

# CIS 5800 HW3 Report

Kevin Paulose (63031616)

November 10, 2023

## 1 Algorithm used for depth map (depth.py)

- Find Valid Flow: Identify valid flow locations based on confidence values exceeding a specified threshold (thres).
- Convert to Normalized Projective Coordinates: Convert the pixel locations of valid flow to normalized projective coordinates.
- Normalize Epipole and Flow Vectors: Normalize the epipole (ep) and flow vectors using the inverse of the intrinsic calibration matrix (K).
- Compute depths using the formula derived from lecture slides, involving the normalized epipole, normalized flow vectors, and pixel coordinates:  $\frac{Z}{V_z} = \frac{\|p - FOE\|}{\|\hat{p}\|}$
- Fill Depth Map: Fill in the depth map with the computed depths at the valid flow locations.
- Truncate Depth Map:
  - Ensure depths are positive.
  - Set depths above a specified bound to 0.
  - Normalize the depth map to the range [0, 1].
- Return the processed depth map (truncated depth map).

---

**NOTE 1:** The plots for optical vector fields, epipole and depth map are as follows. The origin is at the top-left of the coordinate system with negative Y-axis and positive X-axis as our convention for solving for epipoles. Hence the epipolar lines converge at the bottom-left in my case.

**NOTE 2:** I have attached a zip file containing all images of optical flow, epipole and depth map (included in this report as well) separately to my code files and report.

## 2 threshmin = 50

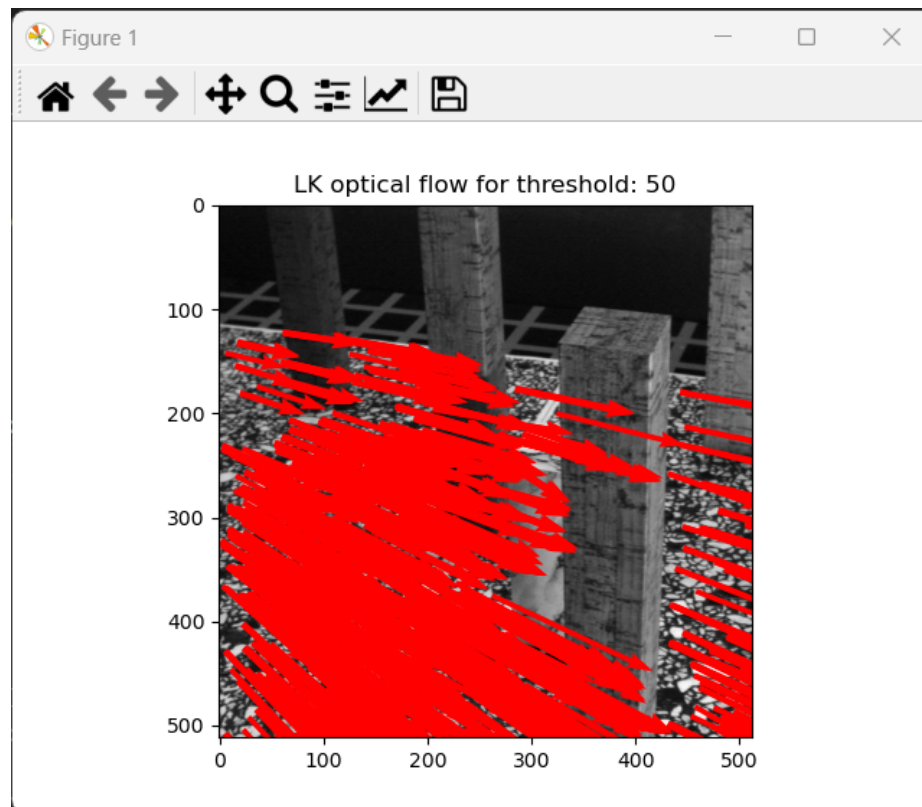


Figure 1: Optical flow lines plot for threshmin = 50

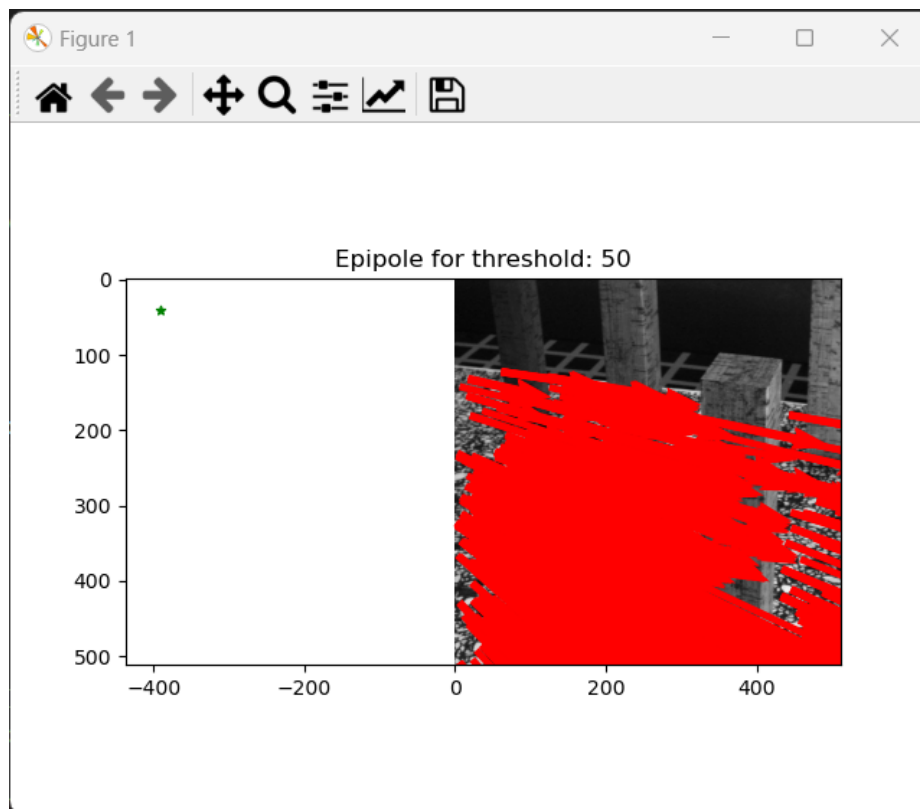


Figure 2: Epipole plot for threshmin = 50

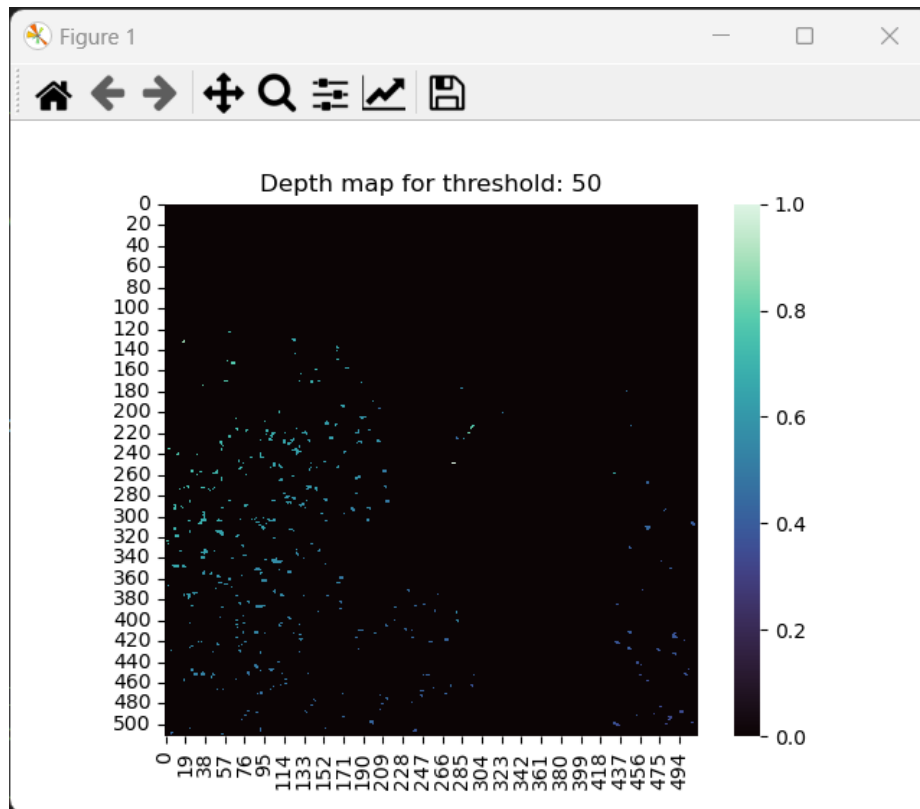


Figure 3: Depth map for threshmin = 50

```
plt.quiver(x[mask], y[mask], u[mask], v[mask], color='red', angles='xy', scale=5,width=0.01)
```

Figure 4: Parameter values used for plt.quiver in plot\_flow function

### 3 threshmin = 30

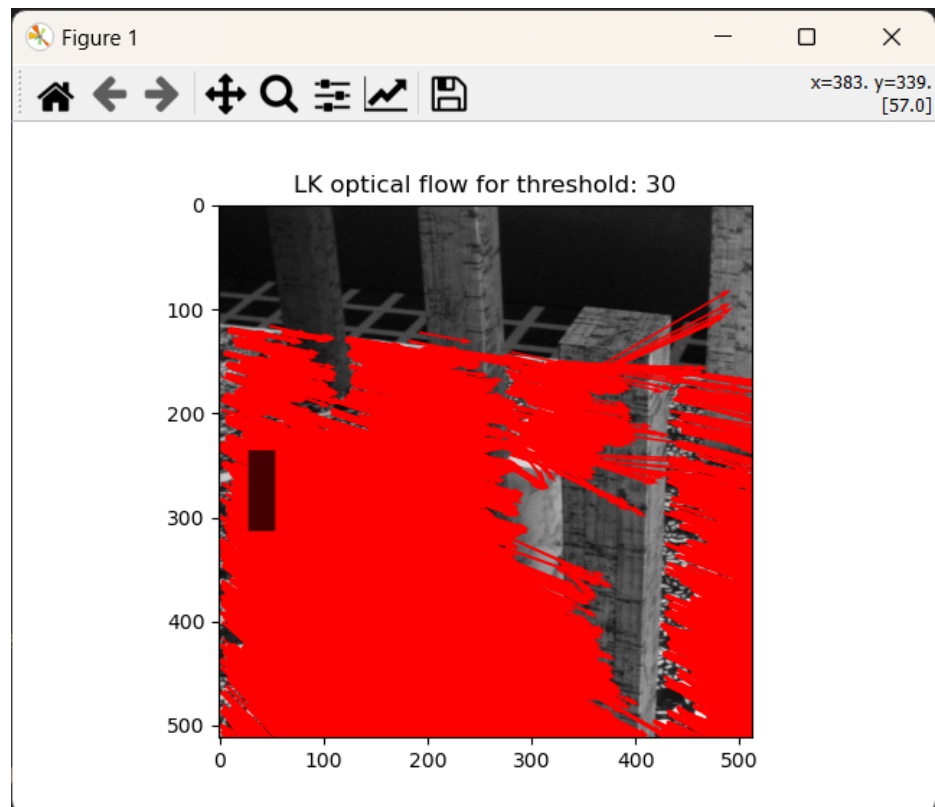


Figure 5: Optical flow lines plot for threshmin = 30

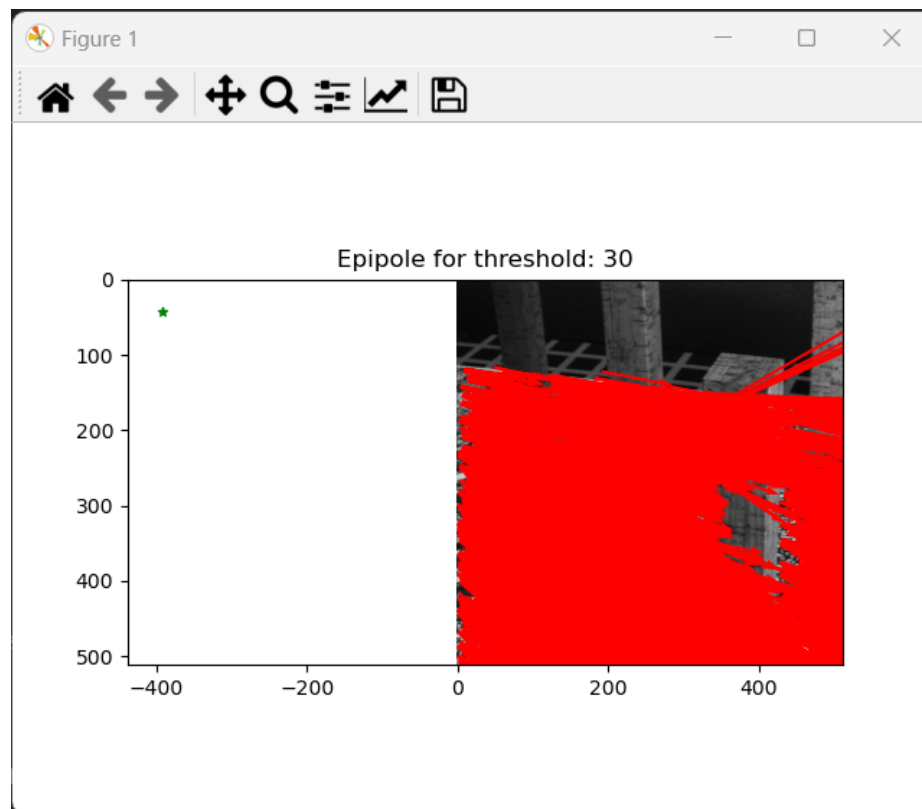


Figure 6: Epipole plot for threshmin = 30

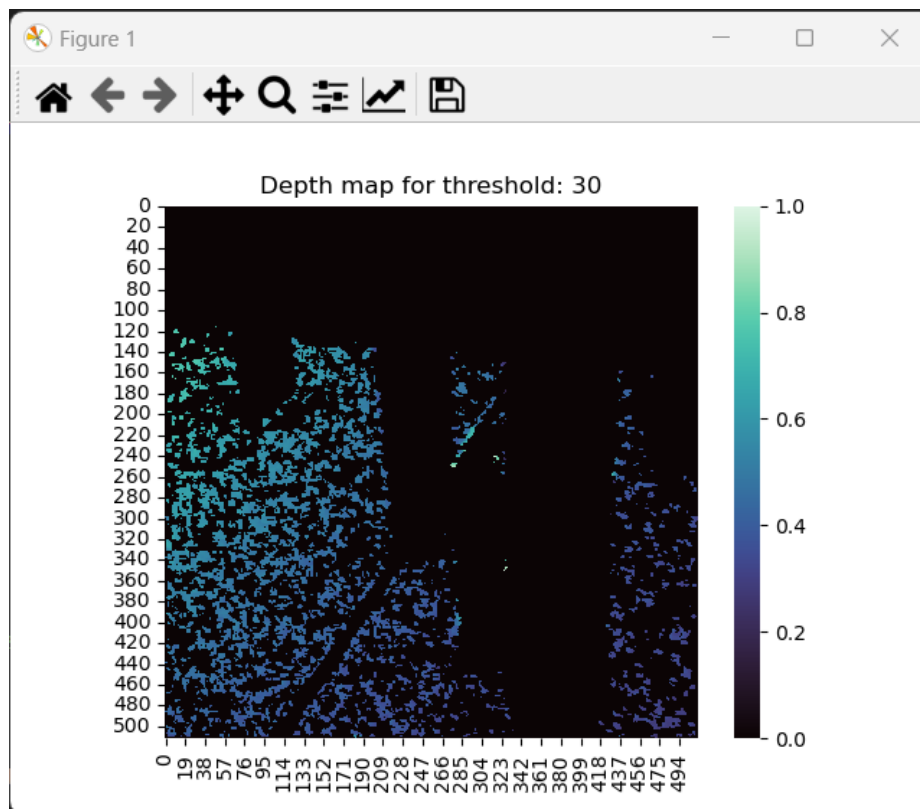


Figure 7: Depth map for threshmin = 30

```
plt.quiver(x[mask], y[mask], u[mask], v[mask], color='red', angles='xy', scale=8, width=0.004)
```

Figure 8: Parameter values used for plt.quiver in plot\_flow function

#### 4 threshmin = 10

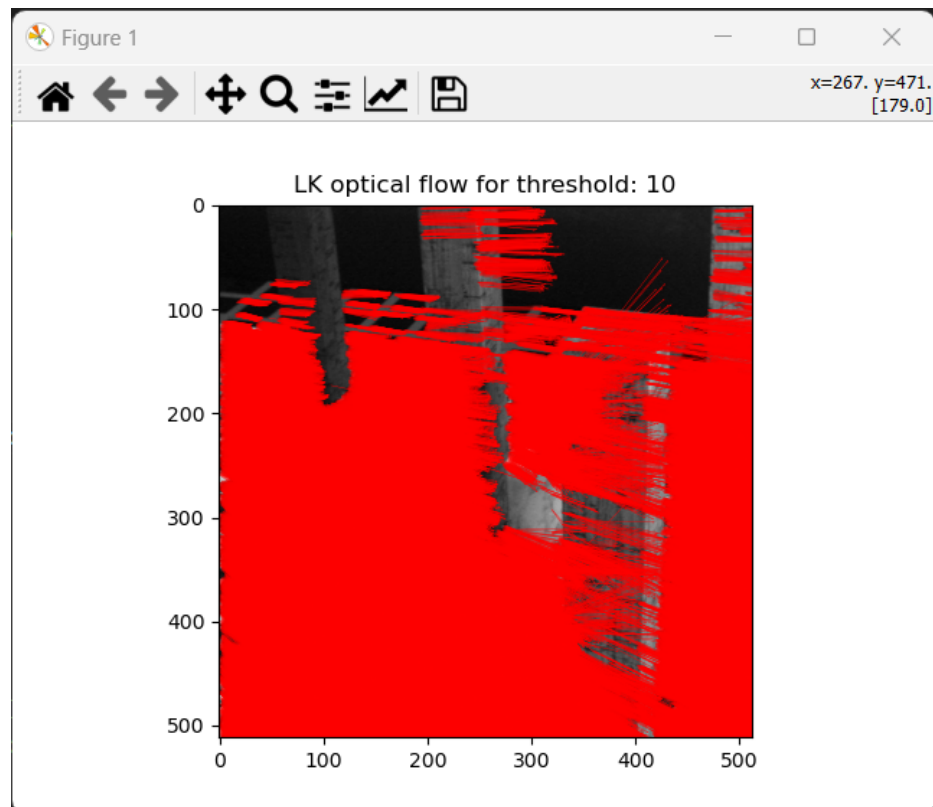


Figure 9: Optical flow lines plot for threshmin = 10



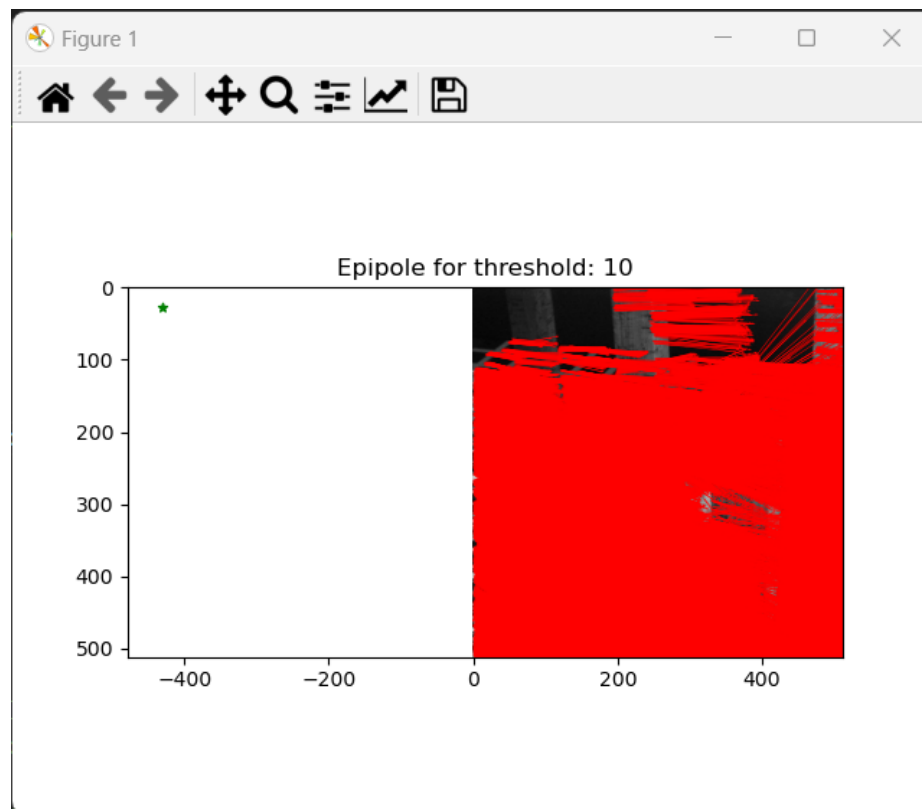
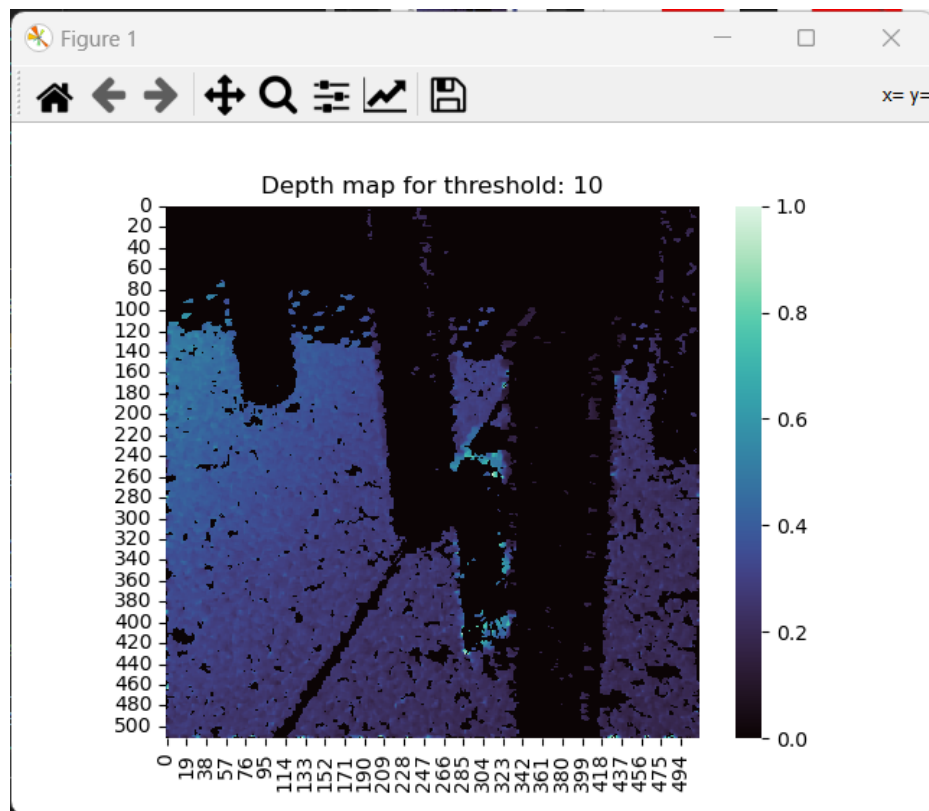


Figure 10: Epipole plot for threshmin = 10



```
plt.quiver(x[mask], y[mask], u[mask], v[mask], color='red', angles='xy', scale=12, width=0.001)
```

Figure 12: Parameter values used for plt.quiver in plot\_flow function

## 5 threshmin = 2

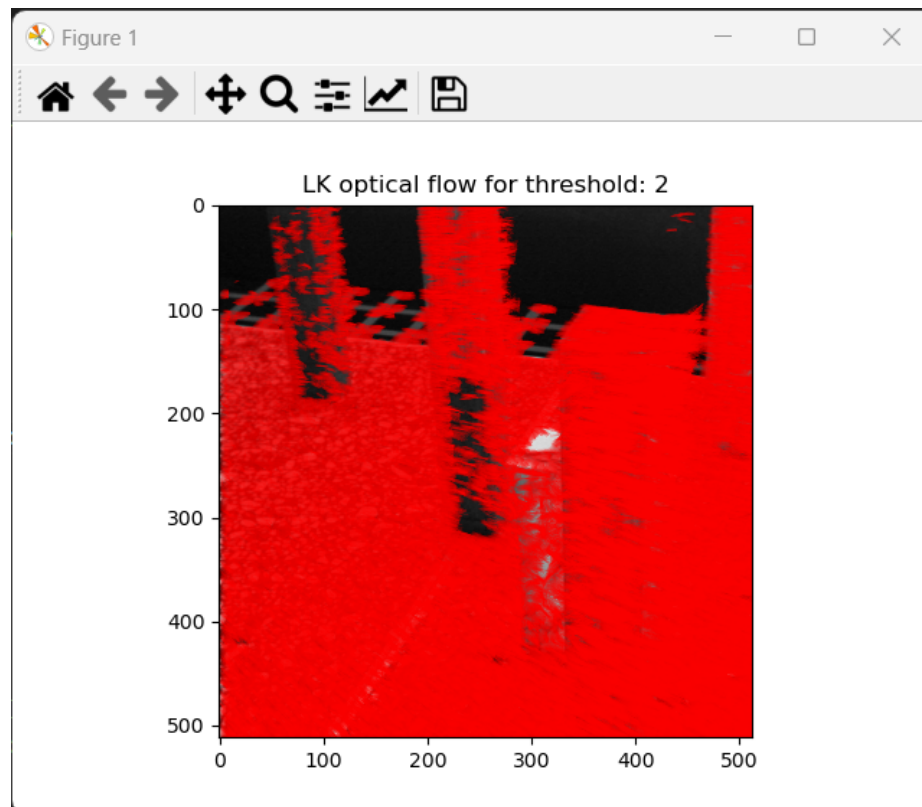


Figure 13: Optical flow lines plot for threshmin = 2

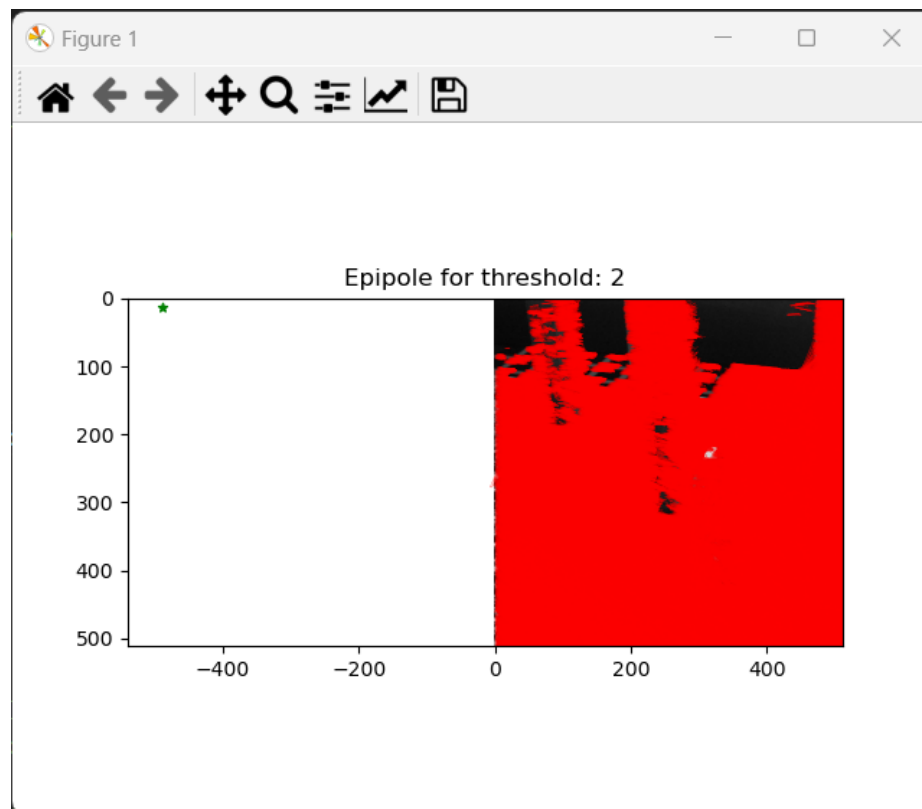


Figure 14: Epipole plot for threshmin = 2

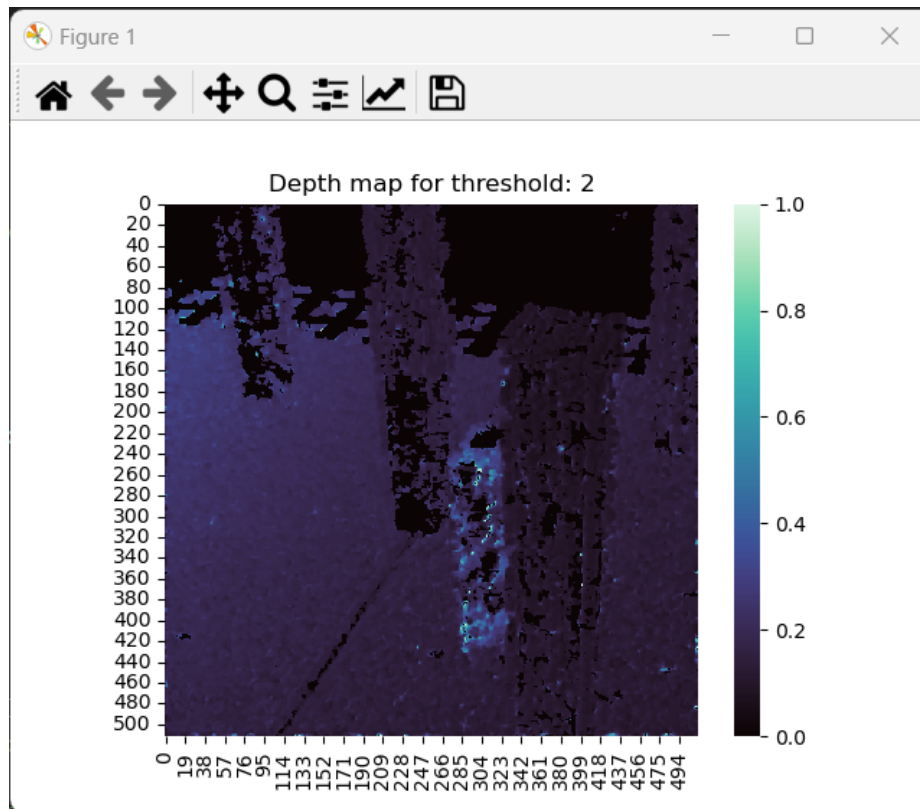


Figure 15: Depth map for threshmin = 2

```
plt.quiver(x[mask], y[mask], u[mask], v[mask], color='red', angles='xy', scale=55, width=0.0004)
```

Figure 16: Parameter values used for plt.quiver in plot\_flow function