Importing and Analyzing a Video in MATLAB

Recent versions of MATLAB import video files (AVI and some other formats) with the commands `VideoReader` and `read`. To read in part of the file `dpexcerpt.avi`, type:

```
>> obj = VideoReader('dpexcerpt.avi');
>> vid = read(obj, [1 500]);
```

(Note: in MATLAB versions 2008a-2010a, `VideoReader` was called `mmreader`.) This reads the first 500 frames; your computer might not have enough memory to read the entire file. The result `vid` is a 4-dimensional array of integers between 0 and 255; each value in the array represents the intensity of a particular color (red, green, or blue) at a particular pixel in a particular frame of the video. Specifically, `vid(v,h,c,f)` is the intensity of color number `c` (where red is 1, green is 2, blue is 3) at the pixel with vertical coordinate `v` and horizontal coordinate `h` (measured from the top left) in frame number `f`. Thus, the following command selects the 108th frame stored in `vid`:

```
>> frame = vid(:,:,:,108);
```

Each frame (as you can check now with `whos`) is a $480 \times 640 \times 3$ array. You can view the frame by typing:

```
>> image(frame)
```

Spend some time inspecting the values in the array by using the Data Cursor (icon with a dark plus sign) and Zoom tool (icon with a magnifying glass and plus sign) in the Figure Window. (Click on the appropriate icon, then click inside the image.)

To work on quantitatively isolating different colors in the frame, it may be useful to define separate arrays for each color, and convert the intensities from unsigned integers to floating point numers (for simpler arithmetic):

```
>> red = double(frame(:,:,1));
>> green = double(frame(:,:,2));
>> blue = double(frame(:,:,3));
```

To visualize (for example) the array of red intensities, you can type:

```
>> imagesc(red)
>> colormap(gray)
```

Or, to visualize the pixels at which the red intensity exceeds the blue intensity:

```
>> imagesc(red > blue)
```