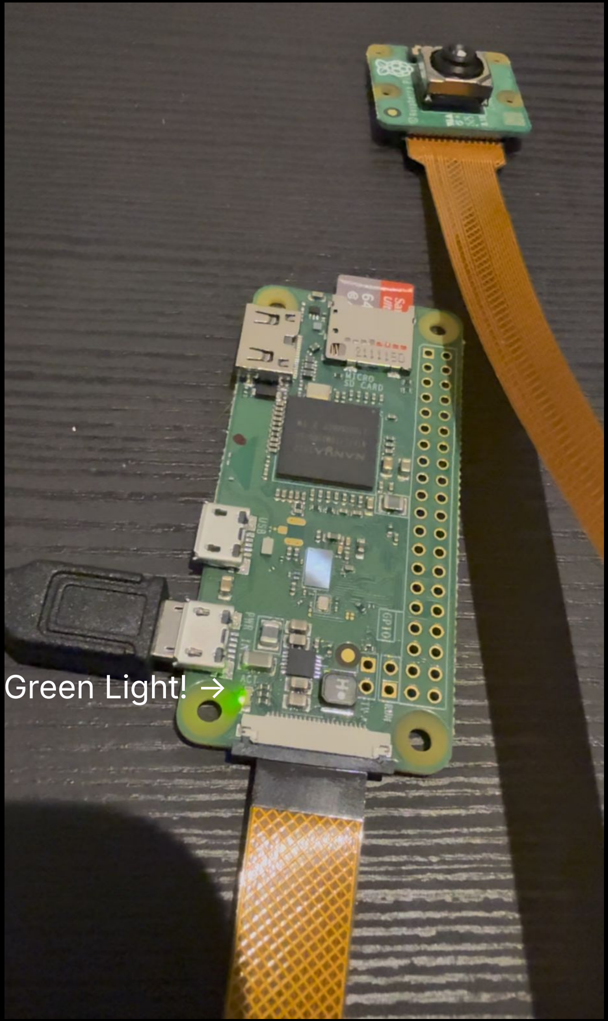
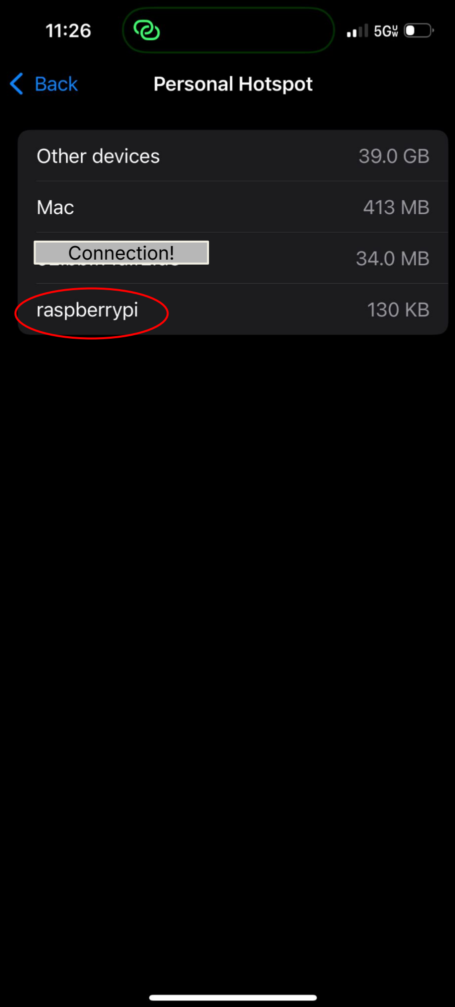
Week 1 Progress

Raspberry Pi Swap Out Camera Cable



Green Light! →

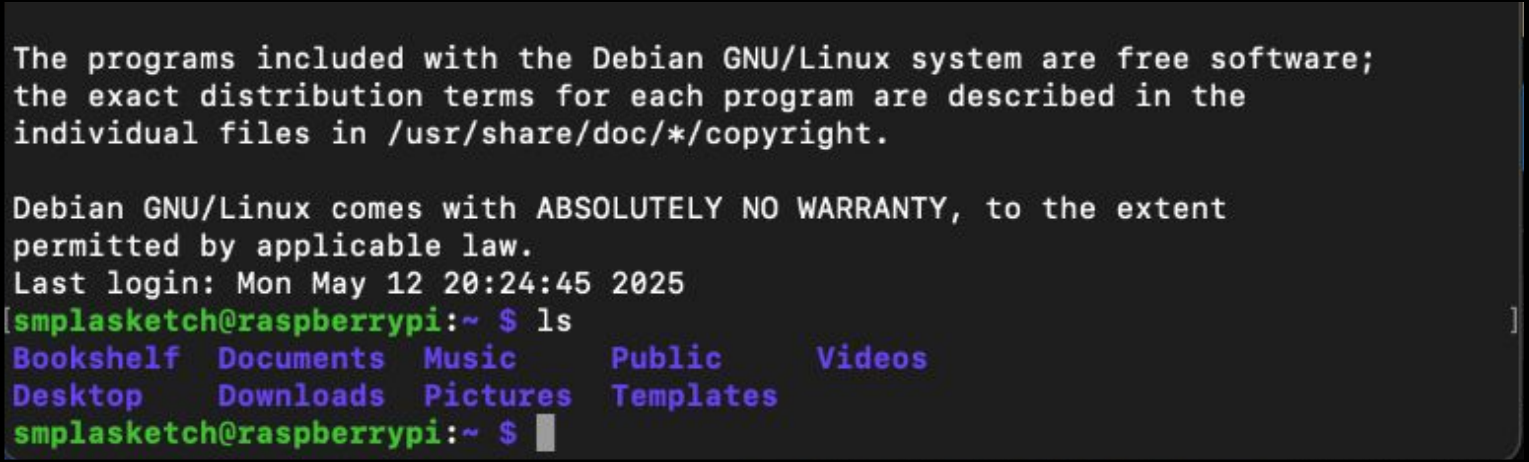
Raspberry Pi Headless Boot



Personal Hotspot

Other devices	39.0 GB
Mac	413 MB
Connection! [redacted]	34.0 MB
raspberrypi	130 KB

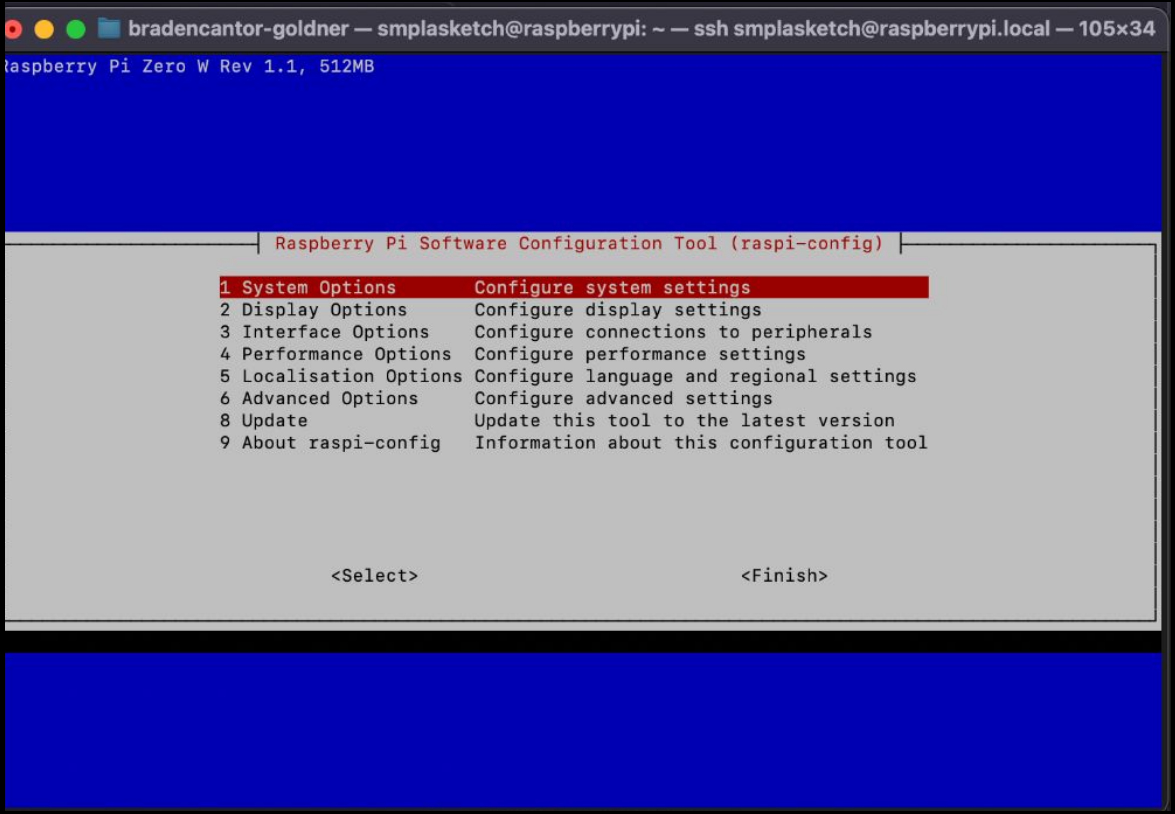
Raspberry Pi Headless Boot



```
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon May 12 20:24:45 2025
smplasketch@raspberrypi:~ $ ls
Bookshelf  Documents  Music     Public     Videos
Desktop    Downloads  Pictures  Templates
smplasketch@raspberrypi:~ $
```

Raspberry Pi Headless Boot



bradencantor-goldner — smplasketch@raspberrypi: ~ — ssh smplasketch@raspberrypi.local — 105×34

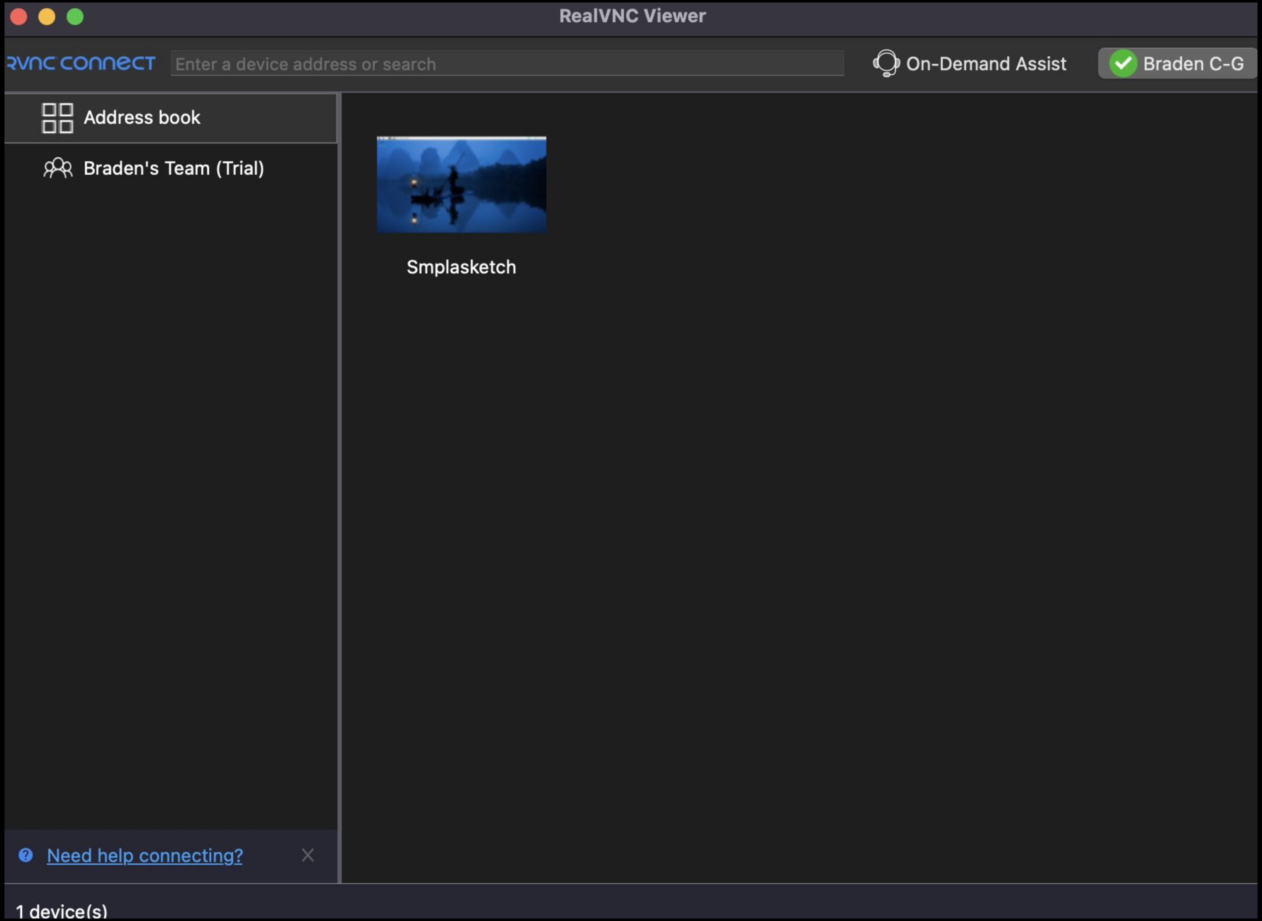
```
aspberry Pi Zero W Rev 1.1, 512MB

           Raspberry Pi Software Configuration Tool (raspi-config)

 1 System Options       Configure system settings
 2 Display Options      Configure display settings
 3 Interface Options    Configure connections to peripherals
 4 Performance Options  Configure performance settings
 5 Localisation Options Configure language and regional settings
 6 Advanced Options     Configure advanced settings
 8 Update               Update this tool to the latest version
 9 About raspi-config   Information about this configuration tool

              <Select>                      <Finish>
```

Raspberry Pi Headless Boot



Raspberry Pi Headless Boot

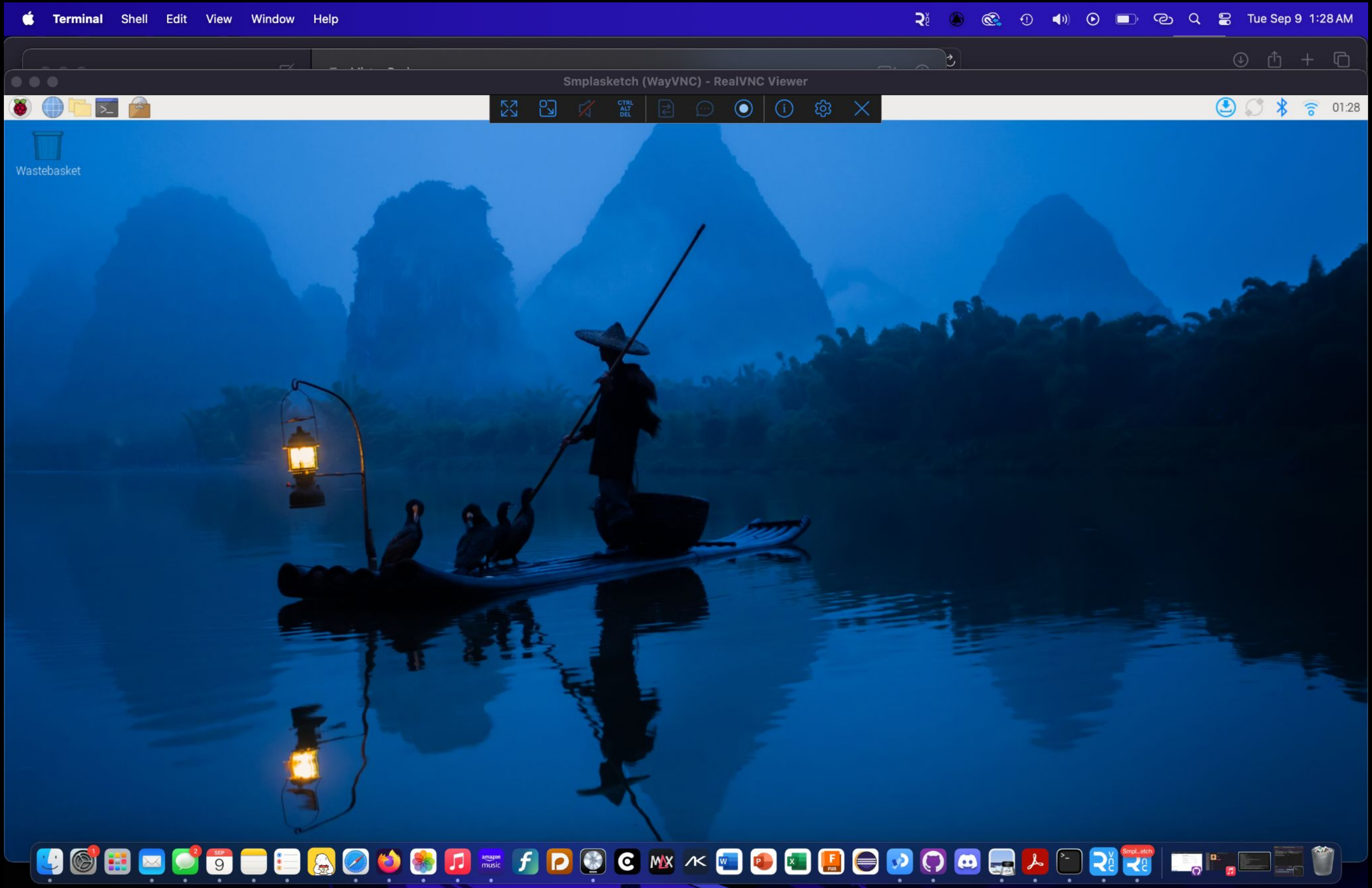
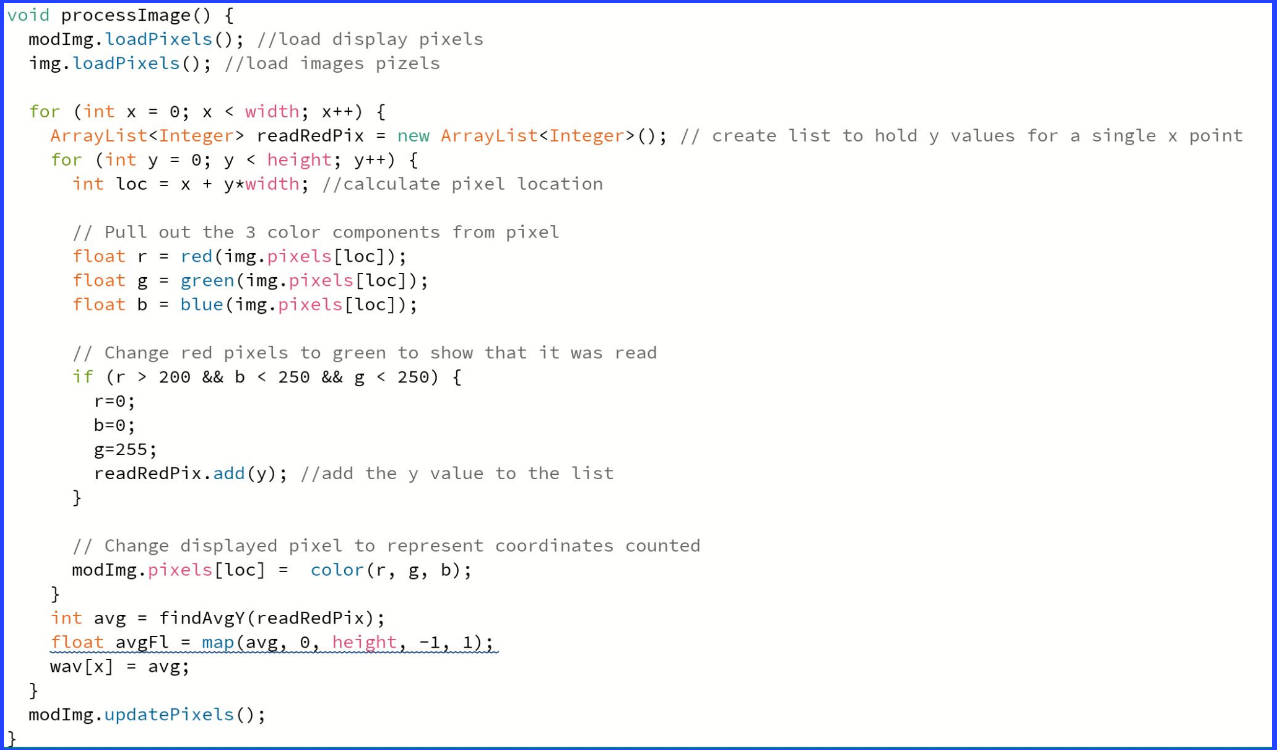


Image Processing Function



```
void processImage() {
  modImg.loadPixels(); //load display pixels
  img.loadPixels(); //load images pizels

  for (int x = 0; x < width; x++) {
    ArrayList<Integer> readRedPix = new ArrayList<Integer>(); // create list to hold y values for a single x point
    for (int y = 0; y < height; y++) {
      int loc = x + y*width; //calculate pixel location

      // Pull out the 3 color components from pixel
      float r = red(img.pixels[loc]);
      float g = green(img.pixels[loc]);
      float b = blue(img.pixels[loc]);

      // Change red pixels to green to show that it was read
      if (r > 200 && b < 250 && g < 250) {
        r=0;
        b=0;
        g=255;
        readRedPix.add(y); //add the y value to the list
      }

      // Change displayed pixel to represent coordinates counted
      modImg.pixels[loc] =  color(r, g, b);
    }
    int avg = findAvgY(readRedPix);
    float avgFl = map(avg, 0, height, -1, 1);
    wav[x] = avg;
  }
  modImg.updatePixels();
}
```

Demo