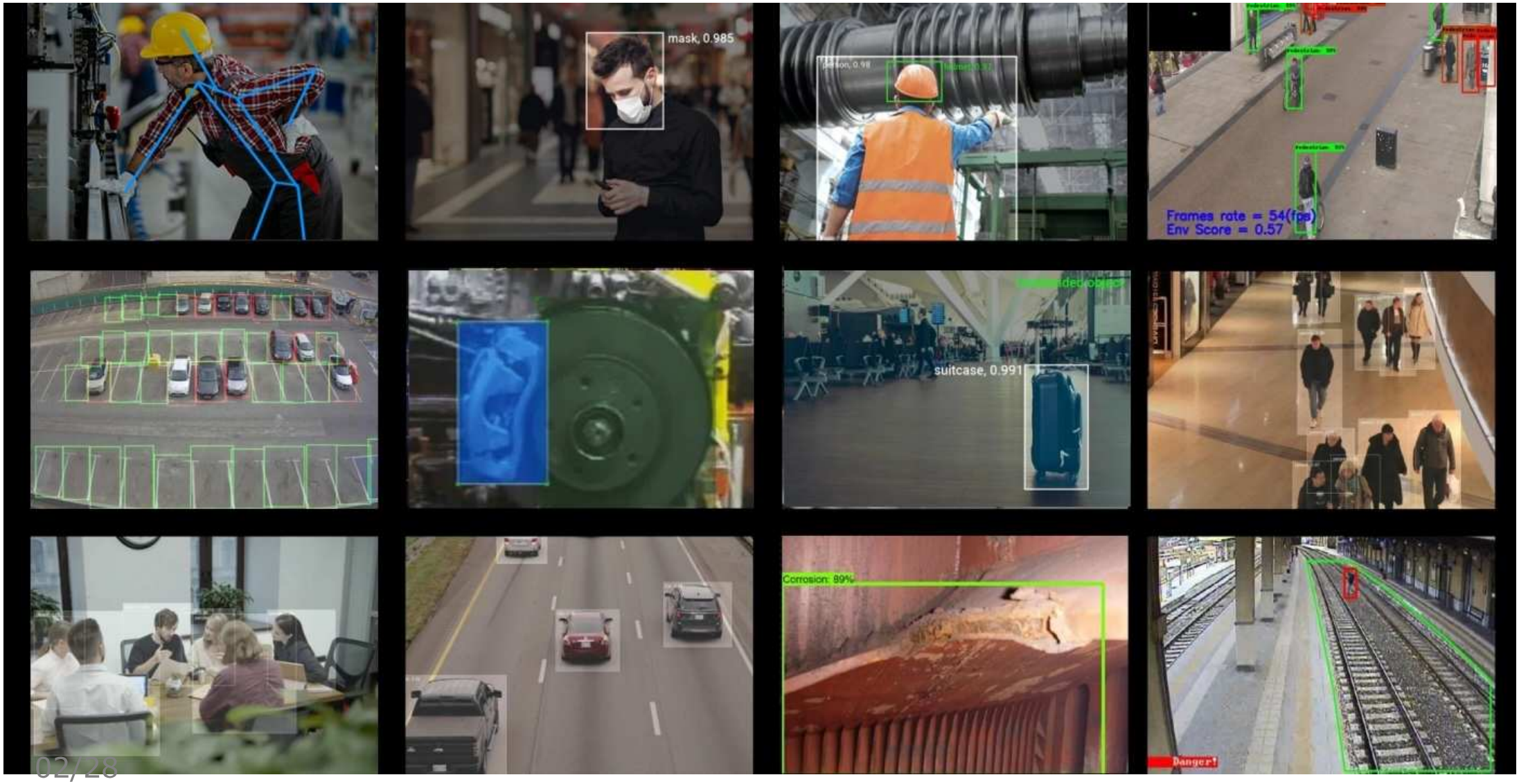
Explorations in Computer Vision

Dr Simon Lock

Example Computer Vision Applications



Different Levels of Comprehension

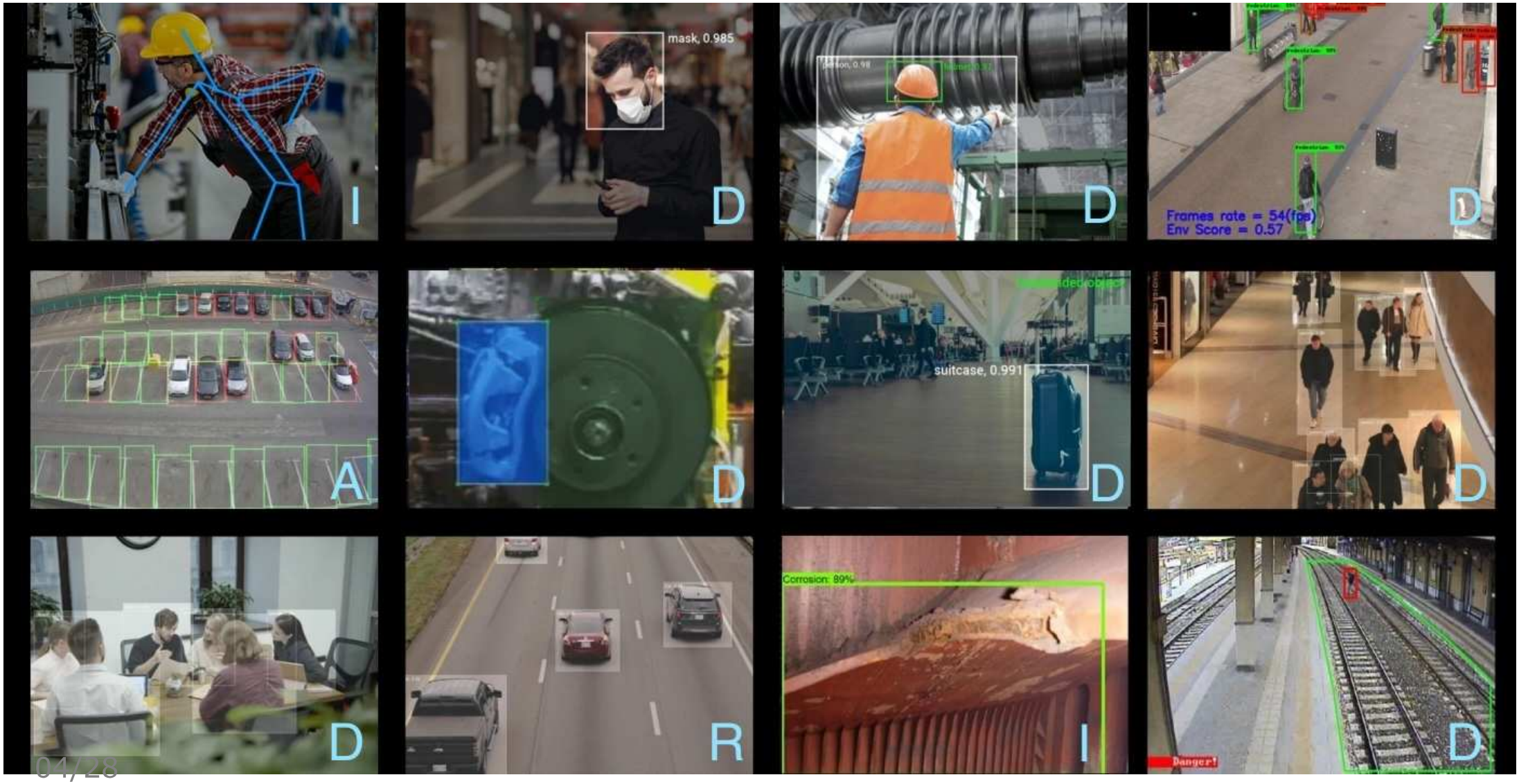
Attention: is there *something* (anything) present ?

Detection: specific *type* of thing (car, person etc) ?

Recognition: which *individual* thing (car, face etc) ?

Interpretation: what is the thing actually *doing* ?

Classification



Our Approach

We *could* try to train a machine learning model
In order to identify objects/structures in an image
But this wouldn't give much insight into the process
(We'd just be training, rather than experimenting)

Instead we will take a much lower-level approach
Focusing on writing code to analyse images/video
Using a number of pixel manipulation techniques...

Pixel Manipulation Techniques

- Searching for specific colours (RGB, Hue, Sat. etc.)
- Frame differencing (change in Brightness, Hue etc.)
- Scanning for specific shapes (person, car, letter etc.)
- Matching relative structures (e.g. face detection)

The Application

We need some form of visual material to process

We'll make use of a recording of a webcam stream

From the "Marine Biological Association" in Plymouth:

3m5!1s0x486c935311b11e0b:0xd40b5a4f7f597d10!8m2!3

It's a nice view, with lots of activity going on !

PlymouthWebcam

Processing Template

To get you started, we've provided a template to:

- Load in an MP4 video file
- Extract a single image ("frame") from the video
- Draw that frame onto the screen
- "Mask off" unwanted areas of the frame
- Define a colour range for analysis
- Analyse remaining "umasked" pixels within the frame

PlymouthCamTemplate

Drawing Pixels onto a Frame

- Before any drawing, call "beginDraw" on the frame
- To mask off areas set "fill" colour to black
- Mask off Rectangle, Triangle or Polygon areas
- To draw individual pixels set "stroke" colour...
- Then draw an individual "point" (pixel) in that colour
- After drawing, be sure to call "endDraw" on Frame

Checking the Colour of Individual Pixels

We can get the colour of a particular pixel using:

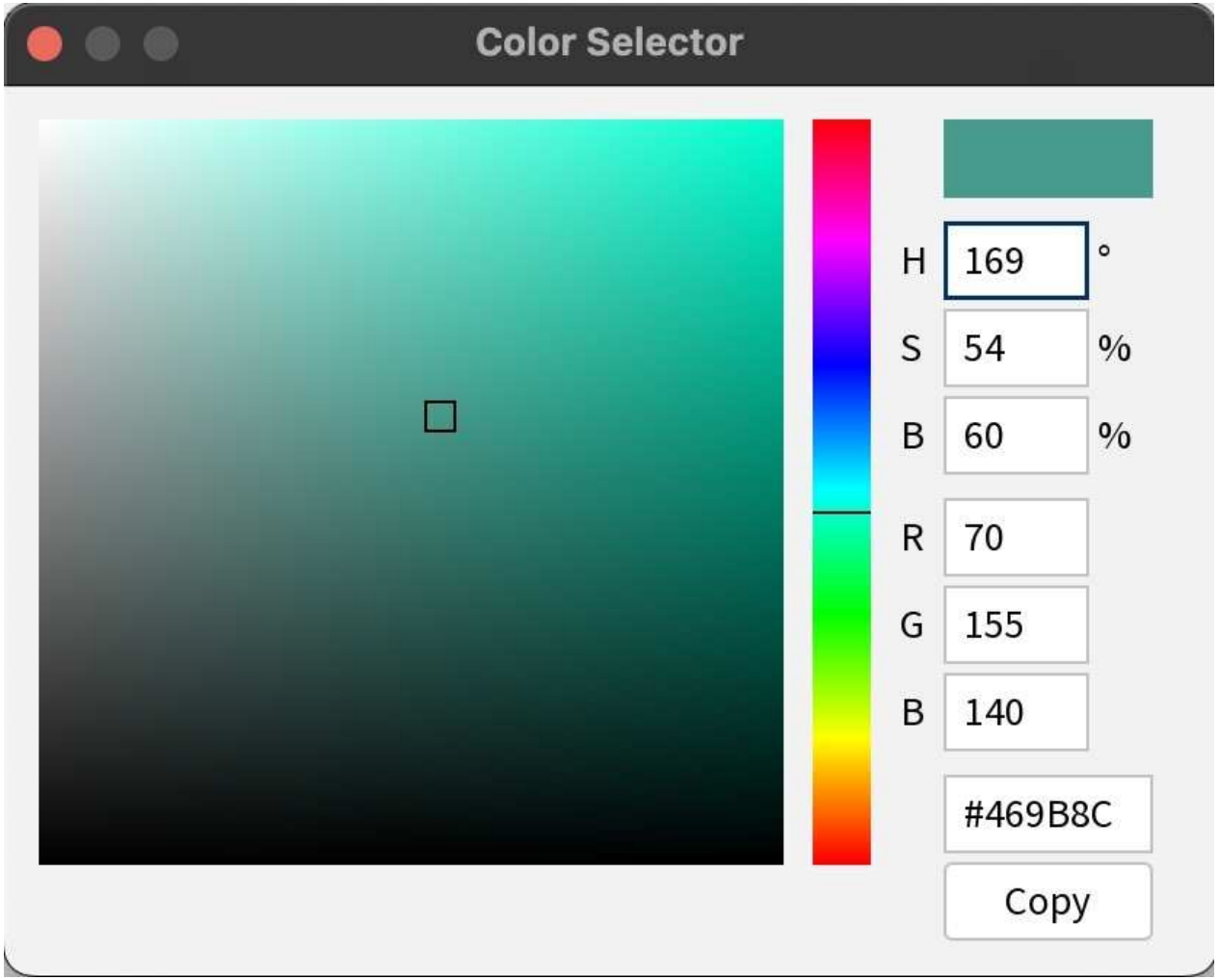
```
int pixelColour = currentFrame.get(x,y);
```

Then extract various properties of the pixel using:

```
int redness = red(pixelColour);
int greenness = green(pixelColour);
int blueness = blue(pixelColour);
int whichColour = hue(pixelColour);
int howBright = brightness(pixelColour);
int howSaturated = saturation(pixelColour);
```

Hue / Saturation / Brightness / RGB

Warning: All values in Processing are in range 0-255



IMPORTANT NOTE

This is a very processor-intensive activity !
280k pixels per frame requires a LOT of calculation

If you do not have a fast laptop
We advise you to use a lab machine !!!
(otherwise it will be painfully slow !)

Download

Download the template code (and these slides) here:

https://tinyurl.com/BristolCamera

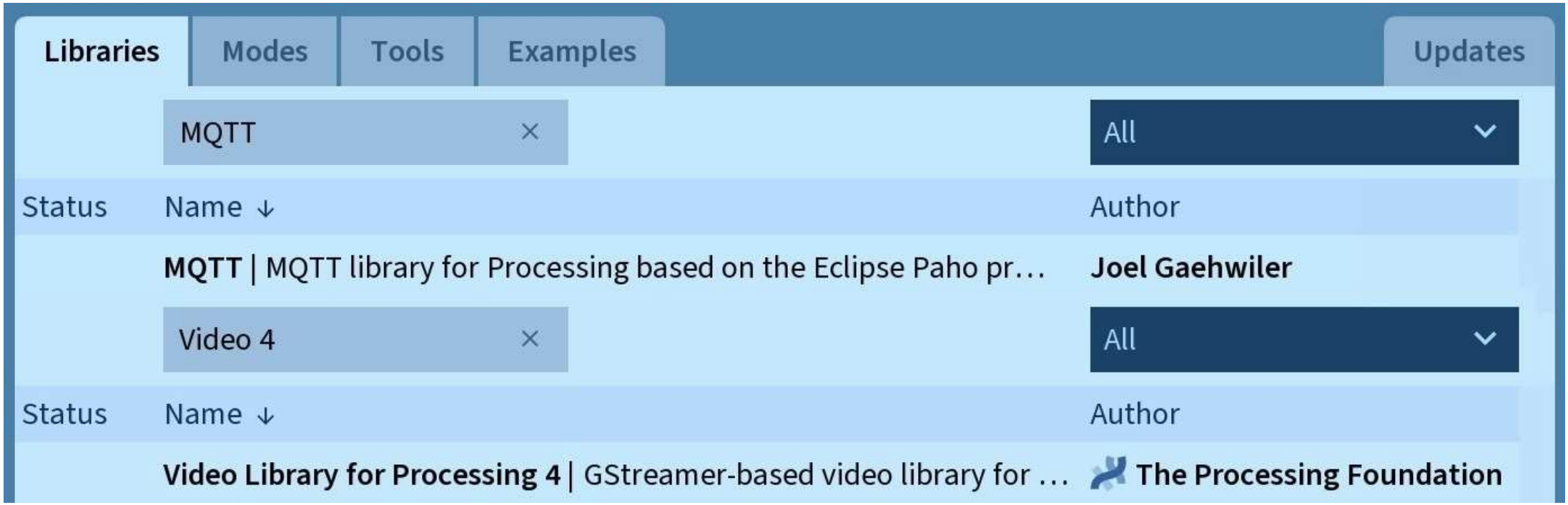
https://tinyurl.com/BristolCamera

Libraries

In order to run the template project

You will need to install a couple of libraries:

`Sketch > Import Library > Manage Libraries`



Your Objective

Select an aspect of the scene to analyse/monitor

You are free to choose any element that you like...

But try to choose something with a clear "purpose"

(Something it might actually be useful to monitor !)

Later, we will be send someone a notification

When a particular situation occurs in the scene

What Will You Choose ?






To Work !