How I improved my project:

Thanks to the advices from my reviewer, I spotted my problems and took his suggestions. Here are what I have done to overcome the problem I had:

1. Adjusting perspective transform src dst

Another problem I had was that the bird view is a little bit off. By adjusting the src dst points, I got a better transform.

2. Draw on the area between left lane and right lane

I also drew the area between two lanes in green to indicate what's between the left lane and the right lane

3. Memory system:

I created a numpy array to remember the last 10 left fit polynomials and the right fit polynomials. When a new frame comes in, I first calculate if the left fit polynomial or the right fit polynomial is too off by setting a few thresholds. If it is, we drop this frame, and use the average values from the memory. Otherwise, we pop the oldest polynomial pair out, push the new polynomial pair in, calculate the average, and use the average as the polynomial values we draw on the frame.

4. Using matchShapes function by openCV

Only using the memory system isn't enough to deal with the my lane detection problem when shadows come in. Therefore, I added matchShapes function to compare if the contour drawn on the new frame matches the shape in the previous frame. By setting anther threshold of the value returned by matchShapes function, I found the problem is decently solved.