Optical Flow Templates
for Obstacle Detection

Presented by Paul Drews


Overview

- Optical flow is the apparent movement of a scene point on the image plane.
- Points move because:
  - Camera is moving
  - Parts of the scene are moving
- For a known depth, you can predict flow due to camera motion.
Optical flow

- Optical flow dependent on:
  - Angular/linear camera velocity (linearly)
  - Scene depth
- Create flow templates
  - Predict flow at pixel locations based on camera velocity
  - For a given scene geometry
Flow Templates

• Templates:
  • Ground Plane (dependent on Theta)
  • Distant Structure \((z = \infty)\)
  • Attitude must be known to calculate \(Z\) for ground plane

• **Linear combination!**

\[
\begin{align*}
\mathbf{u}_i &= \begin{bmatrix}
y_i & \frac{x_i y_i}{f} & -f - \frac{x_i^2}{f} & \frac{x_i}{z_i} & \frac{-f}{z_i} & 0 \\
-x_i & f + \frac{y_i^2}{f} & -\frac{x_i y_i}{f} & \frac{y_i}{z_i} & 0 & -\frac{f}{z_i}
\end{bmatrix} \xi
\end{align*}
\]

\(\xi\) – Camera velocity
\(x_i, y_i\) – Image coordinate
\(z_i\) – Depth at \(i\)
\(f\) – Focal length
Probability of Flow

- Model flow at super-pixel as a Gaussian mixture of flows from each template
  - Ground plane, distant structure, and anomaly categories
  - Probability of flow given:
    - Platform velocity/orientation
    - Category selection

\[
p(u_i | \Lambda_i, \xi) = \mathcal{N}(0, \Sigma^0)^{\Lambda_i^0} \prod_{k=1}^{\kappa} \mathcal{N}(\hat{u}_i^k(\xi), \Sigma)^{\Lambda_i^k}
\]
Refinement

- Bayes rule for likelihood of a label
  \[ l(\Lambda_j | \tilde{u}_j, \xi) = p(\tilde{u}_j | \Lambda_j, \xi) p(\Lambda_j) \]

- Expectation maximization to find \( \xi \)
  \[ \xi \leftarrow \arg \max_{\xi} \langle \mathcal{L}(\xi | \tilde{u}, \Lambda) \rangle \]

- Iterate to convergence
Optical Flow Obstacle Avoidance

Algorithm Overview

Input: Platform attitude

Template 1 ($k = 1$)

- $\omega_x$
- $\omega_y$
- $\omega_z$
- $v_x$
- $v_y$
- $v_z$

Template 2 ($k = 2$)

- $\omega_x$
- $\omega_y$
- $\omega_z$
- $v_x$
- $v_y$
- $v_z$

Each template is a probabilistic optical flow subspace with a basis flow for each velocity component. Linear combination by each component predicts the flow field from egomotion.

Inference

“Obstacle” — not explained by templates

Superpixel labeling

Estimated velocity

Iteratively refine labeling and velocity so observed flow agrees with flow predicted by templates.

Input

Observed frame and optical flow superpixels
Example results

Fig. 2: Example video frames, superpixel optical flow fields, and superpixel labels by our method.
Test results

• Results
  • Blue = extra obstacles
  • Green = Missed obstacles
Failure modes

- Failure modes

(a) Error in labeling the ground as obstacle due to optical flow errors as distant structure due to similarity caused by smooth texture.

(b) Error in labeling ground plane basis flows.
CyCab Obstacle Detection

- Essentially the reverse of the previous paper
  - Calculate theoretical flow for ground
  - Shift image patch by flow amount
  - Calculate image matching
Patch matching methods

- SAD, SSD, ZSAD, ZSSD, ZNCC
- Sum of Absolute Difference and Sum of Square Differences were best

Fig. 6. SSD images corresponding to figure 5. The grey scale corresponds to the SSD value. White means that the SSD is low and black that it is high. The hatched part corresponds to an area of the image which cannot be the ground (the algorithm marks these pixels as unusable).
Pro’s/Cons

- Don’t have to calculate flow in real time
- This method will see a moving shadow as an obstacle
- Assumes flat ground plane and knowledge of platform attitude and distance from ground plane