Abstract

We have created a discrete model of the visual cortex and a program to calculate the angular orientation at each point of an image. Our research is focused on combining these tools to implement an image completion algorithm that is based on biological data of how the brain fills in images.

1 General Introduction

Light is converted to electrical impulses by the retina and then transferred to the primary visual cortex (V1), where simple cells process the image. We wish to model how the brain fills in occlusions; if there is a hole in the image, the brain does not see the hole but somehow fills it in. V1 is made up of simple cells which are tuned to a position \((x, y)\) and an orientation \(\theta\). The simple cell is excited if the dominant orientation of its receptive field, centered at \((x, y)\), is parallel to \(\theta\). This can also be interpreted as an edge detection along the \(\theta\) direction.

2 Modeling Simple Cells

2.1 Gabor Filters

One way to model V1 simple cells is to use Gabor filters. Gabor filters are popularly used to model simple cells because they use two simple and familiar functions. The foundation of the Gabor filter is a product of a sine wave and a Gaussian curve in complex space. This product is then convolved with a patch of the image to produce a response value that we take as the excitement level of the corresponding simple cell. An example of this filter is shown in (a) of the filter comparison figure.

The general equation for a Gabor filter is:

\[
G(x, y) = \exp\left(- \frac{(x')^2 + \gamma^2(y')^2}{2\sigma^2}\right) \exp\left(i \left(\frac{2\pi}{\lambda}x' + \psi\right)\right)
\]

where

\[x' = x \cos \theta + y \sin \theta \text{ and } y' = -x \sin \theta + y \cos \theta \text{ and } i^2 = -1\]

2.2 CORF Filters

A better way to model simple cells is to use CORF filters, which additionally bundle in a model of LGN cells. CORF filters are similar in that they output responses at specific \((x, y, \theta)\), but they more accurately model biological functionality and more effectively detect contours. Each filter employs eight sub-units that each have their own response level and model LGN cells in the brain. By combining each sub-unit’s response via a weighted geometric mean, we can obtain the response levels of the entire image with significantly less noise than with Gabor filters. This accuracy, however, comes at the cost of greatly increased computational complexity.
3 Roto-Translation Space $RT$

We transform the image into $RT = \mathbb{R}^2 \times S$ space, which is very useful for modeling simple cells in V1. By assuming that a simple cell is tuned to a position $(x, y)$ and orientation $\theta$, we find a natural correspondence with the $RT$ space, as the simple cell can be mapped to the point $(x, y, \theta) \in RT$.

Our model of the simple cells in V1 is a discrete lattice with planes corresponding to different angles. Simple cells of V1 correspond to lattice points $(x, y)$. Cells which are excited along a given direction $\theta_0$ will be in the same plane. Connections between lattice points can only be formed in the following two cases:

1. To lattice points with the same $(x, y)$ coordinates in the planes immediately above or below.
2. To lattice points in the same plane which lie along the $\theta_0$ direction.

4 Programs

4.1 $RT$-Lifting and Diffusion

The first step is to transform an image into simple cell response values. This Mathematica notebook loops over the image using 12 different Gabor filters tuned to angles between 0 and $\pi$ in steps of $\frac{\pi}{12}$. The radius for the filter can be changed, but generally a radius of about 5 pixels seems to work best. The Gabor responses for each pixel are stored into an array and normalized to fall in the interval $[0, 1]$. These responses can be visualized using two different methods: one method is to sum the 12 Gabor responses for each pixel, and the other is to just take the maximum value out of the 12 responses.

After obtaining the response array, we construct a discrete $RT$ graph. Each vertex in the graph corresponds to a simple cell and has $(x, y, \theta)$ coordinates. Planar connections are only allowed between vertices that lie along the $\theta$ direction, and inter-planar connections are allowed for vertices with the same $(x, y)$ coordinates.

We then run a basic diffusion algorithm that uses the discrete $RT$ graph to determine which model neurons are allowed to communicate with each other and spread their excitement level. This process is sped up by only considering the neurons whose normalized excitement level exceeds a defined threshold.

The Mathematica notebook for this process is named fast_diffusion.nb

4.2 CORF Filter

LGN cells, a cell type overlooked by Gabor filters, are modeled by using a pool of center-on and center-off cells dispersed through the input image. These cells, modeled as difference-of-Gaussian
operators, are the smallest agents of contour detection in the CORF model. Each model LGN cell has the same polarity (i.e. the same on or off state) and can see the same number of pixels as their neighboring LGN cells. More specifically, neighboring LGN cells share the same receptive field size. The sum of each LGN cell’s weighted response to contour changes is computed and stored by its parent sub-unit. Naturally, this parent sub-unit’s receptive field is the union of each underlying LGN cell’s receptive field, and the polarity of the sub-unit is the same as the polarities of its child LGN cells. As a result of the former, sub-units can detect contour changes on wider areas than lone LGN cells can simply because they are privy to more pixels.

By combining the sub-units’ response levels with their polarity and orientation assignments, CORF model cells can also exhibit orientation selectivity. CORF model cells consist of—as per the specifications provided by Azzopardi and Petkov—eight sub-units: four center-on and four center-off. The properties of the input pixels determine the configuration of the polarities and orientation preferences of each LGN cell, and the polarities and physical arrangement of the concerned sub-units is determined from the responses of the child LGN cells. Each CORF model cell then assumes the numerical value of the weighted geometric mean of each of its eight sub-units’ individual response levels. Passing the responses through a geometric mean guarantees that the CORF model cells exhibit responses only when all of their sub-units have non-zero response levels. This further eliminates noise from the output.

5 LED Cube

We are also working on a physical model of our work, culminating in an LED cube. An LED cube is an $8 \times 8 \times 8$ grid with LEDs at every intersection. Each layer of the cube will represent one $\theta$ layer of the Gabor filter responses, so that the result is the 2-D image lifted into 3 dimensions. Each layer is controlled by an Arduino, which allows us to set brightness values for each LED at each layer. This allows us to visualize a response value at a specific location in the image.

![Figure 5.1: A small scale rendering of what the cube will look like, with the red dots representing LEDs](image)

6 Models

There are two more models of V1 image completion that we are interested in trying to implement. They build on the same concept but utilize different algorithms to implement it. This concept, suggested by Citti and Sarti in 2006 and used by Hladky and Pauls in a paper from 2010, suggests that the simple cells form long-range connections, so that a simple cell at the boundary of the hole has a long-range connection with another simple cell also at the boundary of the hole. These two cells may then compare their excitement level and thereby ‘guess’ what is supposed to be in the hole. Such connections in $RT$ space turn out to be geodesics, which is a welcomed feature, as we assume that the brain minimizes the
path the signal has to travel. The image in Fig. 6.1 shows the completion algorithm used by Hladky and Pauls.

![Figure 6.1](image)

### 7 Future Work

Our plan for next semester, and future work in general, is to continue to use and improve the tools that we have created to implement these various models for image completion. One major improvement we are already working on is to adapt our CORF filter to take advantage of GPU programming. Since the process is highly parallelizable, we expect GPU programming to provide a significant speed boost to our CORF filter and make it feasible for general use. Our overall goal is to coordinate with researchers at UCSB to determine the effectiveness of our image completion models in order to further improve our work and provide insight as to how the visual cortex processes information.