

Motion Compensation by Simulated Cortical Cells

Al Ahumada
NASA Ames Research Center

Abstract

Motion compensation in visual cortex is modeled using quadrature pairs of receptive fields and a motion signal. Image samples are temporally combined by oscillating filters tuned for the image speed and the receptive field spatial frequency. Simulations of a 1-D model with a Gabor pyramid of simulated cortical cells and single-tuned oscillator integrators demonstrate its potential.

Introduction

Despite the small rapid motions of the eyes, the perceptual image is sharp. Kelly (1985) showed that detection of small details is not worsened by motion until it increases beyond the natural occurring level. Although motion compensation may be done at multiple levels, only very early compensation would easily support detection. MacLeod (2007) has pointed out the large disparity between the high following rate of cortical neurons ($\sim 100/\text{sec}$) and the slow nature of visual perception ($\sim 10/\text{sec}$). The fast response may support combining the images over time in a way that helps preserve spatial detail.

Motion compensation is similar to stereo disparity evaluation in that a multitude of possible motions or disparities must be taken into account. Freeman and Ohzawa (1990) showed that pairs of Gabor receptive fields in quadrature phase can convert the disparity search problem into a phase difference measurement problem. Qian and Anderson (1997) use this representation to compute both stereo disparity and motion energy.

Here the quadrature pair representation is used to reduce motion blur. As an input moves across a Gabor receptive field, the output oscillates. Integration of the output with an oscillator helps deblur the image if the oscillation frequency matches that of the Gabor output. This frequency is the product of the spatial frequency of the receptive field and the component of the image velocity in the direction orthogonal to the receptive field bars. Single tuned oscillators are used because they can be implemented by simple second order recursive filters.

The 1-D Model

Input

The contrast stimulus is a space and time image sequence. Figure 1 shows the case where the input is a single pulse moving with velocity, $V = -1$. Since the system is linear, its response to a single pulse characterizes it.

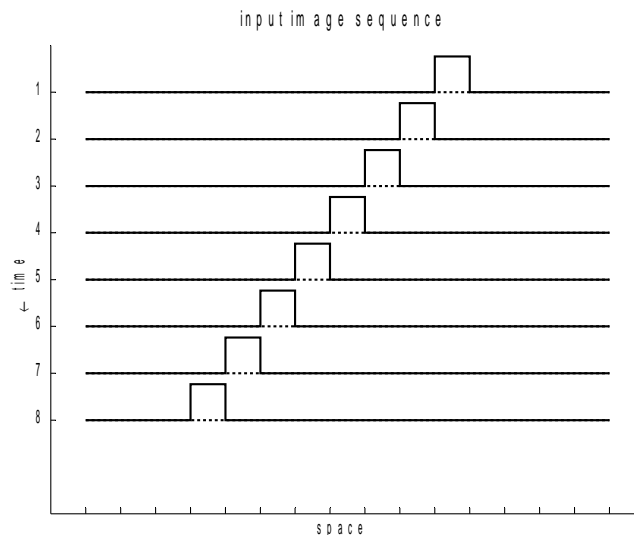


Figure 1. The 1-D image input sequence.

Cortical pyramid

Each image is converted to a cortex-like representation (Watson, 1987a,b; Watson and Solomon, 1997) using a Gabor transformation with quadrature pairs at each spatial position as indicated in Figure 2. The Gaussian standard deviation equals the sine wave period. Each spatial frequency channel has layers of quadrature pairs sampling the image at increasingly dense rates as their spatial frequency increases. The coefficients shown are sampling at the center position. The eight Nyquist frequency coefficients and the DC coefficient (not shown) have no sine phase component. The over-complete representation has $8+8+4+2+1=23$ channels to represent the 16 image values.

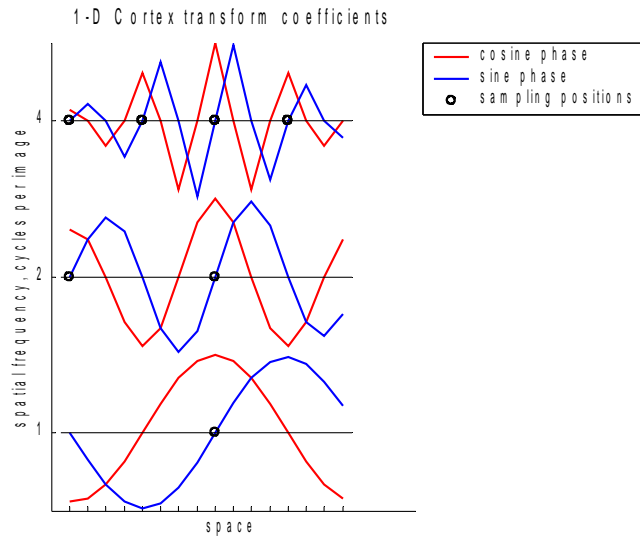


Figure 2. Coefficients of the cortex pyramid transform.

Integrating resonators

Figure 3 shows the temporal oscillators for the transform levels of Figure 2. Their temporal frequency f_T is determined by the image speed V (perpendicular to the Gabor orientation) and the channel spatial frequency f_S , $f_T = V f_S$. Their envelope is exponential with a time constant of three time sample. The model output in the cortex transform domain is the convolution of the input samples with the oscillators (see last lines in the Appendix).

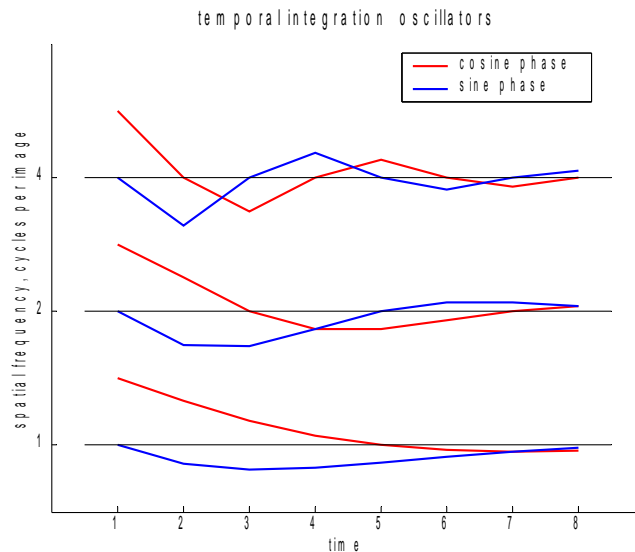


Figure 3. Temporal integrating oscillators. Each oscillator has a frequency proportional to the spatial frequency of the Gabor receptive field and the velocity of the image.

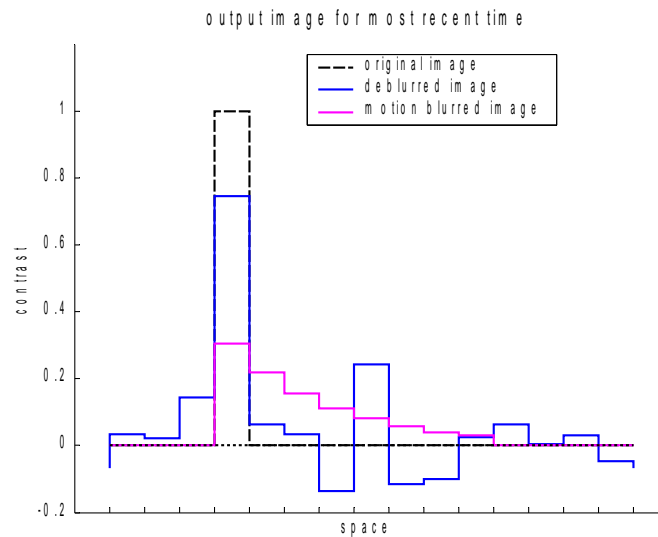


Figure 4. The solid line shows the output of the integrators at the most recent time, the deblurred image. The dashed line shows the original image and the dotted line shows the image exponentially integrated without the oscillators.

Output

Figure 4 shows the model output compared with the original pulse and the pulse temporally smeared with the same exponential integration time. The model output is returned to the input domain using a transformation computed by singular value decomposition. Ahumada (1992) shows how such a reconstruction could be “learned.” Although reconstruction artifacts appear, the model does achieve significant deblurring. Using a quality ratio similar to the Strehl ratio (last two lines of the Appendix) the original pulse has a quality of 1, the deblurred pulse has a quality of 0.89, and the exponentially blurred pulse has a quality of .70.

Discussion

Freeman, Adelson, and Heeger (1991) used quadrature pairs of oriented filters to add motion to stationary images. Here we are trying to invert that process. Better representations might lead to better performance (Simoncelli, Freeman, Adelson, Heeger, 1992). Both the physiological plausibility and the performance might be improved if the representation was organized in cortical columns (van Dijk & Martens, 1997). In such representations, phase may be advanced to make perception predictive. Deblurring might contribute to meta-contrast-like illusions (Breitmeyer, 1984).

Acknowledgments

This work was supported by the NASA Aeronautics Program. NIPS 2007 reviewers made several helpful suggestions.

References

- Ahumada, A. J. Jr. (1992) Learning receptor positions, in *Computational Models of Visual Processing*, eds. M. S. Landy and J. A. Movshon, pp. 23-34, Cambridge, Massachusetts: MIT Press.
- Breitmeyer, B. G. (1984) *Visual masking: an integrative approach*. New York: Oxford University Press.
- Freeman, R. D., and Ohzawa, I. (1990) On the neurophysiological organization of binocular vision. *Vision Research*, 30 (11), 1661-1676.
- Freeman, W. T., Adelson, E. H., Heeger, D. J. (1991) Motion without movement. *Proceedings of the 18th annual conference on computer graphics and interactive techniques*, pp 27 - 30.

Kelly, D. H. (1985) Visual processing of moving stimuli. *Journal of the Optical Society of America A*, 2 (2), 216-225.

MacLeod, D. (2007) Talk at the T. Cohn Memorial Conference, Berkeley, CA.

Qian, N., and Andersen, R. A. (1997) A physiological model for motion-stereo integration and a unified explanation of Pulfrich-like phenomena. *Vision Research*, 37 (12), 1683-1698.

Simoncelli, E. P., Freeman, W. T., Adelson, E. H., Heeger, D. J. (1992) Shiftable multi-scale transforms, *IEEE Transactions, IT-38*, pp. 587-607.

van Dijk, A. M., Martens, J.-B. (1997) Image representation and compression with steered Hermite transforms, *Signal Processing* 56, (1), 1-16.

Watson, A. B. (1987a) The Cortex transform: Rapid computation of simulated neural images, *Computer Vision, Graphics, and Image Processing*, 39, 311-327.

Watson, A. B. (1987b) Efficiency of an image code based on human vision, *Journal of the Optical Society of America A* 4(12), 2401-2417.

Watson, A. B., Solomon, J. A. (1997) A model of visual contrast gain control and pattern masking, *Journal of the Optical Society of America A*, 14, 2379-2391. .

Appendix: Matlab Code

```
% 1D Gabor motion compensation using temporal oscillators

% image parameters
Nf = 4; % log2 image length
Nx = pow2(Nf); % image length = 16
Nx2 = Nx/2 ; % 8
Nt = 8 ; % time samples
imseq = [ ... % image sequence, rows are 8 times, columns are 16 positions
          0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
          0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
          0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
          0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
          0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0
          0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
          0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
          0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
        ] ;
V = -1 ; % velocity

% Gabor cortex transform coefficients
s = 1 ; % Gabor window standard deviation in periods of sine wave
f = 2.^[0:Nf-1] ; % sine frequencies in cycles per image
samples = f/s ; % samples per image at non-DC frequencies
Nc = 1 + sum(samples); % Number of complex coefficients
gabcoef = zeros(Nc,Nx) ;
gabcoef(1,:) = ones(1,Nx)/Nx ; % DC coefficients
w = 2*pi*f/Nx ; % frequency in radians per pixel
sigma = Nx./samples; % pixels
x = [0:Nx/2] ;
ifilt = 1;
for iband = 1:Nf
    env = exp(-0.5*(x./sigma(iband)).^2) ;
    filt = env.*exp(i*w(iband)*x) ;
    ifilt = ifilt +1 ;
end
```

```

gabcoef(ifilt,1:Nx2+1) = filt ;
gabcoef(ifilt,2+Nx2:Nx) = conj(filt(Nx2:-1:2)) ;
offset = [sigma(iband)+[1:Nx-sigma(iband)] [1:sigma(iband)]] ;
for ioff = 1:samples(iband) - 1
    ifilt = ifilt +1 ;
    gabcoef(ifilt,:) = gabcoef(ifilt-1,offset);
end
end

% Gabor to image transformation
realgabcoef = [real(gabcoef);imag(gabcoef)] ;
[ugabcoef diaggabcoef vgabcoef] = svd(realgabcoef,0) ;
imrestore = ugabcoef*inv(diaggabcoef)*vgabcoef' ;

% make temporal integrators
t0 = 3 ; % exponential blur duration in pixels
tfilt = zeros(Nc,Nt) ;
wt = 2*pi*f*V/Nx; % radians per time sample
t = [0:Nt-1] ;
tfilt0 = exp(-t/t0) ;
tfilt0 = tfilt0/sum(tfilt0) ;
tfilt(1,:) = tfilt0 ;
ifilt = 2 ;
for iband = 1:Nf
    tfilti = tfilt0.*exp(i*wt(iband)*t) ;
    tfilt(ifilt:ifilt+samples(iband)-1,:) = repmat(tfilti,[samples(iband),1]) ;
    ifilt = ifilt +samples(iband) ;
end

% transform, integrate, and restore integrated image
gabim = gabcoef*imseq' ; % Gabor transform of image sequence
gabv = dot(conj(tfilt(:,Nt-[0:Nt-1])),gabim,2) ; % convolve with integrators
imout = [real(gabv);imag(gabv)]'*imrestore ;
figure(1); clf; plot(imout)
qualityratio = sqrt(max(imout).^2/sum(imout.^2)) % 0.8911
blurqualityratio = sqrt(1/sum(exp(-2*[0:7]/t0))) % 0.6992

```