Visualization Manual

Philipp Hannibal

March 5, 2015

Manual

This manual explains the usage of the visualization program at hand which is designed and tested for Linux Ubuntu 14.04.

Installation

The program is provided in a developer version. Each user is free to change the code at his own will. Yet it is necessary to install Qt-libraries by executing

$ sudo apt-get install qt5-default
$ sudo apt-get install qtcreator

in the shell (indicated by $). The use of qtcreator is recommended to examine the source code. When Qt is set up, download the program files from []. Those should contain a *main.cpp* and a header- (.h) and a source code- (.cpp) file for *QVisualization*, *VisualizationView*, *TopologyCalculator* and *ControlPanel* and a project (.pro) file. If the .pro-file is lost, type

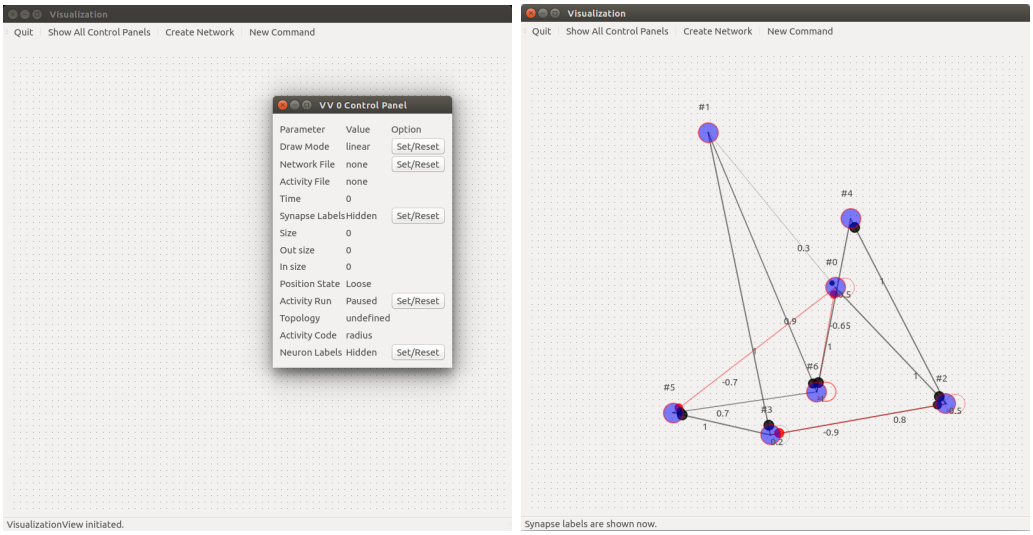
$ qmake -project

in the shell with the project folder as working directory. As soon as it is ensured that the created .pro-file includes

"QT += core gui
greaterThan(QT_MAJOR_VERSION, 4): QT += widgets"

(add, if missing), use the QtCreator interface to open, make and run the project from QtCreator using the green triangular. Without QtCreator execute:

$ qmake && make
$ program directory/executable's name



(a) (b)

Figure 1: Screenshots of the visualization program. (a) Default graphical user interface. (b) Example of a 7x7-matrix in random mode with neuron and synapse labels.

within that directory. The second line should start the visualization. The executable's name is the project name which is the program directory's name by default.

Interface

After starting the visualization two windows pop up. The main window displays a toolbar and a grid where a network is drawn and the control panel for the first network (the network's index starts at 0). A screenshot of this initial default state is shown in fig. 1a.

The toolbar and control panel buttons are explained in tab. 1.

An example of an arbitrarily composed 7x7-matrix is printed in fig. 1b.

Command Based Control

The whole program is designed to be command controlled as well. These commands can either be typed using the "New Command" button or using the "-pipe" option when starting the program. For the latter select a program writing to stdout and pipe its output using the shell piping command "|":

$ other program | visualization path/executable -pipe

Button	Location	Functionality
Quit	toolbar	Quits the visualization and closes all control panels.
Show All Control Panels	toolbar	Arranges the control panels of all networks on the left side of the screen. Closed ones will be shown again.
Create Network	toolbar	Enables the display of multiple networks. An additional control panel pops up. The network images are rearranged.
New Command	toolbar	Interaction window pops up to enter a command listed in tab. 2.
Draw Mode Set/Reset	control panel	Directly type the mode. Chose from linear, map, random, top, ff
Network File Set/Reset	control panel	File browser opens up to select a matrix file containing synaptic data. Will be displayed directly after confirmation.
Synapse Labels Set/Reset	control panel	Toggles synaptic labels on/off Not available for map or linear mode.
Activity Run Set/Reset	control panel	Starts or pauses looping over read activities.
Neuron Labels Set/Reset	control panel	Toggles neuron labels on/off. Indication with #.

Table 1: Overview on all available buttons.

For example, with the program "cat" preinstalled in Linux the data of a file is printed into stdout, thus allowing to prepare a setup file storing information about chosen visualization options and matrix. Separated by a space-character all commands consist of the syntax: target-index command-shortcut command-option. The target index I is an integer and either -1 for global commands (processed by QVisualization) or non negative as index of the network that is addressed (processed by VisualizationView). The command shortcut consists of four capital letters representing the written-out command. Depending on the chosen command options enable specifications of the command. A whole list with possible commands and their syntax is given in tab. 2.

It should be pointed out that the MODE, AMOD, RNTW data and the ACTL commands are the most frequently used ones. For an example how to create a picture an examplary command series is given in tab. 3. Generally make sure that the command lines do not end with a space. The RNTW data command is not suitable for usage without piping. On deletion of a network the networks' target index shifts one down for all networks with a higher index than the deleted one. The network's index I is counted primarily from left to right and secondary from top to bottom.

Mode Explanation

In the random mode pseudo-random values for the x- and y-positions of the perceptrons are calculated and they are settled there as circles. The random number calculation is started with a fixed seed so that each arrangement can be reproduced. The connections are drawn as straight lines.

The linear mode displays all perceptrons in a single horizontal row. In this mode the connections are drawn as curved arcs. This prevents them from painting over each other. Those arcs have to be read counterclockwise (moving along the arc in from-to direction) for interpretation. Labels are relinquished and activities are not encoded.

The map mode displays all perceptrons as neighbouring rectangles without any representation of the weights. This is intended for studies on the activity behavior of bigger networks. It always comes along with an unlabelled legend of the activity: top is $a = 1$ is red and bottom is $a = 0$ is blue.

The ff mode - with ff for feedforward - displays the network in vertically oriented layers. The layers have from the left to the right the sizes l_i as given with the MODE ff[l_1 l_2 ...l_k] command (cf. tab. 2). If the network size n exceeds the distributed perceptrons $\sum_{i=1}^{k} l_i$ all remaining perceptrons are thrown in an additional last layer. If one attempts to distribute more

I	Shortcut	Options, Descriptions and Comment
-1	QUIT	-, Quits the visualization
-1	CRTN	-, Creates network. Same as "Create Network" button
-1	DELN	I, Deletes the network I. Always one network remains.
≥ 0	RPNT	-, Forces refreshing (repainting) of the network image.
≥ 0	MODE	random||linear||map||top||(ff[l_1 l_2 .. l_k]), Draws the network in the chosen mode, $l_1..l_k$ are integers Indicating the layer sizes for a feedforward (ff) depiction.
≥ 0	AMOD	(radius||color||opacity) R, Sets the activity modulation coding and R is the maximum perceptron radius in pixel
≥ 0	RNTW	(file file name)||(data n; n-times: space seperated matrix line;), Reads the network data. The "data" option is designed for piping. Network size defined by n.
≥ 0	ACTL	[a_1 a_2 .. a_k], Reads the activity for the first k perceptrons and directly displays it. If $k < n$ all other activites are set zero. If $k > n$ additional activities will be ignored. Ensure $0 < a_i < 1$ for $i = 1..k$.
≥ 0	RUNA	-, Starts (or stops) the activity loop. All read activity lines will be displayed in series. The current time step is printed in the control panel under "Time".
≥ 0	BRKA	-, Stops the activity loop and resets the "Time"
≥ 0	SLBL	[on||off], Switches the synaptic labels on or off
≥ 0	NLBL	[on||off], Switches the perceptron labels on or off
≥ 0	RACT	filename, Reads the lines of the filename file as activity lines. It is recommended to use the ACTL command for that.
≥ 0	RSET	-, Resets some intern boolean values. No direct effect.

Table 2: Possible commands and their syntax. If $I \geq 0$ a single network is addressed. All " " represent an essential space, "||" represents an "or", "-" indicates no further options, ";" represents the end of a line. Everything in ()s belongs together, everything in []s is falcultative and the brackets do not belong to the command.

Command	Commentary
0 MODE random	Switches to random mode.
0 RNTW 7	The following 7 lines will be interpreted as weight matrix
-0.5 0.3 0 0 0 0 -0.65	Incoming weights of perceptron 0
0 0 0 0 0 0 0	Incoming weights of perceptron 1
1 0 -0.5 0.8 0 0 0	Incoming weights of perceptron 2
0 1 -0.9 0.2 0 0 0	Incoming weights of perceptron 3
0 0 1 0 0 0 0	Incoming weights of perceptron 4
-0.7 0 0 1 0 0 0.7	Incoming weights of perceptron 5
0 0.9 0 0 1 0 -1	Incoming weights of perceptron 6.
0 AMOD radius 30	Perceptrons will be blue circles of 30 pixel radius
0 SLBL on	Weight values are printed at $\frac{2}{3}$ of the arc.
0 NLBL on	Perceptrons will be labelled from #0 to #6

Table 3: Commented example command series.

perceptrons than weight information is given for, $\sum_{i=1}^{k} l_i > n$, exceeding layers will be ignored.

The top mode takes the random mode as springboard and calculates for several integration steps the forces as discussed in section 2.1.3. Applied Algorithms and changes the x- and y-positions accordingly. In other aspects it doubles the random mode.

An example screenshot of a multi-network depiction of one network in four different modes can be seen in fig. 2.

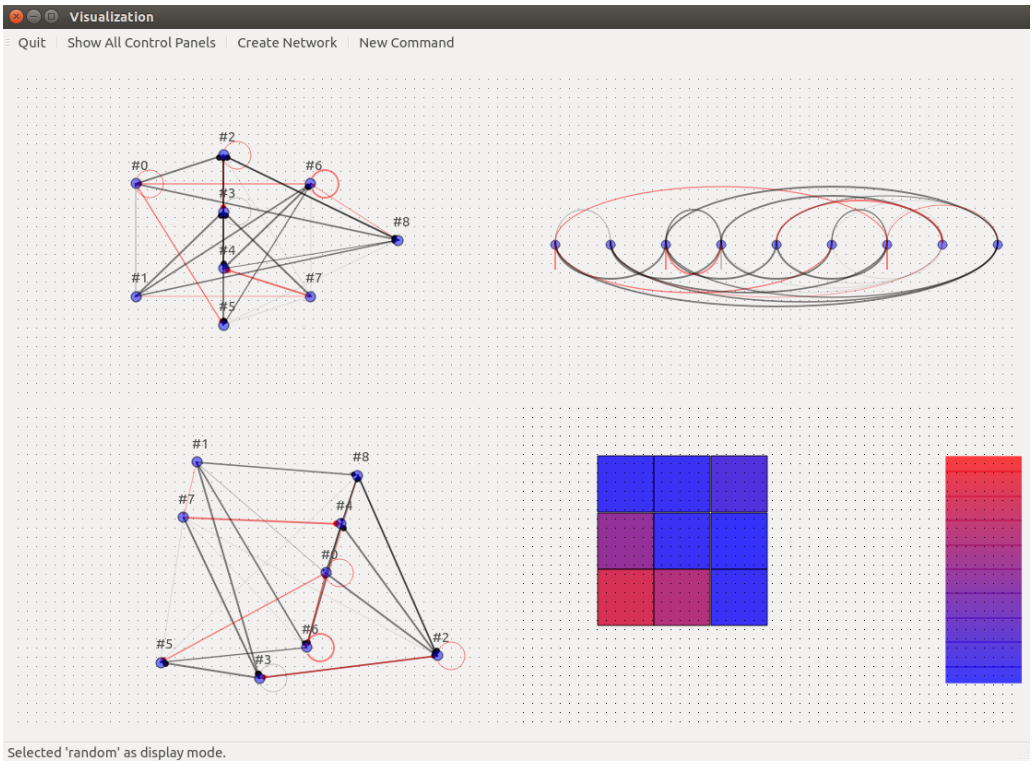


Figure 2: Screenshot of the visualization set up with an arbitrarily valued 9x9-matrix in four different modes: ff 2 4 2 1, linear, random, map. For the map display a single activity line was given.