Cue Integration Cosmo 2012

Paul R Schrater,
Departments of Psychology and Computer Science, University of Minnesota
Complex Perceptual Problems are ambiguous

**Recognition, Shape, Material**

A) Invariance to Pose, lighting, and shading.

B) Single image ambiguity: Bas relief transform of shape lighting (viewpoint)

C,D) Reflectivity vs. paint
Ambiguity:

can be characterized by a probability distribution for which multiple possibilities have equal/similar probability.
Overcoming ambiguity requires applying additional knowledge. *Prior knowledge* and *auxiliary information* can further disambiguate candidate scene interpretations.
Outline

• How do we specify/describe what we mean by generative knowledge
• What kinds of generative knowledge do people use?
• How do we test for its use?
Forward models for perception:
Built in knowledge of image formation

Images are produced by physical processes that can be modeled via a rendering equation:

\[ I = f(A,L,V) = f(\text{scene}) \]

\( A \) = object attributes
\( L \) = description of the scene lighting
\( V \) = viewpoint and imaging parameters (e.g. focus)

Modeling rendering probabilistically:

**Likelihood:** \( p(I \mid \text{scene}) \)

*e.g. for no rendering noise*

\[ p(I \mid \text{scene}) = \delta(I - f(\text{scene})) \]

How do we describe the other kinds of generative knowledge?
Bayesian Networks: Modeling complex inferences

This model represents the decomposition:
\[ P(X_1, X_2, X_3, X_4) = P(X_4 | X_2) \cdot P(X_3 | X_1, X_2) \cdot P(X_1) \cdot P(X_2) \]

**Nodes:** random variables
\[ X_1, \ldots, X_4 \]
Each node has a conditional probability distribution

**Links:** direct dependencies

**Data:** observations of \( X_3 \) and \( X_4 \)

**EXAMPLE**
- \( X_1 \) object size
- \( X_2 \) object distance
- \( X_3 \) image size
- \( X_4 \) “felt” distance
Basics of Bayesian inference

• Bayes’ rule:
  \[ P(h \mid d_1, d_2) = \frac{P(d_1 \mid h)P(d_2 \mid h)P(h)}{\sum_{h_i \in H} P(d_1 \mid h_i)P(h_i)} \]

• An example
  – Data: Large image size \( d_1 \) and small felt distance \( d_2 \)
  – Some hypotheses:
    1. Small object, near
    2. Large object, more distant
  – Likelihood \( P(d_1 \mid h) \) is ambiguous with respect to 1 and 2
  – Prior probability \( P(h) \) may favor hypothesis 1 over 2
  – Likelihood \( P(d_2 \mid h) \) favors 1
  – Posterior \( P(h \mid d_1, d_2) \) favors 1 over 2
Generative Knowledge

Knowledge about the dependencies between variables can be represented by a graphical model in two ways:

1) As a connective graph (right)
2) As an inferential graph (explained next)
Forward Graphics Analogy

- Sample a scene type
- Sample \( N \) object classes
- Sample Objects from each class (locations and attributes for each object)
- Sample rendering variables (lights, viewpoint)
- Sample image features from rendered scene
The graphical model for scene *inference* requires different structure for each scene

![Graphical Model Diagram](image-url)

However, this structure is part of what we INFER in scene perception!
Non-parametric Bayes

Random variables for document clustering

- A word is a multinomial random variable $w$
- A topic is a multinomial random variable $z$
- A document is a Dirichlet random variable $\theta$

Treats number of words and topics as random variables
An analogy

- Random variables for document clustering
  - A word is a multinomial random variable $w$
  - A topic is a multinomial random variable $z$
  - A document is a Dirichlet random variable $\theta$

- Random variables for scene inference
  - An object class is a multinomial random variable $w$
  - A subscene is a multinomial random variable $z$
  - A scene is a Dirichlet random variable $\theta$
Non-Parametric Bayes Model

- **Parametric vs. non-parametric Bayes**
  - **Parametric:** Fixed parameterization of the prior
    - Needs prior on space of all possible scenes
    - Difficult to learn models (curse of dimensionality)
    - Has generated skepticism of Bayes for vision
  - **Non-parametric:**
    - Developed in response to limitations of parametric approach
    - *Only generates scene graph during inference*
    - Needs prior on scene construction (not scenes)
    - Parameters naturally coupled, reducing dimensionality
    - Increasingly used for “hard problems” in machine learning
    - Examples: Latent Dirichlet allocation, Chinese restaurant process, Indian buffet process, etc.
Recent work in computer vision using this approach (Sudderth et al, 2006)

- Visual “gist” observations
- Scene category: kitchen, office, lab, conference room, open area, corridor, elevator and street.
- Object class
- Particular objects
- Local image features
“Top-down” information: a representation for image context

Images

80-dimensional representation

Credit: Antonio Torralba
“Bottom-up” information: labeled training data for object recognition.

- Hand-annotated 1200 frames of video from a wearable webcam
- Trained detectors for 9 types of objects: bookshelf, desk, screen (frontal), steps, building facade, etc.
- 100-200 positive patches, > 10,000 negative patches
Vision as probabilistic parsing

(Han & Zhu, 2006; c.f., Zhu, Yuanhao & Yuille NIPS 06)
- **Scene**: type puts distributions on constituents, layout, lighting, etc
- **Object class**: puts distribution on object attributes
- **Image formation**: puts distribution on image measurements given objects
- **Dynamics model**: transformations
Different relationships between image measurements and object attributes lead to different inference problems.

Object property inference frequently requires knowing aspects of the scene (how many objects are present, illumination, object layout and pose, etc).
Testing Image generative knowledge

• How do we test whether people understand the relationship between object attributes and image measurements?

• Difficulty: Experimental design must eliminate *ambiguity in scene perception* (*number of objects, lighting, etc*).
  – (otherwise not studying image formation generative knowledge at all)

• Case studies:
  – Cue integration (quantitative)
  – Explaining away (previously qualitative)
Cue Integration
Explaining away


Surface color: white or pink
Shape: corner or roof

Observed chroma
Observed stereo disparity

Light Source angle

Observed shading
Observed boundary
Experiment 1: Humans use size cues to improve distance perception
The “size / distance” problem

- **Size** and **distance** are *ambiguous* given only a monocular image size cue
  - Emmert’s Law (Boring, 1940; Weintraub & Gardner, 1970)
Quantitative Predictions for Explaining away?

EXAMPLE

- A object size
- B object distance
- X image size
- Y “felt” distance

- Sensory generative knowledge:
  - constrains possible size & distance combinations to those consistent with the image size cue (Epstein et al., 1961)

- Auxiliary size cue:
  - rules out size & distance combinations that are inconsistent with auxiliary cue
  - allows unambiguous inference of distance

- Consistent with feature of Bayesian reasoning: Explaining Away (Pearl, 1988)
Psychophysical Methods 1: Trial procedure

- 6 human participants in virtual reality workbench (PHANToM & 3D graphics)
- **Exploration phase:**
  - **NO-HAPTIC:** view ball, no touching
  - **HAPTIC:** view ball and touch surface with fingertip

![Image of a person using a virtual reality setup]

**Interception**
Methods 2

• **Interception phase:**
  – Depress mouse
  – Ball moves to left of scene
  – Begins to approach and move rightward
  – Participant positions fingertip along “constraint line” to intercept

• **Computer records:**
  – True distance as *crossing distance*
  – Fingertip position as *judged distance*
Predictions:

1) NO-HAPTIC case:
   - Judged distances depend on **ball size**
   - Substantial errors in judged **distances** due to ambiguity

2) HAPTIC case:
   - Judged distances depend **LESS** on **ball size**
   - Reduced errors due to *explaining away* of inconsistent distances
Judged distances vs. crossing distances (participant 4)
Results:

Size dependence

Accuracy
• Bayesian model does a good job of predicting data

• Modeling the participants as “sampling from their posteriors” does better job of predicting data than modeling them as “MAP estimators”

• Reasonable noise estimates:
  - Vis. angle noise std. dev. $\sim [6, 30]$ minutes $@ [81, 410]$
  - Haptic size noise std. dev. $\sim [2, 5]$ mm $@ [14, 42]$
Size-change perception

- Extension of *size/distance* problem:
  - *size-change* perception

- Example:
  - Imagine viewing a balloon whose *retinal image size* is *shrinking*
  - The balloon may be *deflating*, OR *inflating* and receding rapidly
  - Knowing the *distance-change* rate can disambiguate the *size-change* rate

- Experimental question:
  - Can auxiliary *distance-change cues* improve *size-change* judgments?
  - Are both HAPTIC and STEREO *distance-change* cues effective?
Psychophysical Methods 1

- 11 human participants in virtual reality workbench (PHANToM & 3D graphics)
  - (1 outlier was removed)

- **Stimulus**: monocularly-viewed ball that changed in size and distance

- Distance-change cues:
  - **HAPTIC**: 1 fingertip “stuck” in center of ball as it moves
  - **STEREO**: binocular images consistent with real physical projection

- After 1000ms, participant chooses:
  - **INFLATING** or **DEFLATING**
Methods 2:

- 330 trials per 4 distance-cue cases:
  1) No Auxiliary cues
  2) Haptic-only
  3) Stereo-only
  4) Haptic & Stereo

- Each case: 3 psychometric functions - 11 points x 10 repetitions per point (black dots) - were measured.

- Diagonal, dashed line: size- & distance-change combinations that yield ZERO image size-change.

- Vertical, dotted line: boundary of unbiased discrimination between inflating and deflating sizes.
Predictions:

2 predictions for “explaining away” observer:

1. **No Auxiliary case**: psychometric curves along the diagonal, dotted line

2. **Auxiliary cases**: psychometric curves along the vertical, dotted line
Results

1. No Auxiliary case: the size-change judgments are based on image size-change.

2. Haptic-only, Stereo-only, Haptic & Stereo: increased veridicality, physical size-change is more accurately judged.
Summary of participants’ normalized slopes

Image size prediction

Normalized slope vs. Participant #

Image size prediction

veridical
Why is \textit{stereo} > \textit{haptic}?

- \textbf{Follow-up experiment}: measured stereo \& haptic distance-change cue reliabilities (Ernst, 2005)

- \textbf{2IFC}: “Which interval contained faster ball?”

- Psychometric function (cumulative normal) slope gives us each cue’s noise std. dev.
NO CORRELATION $\rightarrow$ not simply a difference in auxiliary cue quality
Experiment 2: Conclusions

• Participants use **distance-change cues** to improve their **size-change** perception.

• **Stereo distance-change cue** is more useful than **haptic**
  – There is a discrepancy between how haptic and stereo distance information are used to improve size-change judgments.

• **Haptic** and **stereo** distance-change cues have similar reliability
  – (perhaps even haptic > stereo)

Possible reasons for stereo/haptic discrepancy:
  – Brain is suboptimal - does not exploit haptic cue’s full potential
  – Brain understands haptic distance cue is less likely to be causally-related to image size cue, thus only integrates it partially (Koerding et al., 2007)

• Next steps:
  – Quantitative Bayesian model
  – Causal model
General Conclusions

• Uncertainty and ambiguity plague perceptually-guided actions.

• The brain has knowledge of each, and forms percepts and plans actions to overcome their negative consequences.

• Generative knowledge has (potentially) a hierarchical structure.

• Non-parametric Bayesian models provide a language to handle the difference between fixed relationships and those that vary from scene to scene, sharing relevant information across scenes.

• Such processing is characteristic of Bayesian reasoning and decision-making.
Quantitative Predictions for Explaining away?

**EXAMPLE**

- $A$ object size
- $B$ object distance
- $X$ image size
- $Y$ “felt” distance
Making more Complex Qualitative Predictions

- Given a network structure
- Linearize around values of hidden variables to 2nd order (moment matching, taylor, Laplace)

\[
\begin{bmatrix}
X \\ Y
\end{bmatrix} = T \cdot \begin{bmatrix}
A \\ B
\end{bmatrix} + \begin{bmatrix}
\omega_X \\ \omega_Y
\end{bmatrix}
\]

GOAL: Not meant to be a substitute for modeling, but how do you get cute “cue weight formulas” for complex models
Making more Complex **Qualitative** Predictions

**GOAL:** Not meant to be a substitute for modeling, but how do you might get cute “cue weight formulas” for complex models

- **Linearization**

\[
\begin{bmatrix}
  x \\
  y 
\end{bmatrix} = \begin{bmatrix}
  1 & 0 \\
  1 & 1 
\end{bmatrix} \begin{bmatrix}
  a \\
  b 
\end{bmatrix} + \begin{bmatrix}
  \omega_x \\
  \omega_y 
\end{bmatrix}
\]

\[z = Tx + w\]

- **Assume Gaussian Noise**

\[P(a)P(b) = P(x) = N(x \mid \mu_{\text{prior}}, C_{\text{prior}})\]

\[x = \begin{bmatrix}
  a \\
  b 
\end{bmatrix} \quad C_{\text{prior}} = \begin{pmatrix}
  \alpha & 0 \\
  0 & \beta 
\end{pmatrix}\]

- **Likelihood**

\[P(z \mid x) = N(z \mid Tx, C_{xy})\]

\[C_{xy} = \begin{pmatrix}
  \sigma_x^2 & 0 \\
  0 & \sigma_y^2 
\end{pmatrix}; \quad T = \begin{pmatrix}
  1 & 0 \\
  1 & 1 
\end{pmatrix}\]
Making more Complex **Qualitative** Predictions

PRIOR

\[ P(a)P(b) = P(x) = N(x \mid \mu_{\text{prior}}, C_{\text{prior}}) \]

\[ x = \begin{bmatrix} a \\ b \end{bmatrix} \quad C_{\text{prior}} = \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix} \]

LIKELIHOOD

\[ P(z \mid x) = N(z \mid Tx, C_{XY}) \]

\[ C_{XY} = \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix}; \quad T = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \]

WANTED

\[ P(b \mid z) \]
The Gaussian Distribution

\[ \mathcal{N}(x|\mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp \left\{ -\frac{1}{2\sigma^2} (x - \mu)^2 \right\} \]

\[ \mathcal{N}(x|\mu, \Sigma) = \frac{1}{(2\pi)^{D/2} |\Sigma|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right\} \]
Geometry of the Multivariate Gaussian

$$
\Delta^2 = (x - \mu)^T \Sigma^{-1} (x - \mu)
$$

$$
\Sigma^{-1} = \sum_{i=1}^{D} \frac{1}{\lambda_i} u_i u_i^T
$$

$$
\Delta^2 = \sum_{i=1}^{D} \frac{y_i^2}{\lambda_i}
$$

$$
y_i = u_i^T (x - \mu)
$$
Moments of the Multivariate Gaussian (1)

\[
\mathbb{E}[x] = \frac{1}{(2\pi)^{D/2} |\Sigma|^{1/2}} \int \exp \left\{ -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right\} x \, dx
\]

\[
= \frac{1}{(2\pi)^{D/2} |\Sigma|^{1/2}} \int \exp \left\{ -\frac{1}{2} z^T \Sigma^{-1} z \right\} (z + \mu) \, dz
\]

thanks to anti-symmetry of \( z \)

\[
\mathbb{E}[x] = \mu
\]
Moments of the Multivariate Gaussian (2)

\[ \mathbb{E}[xx^T] = \mu \mu^T + \Sigma \]

\[ \text{cov}[x] = \mathbb{E}[(x - \mathbb{E}[x])(x - \mathbb{E}[x])^T] = \Sigma \]
Partitioned Gaussian Distributions

\[ p(x) = \mathcal{N}(x | \mu, \Sigma) \]

\begin{align*}
x &= \begin{pmatrix} x_a \\ x_b \end{pmatrix} \\
\mu &= \begin{pmatrix} \mu_a \\ \mu_b \end{pmatrix} \\
\Sigma &= \begin{pmatrix} \Sigma_{aa} & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_{bb} \end{pmatrix}
\end{align*}

\[ \Lambda \equiv \Sigma^{-1} \quad \Lambda = \begin{pmatrix} \Lambda_{aa} & \Lambda_{ab} \\ \Lambda_{ba} & \Lambda_{bb} \end{pmatrix} \]
Partitioned Conditionals and Marginals

**Conditionals**

\[ p(x_a | x_b) = \mathcal{N}(x_a | \mu_{a|b}, \Sigma_{a|b}) \]

\[
\Sigma_{a|b} = \Lambda_{aa}^{-1} = \Sigma_{aa} - \Sigma_{ab} \Sigma_{bb}^{-1} \Sigma_{ba}
\]

\[
\mu_{a|b} = \Sigma_{a|b} \{ \Lambda_{aa} \mu_a - \Lambda_{ab} (x_b - \mu_b) \}
\]

\[
= \mu_a - \Lambda_{aa}^{-1} \Lambda_{ab} (x_b - \mu_b)
\]

\[
= \mu_a + \Sigma_{ab} \Sigma_{bb}^{-1} (x_b - \mu_b)
\]

**Marginals**

\[ p(x_a) = \int p(x_a, x_b) \, dx_b \]

\[
= \mathcal{N}(x_a | \mu_a, \Sigma_{aa})
\]
Partitioned Conditionals and Marginals
Bayes’ Theorem for Gaussian Variables

• Given

\[ p(x) = \mathcal{N}(x|\mu, \Lambda^{-1}) \]

\[ p(y|x) = \mathcal{N}(y|Ax + b, L^{-1}) \]

• we have

\[ p(y) = \mathcal{N}(y|A\mu + b, L^{-1} + AA^T \Lambda^{-1}) \]

• where

\[ p(x|y) = \mathcal{N}(x|\Sigma\{A^T L(y - b) + \Lambda \mu\}, \Sigma)^{-1} \]

\[ \Sigma = (\Lambda + A^T L A)^{-1} \]
Making more Complex Qualitative Predictions

PRIOR

\[ P(a)P(b) = P(x) = N(x \mid \mu_{\text{prior}}, C_{\text{prior}}) \]

\[ x = \begin{bmatrix} a \\ b \end{bmatrix} \quad C_{\text{prior}} = \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix} \]

LIKELIHOOD

\[ P(z \mid x) = N(z \mid Tx, C_{xy}) \]

\[ C_{xy} = \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix} ; \quad T = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix} \]

WANTED:

\[ P(b \mid z) \]

1 ) Bayes:

\[ \text{Given} \]

\[ P(x) = N(x \mid \mu_{\text{prior}}, C_{\text{prior}}) \]

\[ P(z \mid x) = N(z \mid Tx, C_{xy}) \]

\[ P(x \mid z) = N(x \mid \mu_{\text{post}}, C_{\text{post}}) \]

\[ \mu_{\text{post}} = C_{\text{post}}^{-1} \left( T^T C_{xy}^{-1} z + C_{\text{prior}}^{-1} \mu_{\text{prior}} \right) \]

\[ C_{\text{post}} = \left( C_{\text{prior}}^{-1} + T^T C_{xy}^{-1} T \right)^{-1} \]

2 ) Marginalize a:

\[ P(b \mid z) = N(b \mid \mu_{\text{post}}^b, C_{\text{post}}^{bb}) \]
Making more Complex Quantitative Predictions

EXAMPLE FOR:  \( P(a|z) \)

\[
\bar{\mu}_{post} = \bar{\mu}_{prior} + C_{prior}^T \cdot T^T \cdot \left( T \cdot C_{prior} \cdot T^T + C_{XY} \right)^{-1} \cdot (z - T \cdot \bar{\mu}_{prior})
\]

Different properties than cue combination!

\[
T = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \quad C_{prior} = \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix} \quad C_{XY} = \begin{pmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{pmatrix};
\]

\[
A^* = \frac{\alpha}{\sigma_x^2(\beta + \sigma_y^2) + \alpha(\beta + \sigma_y^2 + \sigma_x^2)} \left\{ (\beta + \sigma_y^2) X + \sigma_x^2 Y + (\beta + \sigma_x^2 + \sigma_y^2) \bar{\mu}_{prior}^A + (\beta + \sigma_y^2) \bar{\mu}_{prior}^B \right\}
\]

Cue weights don’t sum to one, both priors matter, etc.
Making more Complex **Qualitative** Predictions

\[
P(b \mid z)
\]

**WANTED:**

1) **Bayes:**

\[
P(x) = N(x \mid \mu_{\text{prior}}, C_{\text{prior}})
\]

\[
P(z \mid x) = N(z \mid Tx, C_{XY})
\]

\[
P(x \mid z) = N(x \mid \mu_{\text{post}}, C_{\text{post}})
\]

\[
\mu_{\text{post}} = C_{\text{post}}^{-1} \left( T^T C_{XY}^{-1} z + C_{\text{prior}}^{-1} \mu_{\text{prior}} \right)
\]

\[
C_{\text{post}} = \left( C_{\text{prior}}^{-1} + T^T C_{XY}^{-1} T \right)^{-1}
\]

2) **Marginalize a:**

\[
P(b \mid z) = N(b \mid \mu_{\text{post}}^b, C_{\text{post}}^{bb})
\]
Bayesian Networks: Modeling temporal dependence

This is just cue combination
But with a more complex prior.

\[ P(x_1) \quad P(x_2) \]
\[ X_1 \quad X_2 \]
\[ Z_1 \quad Z_2 \]
\[ P(z_1 \mid x_1) \quad P(z_2 \mid x_2) \]

**EXAMPLES**
- Sensori-motor integration
- Calibration
- Learning
- Trajectory Perception
Conceptual Overview
(The Kalman Equations)

• Basic Idea:

  Make prediction based on previous data

  Take measurement

  Optimal estimate (\(\hat{y}\)) =
  Prediction + (Kalman Gain) * (Measurement - Prediction)

  Variance of estimate =
  Variance of prediction * (1 – Kalman Gain)
Sequential Estimation, temporal independence

Contribution of the $N^{th}$ data point, $x_N$

$$
\mu_{ML}^{(N)} = \frac{1}{N} \sum_{n=1}^{N} x_n
$$

$$
= \frac{1}{N} x_N + \frac{1}{N} \sum_{n=1}^{N-1} x_n
$$

$$
= \frac{1}{N} x_N + \frac{N-1}{N} \mu_{ML}^{(N-1)}
$$

$$
= \mu_{ML}^{(N-1)} + \frac{1}{N} (x_N - \mu_{ML}^{(N-1)})
$$

correction given $x_N$
correction weight
old estimate
Conceptual Overview

- At time $t_3$, observer moves with velocity $\frac{dy}{dt}=u$
- Naïve approach: Shift probability to the right to predict
- This would work if we knew the velocity exactly (perfect model)

$$x_t = Ax_{t-1} + Bx + \omega_{\text{walk}}$$
$$y_t = Hx_t + \omega_{\text{sensory}}$$
Conceptual Overview

- Better to assume imperfect model by adding Gaussian noise
- $\frac{dy}{dt} = u + w$
- Distribution for prediction moves and spreads out

But you may not be so sure about the exact velocity
• Now we take a measurement at $t_3$
• Need to once again correct the prediction
• Same as before
Conceptual Overview

- Initial conditions ($x_{k-1}$ and $\sigma_{k-1}$)
- Prediction ($x^-_k$, $\sigma^-_k$)
  - Use initial conditions and model (e.g. constant velocity) to make prediction
- Measurement ($z_k$)
  - Take measurement
- Correction ($x_k$, $\sigma_k$)
  - Use measurement to correct prediction by ‘blending’ prediction and residual – always a case of merging only two Gaussians
  - Optimal estimate with smaller variance
Blending Factor

• If we are sure about measurements:
  – Measurement error covariance (R) decreases to zero
  – K decreases and weights residual more heavily than prediction

• If we are sure about prediction
  – Prediction error covariance $P^{-k}$ decreases to zero
  – K increases and weights prediction more heavily than residual
The set of Kalman Filtering Equations in Detail

**Prediction (Time Update)**

1. Project the state ahead
   \[ \hat{y}_{k} = A\hat{y}_{k-1} + Bu_k \]
2. Project the error covariance ahead
   \[ P_{-k} = AP_{k-1}A^T + Q \]

**Correction (Measurement Update)**

1. Compute the Kalman Gain
   \[ K = P_{-k}H^T(HP_{-k}H^T + R)^{-1} \]
2. Update estimate with measurement \( z_k \)
   \[ \hat{y}_k = \hat{y}_{-k} + K(z_k - H\hat{y}_{-k}) \]
3. Update Error Covariance
   \[ P_k = (I - KH)P_{-k} \]
Model example
Model Example

Models fill in gaps in information
Model example
Extrapolation depends on model
Do we have internal models for everything? **NO!**

**Classic example of a failure to learn Internal model**

Prediction - the reason for models

http://www.youtube.com/watch?v=kOguslSPpqo
Moving Dot task

- Prediction task
- Watch the dots move
- Position “bucket” to catch the emerging dots
Moving Dot task

• Prediction task
• Watch the dots move
• Position “bucket” to catch the emerging dots

Stimuli designed to be optimal for matched Kalman filter
Moving Dot task

- Capture the “bees”

Trajectory = \sim random walk
movie demo
Humans vs. Kalman Filter

- Demonstration of the task, human vs. filter performance
- Kalman filter predicts human behavior well
Matched Kalman excellent predictor
What are Human default Motion Models?

Object velocity:
- speed 5 m/s
- direction south

1. Constant velocity (CV)
   - maintain speed and direction

\[ v_t \rightarrow v_{t+1} = v_t \]

2. Constant acceleration (CA)
   - constant change in speed and/or direction

\[ a_t \rightarrow a_{t+1} \rightarrow v_{t+1} \]
Motion extrapolation task

- Fixation
- After 500ms dot travels
- Extrapolation judgment: “above” or “below”
  ![Diagram of above and below options]
- No reemergence; no feedback
- Determine the PSE based on staircase procedure
Motion extrapolation: Kalman filters for simple motions

Parameters of dot motion:
\[ x_k = [x, y, vx, vy, ax, ay]^T_k \]
position velocity acceleration

Process:

True state:
\[ x_k = A_k x_{k-1} + w_{k-1} \]

“w” \( \sim N(0, Q) \),
“Q” = covariance; reflects trust in prior (“A”)

\[ Q = 0 \rightarrow \text{complete trust} \]

“A” represents the prior model in the absence of data

→ **CV**: constant speed & direction: Linear motion prior
→ **CA**: constant change in direction: Circular motion prior
Motion extrapolation: Model behavior

**CA prediction using a Kalman filter**

\[ x_k = A_k x_{k-1} + w_{k-1} \]

Respond based on average estimate of 2\textsuperscript{nd} derivative
Motion extrapolation: Model behavior

CV prediction using a Kalman filter

\[ x_k = A_k x_{k-1} + w_{k-1} \]
Stimulus manipulations:

- Dot speed: 5 deg/s (constant)

- 2 staircases (i.e. 1U-2D, 2U-1D) per condition (curvature x sampling)
  - 100 trials per staircase

- 10 participants unaware of the purpose of the experiment
Motion extrapolation: Model behavior

The simple linear process predicts a wide range of behaviors by varying:

i. The specific internal model (CA, CV)
ii. Trust in model predictions vs. measurements

Decreased trust ("Q"):
- CA – flatter extrapolation
- CV – no change

(Less of the curve is used)
Motion extrapolation: Model behavior

The model predicts a wide range of behaviors by varying:

i. The specific internal model (CA, CV)
ii. Trust in the model
iii. Motion sampling

- CA – slightly more curved
- CV – little change
Results
Temporal dependence in cue weighting
Position uncertainty and blur

A

$\sigma_{\text{blur}} = 4^\circ \times 4^\circ$  $\sigma_{\text{blur}} = 24^\circ \times 24^\circ$

B

Visual JND (deg)

C

Horizontal error (deg)

Vertical error (deg)

Trial number in step phase

D

$\sigma_{\text{blur}} = 24^\circ$  ○  $\sigma_{\text{blur}} = 4^\circ$

E

$log(\lambda)$

$log(\text{Visual JNDs (deg)})$
Predictions

Mapping uncertainty parameter ($\hat{\sigma}_x$)

A

 mappings

$X_t$

$\hat{X}_t$

Low

B

High

 trial number

10 60 110 160
Directional Blur

\[ \sigma_{\text{blur}} = 4^\circ \times 4^\circ \]

\[ \sigma_{\text{blur}} = 24^\circ \times 24^\circ \]

\[ \sigma_{\text{blur}} = 4^\circ \times 24^\circ \]

\[ \sigma_{\text{blur}} = 24^\circ \times 4^\circ \]

B

Horizontal error (deg)

Vertical error (deg)

Trial number in step phase

C

D

Vertical error (deg)

Horizontal error (deg)

Horizontal error (deg)
Random walk increases adaptation rate
Bayesian sensory- and motor-adaptation model.

Shaded circles represent observed random variables
Unshaded circles represent unobserved random variables
Rewrite as Kalman

\[ v_t = y_t + r_t^y + \varepsilon_t^y \]
\[ p_t = y_t + r_t^p + \varepsilon_t^p \]
\[ y_t = u_t + r_t^y + \varepsilon_t^y \]

**Problem:** This mixes observable and unobserved variables
Because Linear and Gaussian, we can rewrite:

\[ v_t = y_t + r_t^y + \varepsilon_t^y \]
\[ p_t = y_t + r_t^p + \varepsilon_t^p \]
\[ u_t = y_t - r_t^y - \varepsilon_t^y \]
Rewrite as Kalman

\[ v_t = y_t + v_t^r + \epsilon_t^v \]
\[ p_t = y_t + p_t^r + \epsilon_t^p \]
\[ u_t = y_t - v_t^r - \epsilon_t^v \]

\[
\begin{bmatrix}
v_t \\
p_t \\
u_t
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & 0 & 1 \\
0 & 1 & 0 & 1 \\
0 & 0 & -1 & 1
\end{bmatrix}
\begin{bmatrix}
v_t^r \\
p_t^r \\
v_t^y \\
y_t
\end{bmatrix} +
\begin{bmatrix}
\epsilon_t^v \\
\epsilon_t^p \\
-\epsilon_t^v
\end{bmatrix}
\]
Rewrite as Kalman

\[
\begin{bmatrix}
  v_t \\
  p_t \\
  u_t
\end{bmatrix}
= 
\begin{bmatrix}
  1 & 0 & 0 & 1 \\
  0 & 1 & 0 & 1 \\
  0 & 0 & -1 & 1
\end{bmatrix}
\begin{bmatrix}
  r_t^v \\
  r_t^p \\
  r_t^y \\
  y_t
\end{bmatrix}
+ 
\begin{bmatrix}
  \varepsilon_t^v \\
  \varepsilon_t^p \\
  -\varepsilon_t^y
\end{bmatrix}
\]

THEY DIDN’T DO THIS, BUT COULD HAVE
Rewrite as Kalman

\[ y_t = u_t + r_t^y + \varepsilon_t^y \]

\[ v_t = (u_t + r_t^y + \varepsilon_t^y) + r_t^v + \varepsilon_t^v \]

\[ p_t = (u_t + r_t^y + \varepsilon_t^y) + r_t^p + \varepsilon_t^p \]

\[
\begin{bmatrix}
  v_t - u_t \\
p_t - u_t 
\end{bmatrix} =
\begin{bmatrix}
  1 & 0 & 1 \\
  0 & 1 & 1 
\end{bmatrix}
\begin{bmatrix}
  r_t^y \\
r_t^p 
\end{bmatrix} +
\begin{bmatrix}
  1 & 0 & 1 \\
  0 & 1 & 1 
\end{bmatrix}
\begin{bmatrix}
  \varepsilon_t^y \\
  \varepsilon_t^p \\
  \varepsilon_t^v 
\end{bmatrix}
\]

\[ z_t = Hr_t + H\varepsilon_t \]
Simple Kalman Filter

**Dynamics Model**

\[
\mathbf{r}_{t+1} = A\mathbf{r}_t + \mathbf{\eta}_t
\]

\[
A = \begin{bmatrix}
a^v & 0 & 0 \\
0 & a^p & 0 \\
0 & 0 & a^v
\end{bmatrix}
\]

\[
\mathbf{\eta}_t \sim N(0, Q)
\]

\[
Q = \begin{bmatrix}
q^v & 0 & 0 \\
0 & q^p & 0 \\
0 & 0 & q^v
\end{bmatrix}
\]

\[
\mathbf{z}_t = H\mathbf{r}_t + H\mathbf{\epsilon}_t
\]

\[
H = \begin{bmatrix}
1 & 0 & 1 \\
0 & 1 & 1
\end{bmatrix}
\]

\[
\mathbf{\eta}_t \sim N(0, R)
\]

\[
R = \mathbb{E}[(H\mathbf{\epsilon}_t)(H\mathbf{\epsilon}_t)^T] = \begin{pmatrix}
\sigma_v^2 + \sigma_u^2 & \sigma_v^2 \\
\sigma_v^2 & \sigma_p^2 + \sigma_u^2
\end{pmatrix}
\]
Experimental results

(a) Mean Localization Error – x
(b) Mean Localization Error – y

(a) Vision
(b) Proprioception

Modalities

Mean Error / cm

Pre–Adaptation
Post–Adaptation

Trial Number

Directional Error

Data
Bayesian Model
MLE Model

0 5 10 15 20 25 30

0 5 10 15

Vision
Modality
Proprioception
Results contd

Three tasks: Reach to target (right hand), left hand to visual left hand to right hand’s location
Learning Cues

• How do we get our understanding of what cues are available? We will explore this idea in the afternoon.