Modeling the Initial Steps of Human Vision

Brian Wandell, Stanford University
David Brainard, University of Pennsylvania
ISETBio and ISET3d: Modeling 3D scenes and human image formation

Brian Wandell and David Brainard

QUANTITATIVE MEASUREMENTS

COMPUTATIONAL MODELS

CHECK AND SHARE
What I review and why

• **Background:** ISETBio (Image Systems Engineering Tools for Biology) provides computational tools that implement the ideas developed by vision scientists.

• **What:** ISET3d is a tool that extends ISETBio computations from planar images into three-dimensional scenes. My goal today is to explain ISET3d.

• **Why:** The extension to 3D may be relevant to scientists and engineers who aim to
  o Model and understand the visual encoding of natural images and stereo vision,
  o Optimize devices, including cameras and displays, for capturing and rendering 3D scenes.
3D scene spectral radiance in the world and at the eye

Gershun (1936)

Ray intensities: \( L(x,y,z,\alpha,\beta,\lambda,\theta) \)
- Position \((x,y,z)\)
- Azimuth and elevation \((\alpha, \beta)\)
- Wavelength \((\lambda)\)
- Polarization \((\theta)\)

Adelson and Bergen (1991)

Ray intensities: \( L(u,v,\alpha,\beta,\lambda) \)
- Position \((u,v)\)
- Azimuth and elevation \((\alpha, \beta)\)
- Wavelength \((\lambda)\)

The world beyond the RGB image representation must be accounted for in the simulation

Light field

Plenoptic function
• There are many tools for creating realistic 3D scene geometries

• We use Cinema 4D from Maxon because it integrates well with ray tracing methods

• Maxon offers **free** Cinema 4D licenses to students and teachers, and low- or no-cost “lab” licenses for schools.
Progress in computer graphics enables us to create synthetic and yet highly realistic input data.

We use PBRT because it is open-source, extensible, and taught at Stanford.

The simulations can maintain meaningful units; quantitative computer graphics.

A GPU version is scheduled to be released by Pharr et al. in about 2 months.
In the next 15 minutes I illustrate the ISET3d computational framework.

- I will show you
  - The kinds of stimuli that we are producing and
  - The programming approach in the specific case of simulating human physiological optics (image formation)

- David and I are producing YouTube videos of the tools that illustrate many more computations

- We use ISET3d for camera design, as well.
Use computer graphics and ray-tracing to model how spectral, 3D scenes are transformed by human optics to the retinal irradiance.
Comparison of eye models

The code flexibility accommodates the major human eye models (Lian et al. 2019, Journal of Vision).
This script illustrates the programming philosophy from the user’s perspective.

By default the sceneEye uses the Navarro model to render a 3D scene; LeGrand and Arizona eye models are also included.

Example code

```plaintext
thisSE = sceneEye('letters at depth', 'human eye', 'legrand');

>> thisSE

thisSE = sceneEye with properties:

    name: 'lettersAtDepth'
    modelName: 'legrand'
    usePinhole: 0
    recipe: [1×1 recipe]
    lensDensity: 1
```

PBRT files

Human eye model
Example code

- The code doing the computational work in ISET3d is managed within
  - The set/get methods
  - PBRT calculations
- You can ‘set’ many camera, rendering, and scene parameters
- You can ‘get’ many more parameters by calculation
- There are a number of methods ‘render’, ‘summary’ and others

```matlab
% Suppose you are in focus at the proper distance to the edge. And we turn % on chromatic aberration. That will slow down the calculation, but makes % it more accurate and interesting. We only use 8 spectral bands for % speed. You can use up to 31.
nspectralbands = 8;
thisse.set('chromatic aberration', nspectralbands);

% This is the distance we calculate above
thisse.set('focal distance', 1);

% Controls the rendering noise vs. speed by setting the number of rays.
thisse.set('rays per pixel', 128);

% Increase the spatial resolution by adding more spatial samples.
thisse.set('spatial samples', 384);

% This takes longer than the pinhole rendering, so we do not bother with % the depth.
oi = thisse.render('render type', 'radiance');
oiwindow(oi);
```
Stereo pairs: move the camera position by 6 cm

```
thisEye.set('from', loc)
```
• This ISET3d code makes the stereo pair of the Chess retinal irradiance, imaged through the LeGrand model eye

• I set the lens density to 0 so the scene would not look very yellow. I will explain this in a moment

```matlab
%% Make an oi of the chess set scene using the LeGrand eye model
thisSE = sceneEye('chess set scaled','human eye','legrand');
thisSE.set('lens density',0);  % Just because I can
thisSE.set('rays per pixel',512);  % Pretty quick, but not high quality
oiLeft = thisSE.render;  % Render and show
oiWindow(oiLeft);

%% Shift the eye position
% Change the eye position (from) but stay focused on the same object (to).
% I shifted the eye position by a lot (12 mm) so the image difference is be
% easy to see. The inter-pupil difference is really only 6-8 mm
from = thisSE.get('from');  % Current camera location
thisSE.set('from',from + [0.012,0,0]);  % Shift it 12 mm
oiRight = thisSE.render;
oiWindow(oiRight);
```
Inert pigments (e.g. lens transmission) are included and controlled.
Vergence and Accommodation

Where the eye (or eyes) is looking is controlled

```
thisEye.set('to', loc)
```

1.66 D (Left)

1.66 dpt (Right)
Calculating cone responses and eye movements

Scenes can be quite complex and realistic

- We have more than 25 high quality scenes like these
- The geometry, reflectance, lighting and textures can be edited (ask me)
- This collection will grow and already includes HDR, inter-reflections, many types objects, materials, textures, shadows, occlusions
• We have more than 25 high quality scenes like these

• The geometry, reflectance, lighting and textures can be edited (ask me)

• This collection will grow and already includes HDR, inter-reflections, many types objects, materials, textures, shadows, occlusions
Scenes can be quite complex and realistic

- We have more than 25 high quality scenes like these
- The geometry, reflectance, lighting and textures can be edited (ask me)
- This collection will grow and already includes HDR, inter-reflections, many types objects, materials, textures, shadows, occlusions
Scenes can be quite complex and realistic

- We have more than 25 high quality scenes like these

- The geometry, reflectance, lighting and textures can be edited (ask me)

- This collection will grow and already includes HDR, inter-reflections, many types objects, materials, textures, shadows, occlusions
Scenes can be quite complex and realistic

- We have more than 25 high quality scenes like these
- The geometry, reflectance, lighting and textures can be edited (ask me)
- This collection will grow and already includes HDR, inter-reflections, many types objects, materials, textures, shadows, occlusions
ISETBio Team and Funding

- Brian Wandell
- Trisha Lian
- Haomio Jiang
- James Golden
- David Brainard
- Nicolas Cottaris
- Xiaomao Ding
- Lingqi Zhang
- E.J. Chichilnisky
- Fred Rieke
- Joyce Farrell
- Jon Winawer

facebook research
SIMONS FOUNDATION
NIH National Eye Institute
ISETBio: Modeling the Initial Steps of Human Vision

David H. Brainard and Brian A. Wandell
Thanks to Nicolas P. Cottaris
ISETBio Team and Funding

Facebook Research

Simons Foundation

NIH National Eye Institute
Encoding of light at the retina has large implications for perception
Color matching is mediated by encoding of light spectra by the cones.
We would like to understand more generally implications of early vision

- Formation of the retinal image and optical blur
  - Including effects of 3D scene structure
- Spatial and spectral sampling by the interleaved cone mosaic
- Phototransduction
- Fixational eye movements
- Bipolar and retinal ganglion cell processing
• ISETBio is a set of open source Matlab tools that quantitatively model early vision.
• ISETBio is image computable.
• Helps clarify how different elements of the eye and neural processing impact visual perception.
• Today’s webinar is an introduction to ISETBio.
ISETBio components – scene and retinal image

- **Scene object**
- **Optics object**

- **Photons/pixels/nm/sec**

- Courtesy Nicolas Cottaris
ISETBio components – scene and retinal image

- Scene object
- Optics object
- Retinal image object

Photons/pixels/nm/sec
ISETBio components – retinal image and cone isomerizations

- retinal image object
- cone mosaic object

Photon flux per pixel per nm per second

Courtesy Nicolas Cottaris
ISETBio components – retinal image and cone isomerizations

retinal image object → cone mosaic object → cone photopigment mosaic excitation

photons/pixels/nm/sec → R*/cone/sec
Example: cone mosaic isomerizations to gratings at different contrasts

scene \((c, sf)\)

\(c = 0\)

\(c = 100\%, sf = 16\ c/\text{deg}\)

Courtesy Nicolas Cottaris
Accounting for absolute sensitivity: fixational drift

- Cone photopigment excitation
- Fixational eye movement object
- Cone photopigment excitation sequence

R*/cone/sec
Accounting for absolute sensitivity: photocurrent transduction

cone photopigment excitation sequence → outer segment object → cone outer-segment photocurrent response

- **R*/cone/sec**
- **30 cd/m²**
- **0.6 0.4 0.2 0.0**
- **100**
- **power (pA² Hz⁻¹)**
- **10⁻¹ 10⁻² 10⁻³ 1**
- **time (sec)**
- **0.0 0.1 0.2 0.3 0.4**
- **frequency (Hz)**
- **3 10 30 100 300**
- **pAmps**
Accounting for contrast sensitivity

1. Updated optics & cone mosaic modeling has a minor impact relative to the Banks ’87 estimate (factor of 1.7 at 2 c/deg),

2. Computational observers, which learn visual tasks by observing neural responses, result in a significant sensitivity drop across the entire spatial frequency range (accumulated factor of 2-3).

3. Inclusion of fixational eye movements, requires non-linear computational observers, and further reduces sensitivity across the entire spatial frequency range (accumulated factor: 7-10).

4. Inclusion of photocurrent encoding further reduces sensitivity approaching psychophysical limits (accumulated factor: 18-30).
Open-Source, Matlab. ISETBio itself is available here: https://github.com/isetbio/isetbio
Download and add to your Matlab path

Examples from the next part of this talk are here: https://github.com/isetbio/ISETBioLiveScript
Download and put wherever you like

Videos on YouTube. Search Google with “ISETBio Tutorials” and look under videos:
1) ISETBio SceneLiveScriptTutorial
2) ISETBio ComputeIsomerizationsTutorial
3) ISETBio EyeMovementsPhotocurrentTutorial
4) ISETBio ContrastDetectionPerformanceTutorial
Tools for modeling image systems engineering in the human visual system front end

Manage topics

- calculators
- configuration
- data
- demoapps/fixationalEyeMovements
- external
- isettools
- local
- scripts
- tutorials
- validation
- .gitignore
- Contents.m
- LICENSE
- README.md

Latest commit 38db4ec 22 minutes ago

Nicolas Cottaris Undoing the (wrong) y-coordinate flip.
Some Papers that Use ISETBio

A computational-observer model of spatial contrast sensitivity: Effects of wave-front-based optics, cone-mosaic structure, and inference engine

Nicolas P. Cottaris; Haomiao Jiang; Xiaomao Ding; Brian A. Wandell; David H. Brainard

Journal of Vision April 2019, Vol.19, 8. doi:https://doi.org/10.1167/19.4.8

Ray tracing 3D spectral scenes through human optics models

Trisha Lian; Kevin J. MacKenzie; David H. Brainard; Nicolas P. Cottaris; Brian A. Wandell


Modeling visual performance differences ‘around’ the visual field: A computational observer approach

Eline R. Kupers; Marisa Carrasco, Jonathan Winawer

Published: May 24, 2019 • https://doi.org/10.1371/journal.pcbi.1007063

Spatial summation in the human fovea: Do normal optical aberrations and fixational eye movements have an effect?

William S. Tuten; Robert F. Cooper; Pavan Tiruveedhula; Alfredo Dubra; Austin Roorda; Nicolas P. Cottaris; David H. Brainard; Jessica I. W. Morgan

Journal of Vision August 2018, Vol.18, 6. doi:https://doi.org/10.1167/18.8.6
Eye Movement Model Demo

Open-Source, Matlab. ISETBio itself is available here: https://github.com/isetbio/isetbio
Download and add to your Matlab path

Examples from the next part of this talk are here: https://github.com/isetbio/ISETBioLiveScript
Download and put wherever you like