Vision Documentation

Team 4145

Camera Installation Process

This process should be done in the pit area BEFORE THE FIELD WALKTHROUGH or going to the practice field.

1. Ensure the Raspberry Pi is powered and has the latest SD card installed
2. Ensure the camera LEDs are powered and centered on the Vision camera
3. Ensure the USB hub is powered by the buck boost USB connector
4. Ensure each camera is plugged in the corresponding port on the USB hub
    a. If the cameras are in the wrong order, power off and switch the plugs so that they are in the correct order
    b. Once the cameras are in the correct order, label the plugs/hub
5. Ensure the USB hub is plugged into he Pi
6. Ensure the USB flash drive is plugged directly into the Pi

FRCVision Setup Process

This process should be done in the pit area BEFORE THE FIELD WALKTHROUGH or going to the practice field.

1. Connect the laptop to the robot wifi (4145 or 9145 for practice)
2. Connect to http://frcvision.local
3. Navigate to the “Vision Settings” tab
4. Verify the team number (4145 or 9145 for practice)
5. Verify the cameras are in the correct order
    i. Vision
    ii. Front
6. Verify camera settings are correct for each camera
    a. Vision
        i. Width = 320, Height = 240
        ii. FPS = 5
        iii. Exposure = 5
            1. **Change ONLY if camera feed is too bright**
        iv. Brightness = 50
        v. Contrast = 50
        vi. Saturation = 75
        vii. All automatic settings off
    b. Front
        i. Width = 320, Height = 240
        ii. FPS = 10
        iii. All automatic and default settings on

GRIP Tuning Process Pt 1

This process must be done DURING THE FIELD WALKTHROUGH.

1. Place the robot on the field in front of a target and power on the robot
2. Connect to the robot wifi (4145 or 9145 for practice)
3. Open the competition.grip file in GRIP
    a. Documents/GitHub/2019Vision/resources/competition.grip
4. **Tune the HSV threshold until the target shows clean contours**
    a. Tune Hue first
    b. If there is still no clean contour, tune Value next
    c. SAVE THE FILE
5. Still images will be recorded on the USB flash drive
    a. images/DATE/TIME.jpeg
    b. If for any reason the target is not being picked up during a match, use the images in the GRIP pipeline to debug on the laptop

GRIP Tuning Process Pt 2

This process can be done during the field walkthrough OR in the pits if necessary

6. Export the grip file as a python script called “grip.py”
    a. Save the file in the vision folder of the GitHub repository
        i. Documents/GitHub/2019Vision/src/vision/grip.py
7. Connect to http://frcvision.local
    a. Set the SD card to “Writable” using the top button
8. Open WinSCP and connect to the Pi
    a. Hostname = frcvision.local, Port = 22, Username = pi, Password = raspberry
    b. Copy the entire vision folder the “/home/pi/” directory
        i. Documents/GitHub/2019Vision/src/vision
9. Connect to http://frcvision.local
    a. Navigate to the the “Application” tab
    b. Select “Uploaded Python file”
        i. Documents/GitHub/2019Vision/FRCVision/src/main.py
        ii. Hit save
    c. Set the SD card to “Readable” using the top button
    d. Reboot and verify that the script connects at startup
10. Verify the target is being identified
    a. http://frcvision.local:1181/stream.mjpg
11. Verify the Vision Network Table is being updated
    a. SmartDashboard/Vision/angleOffset

Vision PID Tuning Process

1. Place the robot in front of a vision target
2. Verify that the target is being identified
    a. Remember that the practice field may not have the same lighting as the real field
    b. When the target is identified, hold button 7 to enable gyro PID
    c. Test from a variety of starting angles
        i. I.e. close to target, far from target, etc.

- **If robot is NOT TURNING with enough power to reach target when close, increase MIN_OUTPUT in increments of 0.01**

  subsystems/vision/AnglePIDOutput.java:
  private static final double MIN_OUTPUT = 0.31;

- **If robot is CHATTERING when target is close, increase THRESHOLD in increments of 0.01**

  subsystems/vision/AnglePIDOutput.java:
  private static final double THRESHOLD = 0.13;

- **If robot is CUTTING POWER to motors too early and not reaching target, decrease THRESHOLD in increments of 0.01**

  subsystems/vision/AnglePIDOutput.java:
  private static final double THRESHOLD = 0.13;

- **If robot is CONSISTENTLY OVERSHOOTING the target, increase ANGLE_KD in increments of 0.01 (don't go over 0.1)**
    - You may also need to increase ANGLE_KP after
        - Never increase KP and KD in the same test/deploy
        - Increase in increments of 0.01 (don't go over 0.1)

  Constants.java:
  public static final double ANGLE_KP = 0.04;
  public static final double ANGLE_KD = 0;