…Curvature Continued

Possible Approach:

- Knowing curvature near camera gives maximum speed
- Can't change speed instantaneously:

  $a_{Total} = \{(\underline{v}^2 / \rho)^2 + (\Delta v/\Delta t)^2\}^{1/2}$, where $\underline{v} = (v_1+v_2)/2$, $\Delta v = v_2 - v_1$

  Quartic equation for v_2 (has exact solution, though even this could be much simplified under certain assumptions)
- Correspond high curvature points between images
- Find time to collision with image plane