Color Camera

Using the color camera
Setting the color tracking channels
About color tracking
Camera functions
For this activity, you will need the **camera**.

- The camera plugs into one of the USB (type A) ports on the back of the Wallaby.
- **Warning:** Unplugging the camera while it is being accessed can freeze the Wallaby, requiring it to be rebooted.
Setting the color tracking channels

1. Select **Settings**
2. Select **Channels**
3. To specify a **camera configuration**, press the *Add* button.

4. Enter a configuration name, such as **find_green**, then press the *Ent* button.

5. Highlight the new configuration and press the *Edit* button.

---

**Note:** if there is more than one configuration, select one, and press the “Default” button to make it be the one in use!
6. Press the *Add* button to add a channel to the configuration.
7. Select **HSV Blob Tracking**, then *OK* to make this track color.
8. Highlight the channel, then press *Configure* to edit settings.
   - The first channel is 0 by default. You can have up to four: 0, 1, 2, and 3.
9. Place the colored object you want to track in front of the camera and touch the object on the screen.
   • A bounding box (dark blue) will appear around the selected object.

10. Press the Done button.
Verify the color channel is working

1. From the **Home** screen, press *Motors and Sensors* button.
2. Press the *Camera* button.
3. Objects specified by the configuration should have a **bounding box**.
• You can use the **position** of the object in relation to the **center x (column)** of the image to tell if it is to the **left** or **right**.
  • The image is **160 columns wide**, so the **center column (x-value)** is 80.
  • An **x-value** of 80 is straight ahead.
  • An **x-value** between 0 and 79 is to the **left**.
  • An **x-value** between 81 and 159 is to the **right**.
• You can also use the **position** of the object in relation to the **center y (row)** of the image to tell **how far away** it is.

```cpp
get_object_center_x(0, 0);
// The x-value of the tracked object.
// Note: number between 0 and 159.
```

Channel #  
0, 1, 2, ...  
(largest to smallest)
camera_open_black();
// Opens the connection to the black camera.

camera_close();
// Closes the connection to the camera.

camera_update();
// Gets a new picture (image) from the camera and performs color tracking.

get_object_count(channel #)
// The number of objects being tracked on the specified color channel.

get_object_center_x(channel #, object #)
// The center x (column) coordinate value of the object # on the color channel.

get_object_center_y(channel #, object #)
// The center y (row) coordinate value of the object # on the color channel.