iqr Tutorial 05-Camera-Basic interfacing

iqr tutorials will provide you with a practical introduction to using the neural simulation software and give you an insight into the principles of connectionist modeling. They are intended to be complementary to the detailed operation manual.
The home page of iqr is at iqr.sourceforge.net. Up to date information, documentation and tips and tricks ca be found in the iqr wiki (sourceforge.net/p/iqr/wiki/Home/).
The repository of iqr packages is here: sourceforge.net/projects/iqr/files/. iqr is open source software. You can browser the entire source code of iqr here: http://sourceforge.net/p/iqr/code/HEAD/tree/. Please contribute to iqr by reporting bug (sourceforge.net/p/iqr/bugs/) and requesting features (sourceforge.net/p/iqr/feature-requests/).

Aims

- Use data from an external source (a video camera) to influence processing in a simulation.
- Apply different filters to the video input

Setup

In this tutorial we use a camera and an iqr module that reads from the camera. We look at a camera as a matrix of sensors, where each pixel gives the reading of a particular amount of light and light color. The iqr “video module” should support most USB and laptop cameras. Additionally, we can read video streams using the “mjpegVideo module”. Please refer to the iqr wiki (http://sourceforge.net/p/iqr/wiki/Home/) for the most up to date information on supported devices, tips and tricks, etc.

Building the System

Start with a copy of the simulation you used in Tutorial 3. Rename your existing process to “Processing” add a new process called "Video". Create a linear threshold cell group called "Mono" and make it 40x30 neurons. Remember that large cell groups are very computationally expensive, so we subsample the image to reduce the processing load on the system. Go to the process parameters and set the module type to match the type of video camera you have (ask if you are not sure) and set also the different module parameters available for your camera. Click on the “Module to Groups” tab and make sure that the video output is going to your new "Mono" cell group. The default image parameters should be OK, but if you have any problems getting an image you will need to come back here to change the parameters.

Now connect the Mono cell group in “Video” to the Linear Threshold group “Processing”, making this groups of the same dimensions. To do this, use the window split buttons in the top-right corner of the window so that you can view the cell groups of two processes at the same time. Make sure the synapse properties are Pattern-Mapped (all-all), ArbRect (height = width = 1), FunUniform (delay = 0) and Fixed weight (weight = 1.0). Save the simulation when you have finished.

Create an additional group of Linear Threshold neurons of the same dimensions named “Filtered” and create an excitatory connection from Linear Threshold to it.

Exercise

Start the simulation. You should see a small window appear containing the video image from the camera. If not, go back and check the camera (power on, cable connected to computer) and the module settings. For many cameras you can use the Linux utility xawtv to check the camera outside the environment.

- Move the camera around to generate different inputs. What effect the camera is having on the system.
- If the size of the "Mono" cell group is different to the "RandomSpike" cell group that it is connected to. How is this handled by ? Was this a problem? or why not?
- Play with the connection to "Filtered" to achieve the following image processing:

Connection	Blur	Edge extraction
Pattern		
Arborization		
Attenuation		
Synapse		

- And now modify the neuron parameters of "Filtered" to achieve a segmentation of bright image spots (light sources). What are those parameters?

Group name	LinTh
Size (cells)	
Cell Type	Linear threshold
Excitatory gain	
Probability	
Membrane persistence	
Threshold	
Membrane potential reset	
Spike amplitude	
Sigmoid midpoint	
Sigmoid slope	

- Choose now a color module for the video camera. Now you have 3 input groups, Red, Green and Blue. Is it possible to segment a specific color with this information? How can you do it?

Save the simulation again