**The Divisive-Normalization Model of V1 Neurons: A Comprehensive Comparison of Physiological Data and Model Predictions**

Manual for software of the Divisive Normalization Model

Tadamasa Sawada1 and Alexander A. Petrov2

1 School of Psychology, National Research University Higher School of Economics

2 Department of Psychology, Ohio State University

This document explains how to use our Matlab software of the Divisive Normalization Model (DNM) following three sample scripts: Manual\_Sample\_1, 2, 3.m.

# Manual\_Sample\_1.m

The software takes bitmap image data as a visual stimulus and outputs set of responses of the DNM to the stimulus. In this sample script, the software is given a single visual stimulus loaded from an image file (SampleImages/SampleImg\_1.bmp) and responses of model neurons to the stimulus are plotted.

## 1) Load an image file (ll.8-16)

The image file is loaded and converted to a 2D matrix. For this script, the size of the image has to be 64 x 64 pixels. If the image has multiple channels, values of the matrix are averaged among the channels (l. 13). Then, the values in the matrix are normalized between 0 and 1 (l. 14).

The stimulus (SampleImg\_1.bmp) is a grating with its orientation 15° and spatial-frequency 1 oct (2 cpd). Its phase is 0° at the center of the stimulus.

## 2) Set-up a "model" structure M with the standard parameter set (ll.17-20)

The software generates image filters of the DNM based on a given parameter set. All the information of the DNM are stored in a structure M (l.31). The standard parameter set (DMPL\_defaultSpecs) is used in this script for simplicity. We will consider modification of the parameters later (Manual\_Sample\_3.m).

## 3) Apply the DNM to the stimulus (ll.21-30)

The stimulus and the model structure M are given as inputs of a function DMPL\_EarlyVis\_FiringRate (l. 35). The output (resp\_DivNorm) of the function is a four-dimensional matrix with 300 elements that are responses (firing rates) of the model neurons with different tunings to the stimulus:

12 orientations (0, 15,30, 45, 60, 75, 90, 105, 120, 135, 160, 165°)

5 spatial-frequencies (0.0, 0.5, 1.0, 1.5, 2.0 oct)

1 rf-location ([0,0])

5 phases (complex-cell, 0, 90, 180, 270°).

## 4) Response of the DNM to the stimulus (ll.31-62)

The output of the function is plotted in three graphs. The abscissas of the graphs represent tunings of the model neurons: orientation, spatial-frequency, and phase.

# Manual\_Sample\_2.m

This script shows the software can process multiple visual stimuli at once.

## 1) Load image files (ll.8-29)

Four images are loaded and are integrated into a single three-dimensional matrix (stimuli).

## 3) Apply the DNM to the stimulus (ll.34-43)

The output (resp\_DivNorm) of the function becomes a five-dimensional matrix. The fifth additional dimension represents the four stimuli. Hence, the matrix has 1200 elements (300 × 4) that are responses (firing rates) of the 300 model neurons to the 4 stimuli:

# Manual\_Sample\_3.m

This script shows how parameters of the DNM can be modified.

## 2) Set-up a "model" structure M with modified parameters (ll.14-59)

The standard parameter set is first obtained from DMPL\_defaultSpecs (l. 15). Then, parameters in the set can be directly accessed and can be modified. After the modification, the function DMPL\_prepareSpecs is called to generate the image filters based on the modified parameters.