OpenCV part 2

OR

What I've done with OpenCV in the last three years.

Adgenda

- Review what I talked about in 2007
- What has happened to OpenCV since then
- The Camera
- Histograms
- Color finding using thresholds
- Noise reductions
- Next steps

Review 2007

- Loading and showing an Image
- Spliting an image into parts (RGB or Sat, Lum, Hue)
- Thresholding
- Bit manipulations I.E. AND, OR, XOR, NOT.
- A little bit of contors
-

OpenCV since 2007

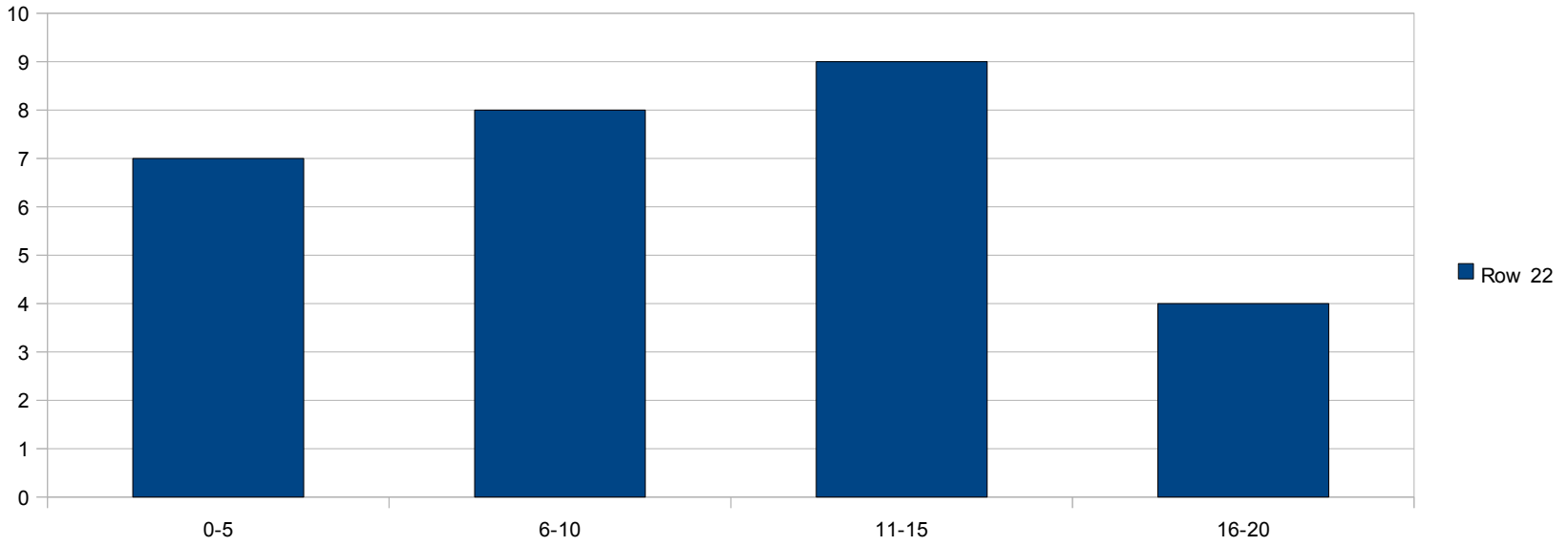
- Wilows Garage "took it over"
- A book was written for it
- Release 2.0
- Release 2.1
- I'm still using 1.0

Camera

- IplImage *theImageFromTheCamera;
- CvCapture* capture;
-
- cvGrabFrame(capture);
- theImageFromTheCamera = cvRetrieveFrame(capture);

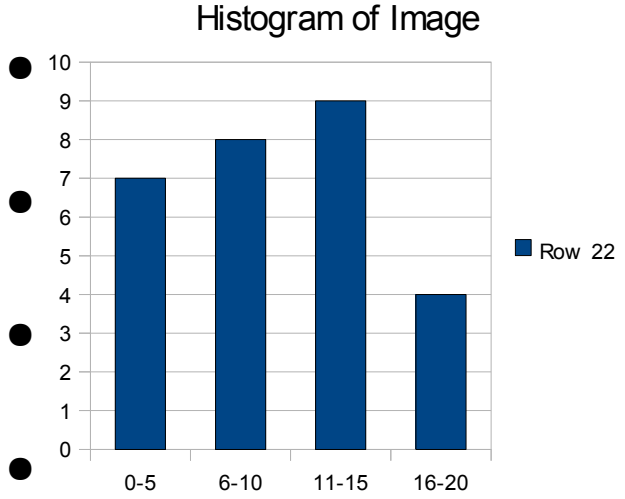
Histogram

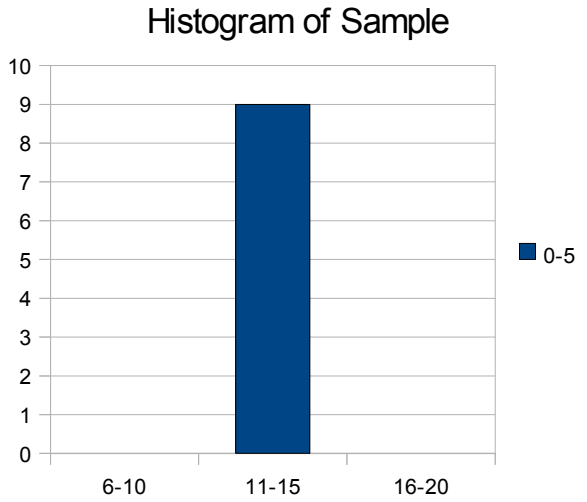
- A chart with a bunch of "buckets"
- 11, 6, 2, 19, 15, 2, 1, 1, 14, 14, 13, 9, 12, 7, 2, 12, 12, 17, 6, 0, 13, 1, 6, 19, 10, 10, 10, 17



Comapairing Histograms

- Find all the items in the image from the sample histogram.





- **11**, 6, 2, 19, **15**, 2, 1, 1, **14, 14, 13**, 9, **12**, 7, 2, **12, 12**, 17, 6, 0, **13**, 1, 6, 19, 10, 10, 10, 17

Histogram's were slow

- When I first started to work on this software I was using a old XP machine.
- The backplane took 1 minute each frame.
- On this machine it's much faster.
- However we are going to pretend it's not that fast.

Color finding using Thresholds

- 11, 6, 2, 19, 15, 2, 1, 1, 14, 14, 13, 9, 12, 7, 2, 12, 12, 17, 6, 0, 13, 1, 6, 19, 10, 10, 10, 17
- I want 11-15.
- Threshold 15 and below:
- **11, 6, 2**, 19, **15, 2, 1, 1, 14, 14, 13, 9, 12, 7, 2, 12, 12,** 17, **6, 0, 13, 1, 6**, 19, **10, 10, 10**, 17
- Threshold 11 and above:
- **11**, 6, 2, **19, 15**, 2, 1, 1, **14, 14, 13**, 9, **12**, 7, 2, **12, 12, 17,** 6, 0, **13,** 1, 6, **19,** 10, 10, 10, **17**
- And the two sets
- **11,** 6, 2, 19, **15,** 2, 1, 1, **14, 14, 13,** 9, **12,** 7, 2, **12, 12,** 17, 6, 0, **13,** 1, 6, 19, **10, 10, 10,** 17

Threshold

- cvThreshold(originalHue,theWorkingMask1,
- maxHue,255,CV_THRESH_BINARY);
- cvThreshold(originalHue,theWorkingMask2,
- minHue,255,CV_THRESH_BINARY);
-
- cvNot(theWorkingMask1,theWorkingMask3);
- cvAnd(theWorkingMask2, theWorkingMask3, destMask);

Now Do this using Hue channel

- Show the demo here, Jim

Getting Rid of Noise

- cvErode(theMask,theRed,0,2);

Next Steps

- Finding the location of objects in the world.
-