Why Perception is Hard…

Steve Scheding
ARC Centre of Excellence for Autonomous Systems
Disclaimer!

- This talk is intended as a discussion. Nothing will be solved!

- One viewpoint presented – it is not the only way to interpret the problem

- Language Warning
  - Some terms used in this talk, such as ‘Reasoning’ may not be used in the traditional manner
Motivation

- Outdoor natural terrain is subtle and ever changing
- True reliable autonomy requires machine perception beyond current demonstrated capabilities
- Shrub vs. Rock, Salt-pan vs. Lake
Motivation – Why Engineer a solution?

- Engineering is the science of compromise

- With perceptual systems however, it is not always clear what trade-offs are being made

- Perceptual systems are inherently application specific – which typically leads to custom solutions to new problems
Essential Characteristics of a General Theory?

• Decision making
  • Classification/Reclassification, target identification, planning etc.

• Flexible, task-specific capabilities
  • Cannot, in general, know relationship between sensors and decisions \textit{a priori} (even if we know the decisions)
  • Delay decisions until absolutely necessary
  • Implies sensor-centric representation and data retention

• Real-world is not a state-space
  • Physical qualities are functions of space
  • Requires approximation of infinite-dimensional spaces
Essential Characteristics

- Sensor-centric data storage (system-dependent)
- Probabilistic
- Reasoning corresponds to abstraction/interpretation (task-specific)
The Sensor Space

- Figuratively represent relationships between sensors
  - High-dimensional space with non-orthogonal (potentially non-linear) relationships between cues
  - Represents physical and statistical relationships between sensors (cf. the functional state)

Data Transformation

Application  
Location 4
A toy example

- Height-measuring device moving along a line
Functional Representations

Sensor Space

Sensor 1

Sensor 2

Physical Model

Likelihood Generator

Bayes Update

Performance Measures

\[ H(Z_y) = I(Z_y; X_y) + H(Z_y|X_y) \]

- Quantity of noise
- Information preservation

\[ P[x(y)|\tilde{Z}_1(v)\tilde{Z}_2(v)] = \frac{P[\tilde{Z}_1(v)|x(y),\tilde{Z}_2(v)]P[x(y)]}{P[\tilde{Z}_1(v)],P[\tilde{Z}_2(v)]} \]
Functional Forms

- Assumption: treat the world as a physical space which has measurable properties at each point – a (tensor) field
- Let:
  - \( y \) be the a location (position) in \( n \) dimensions
  - \( x \) be \( m \) ‘state’ values at each \( y \) at time \( k \)
    - Same as an \( m \times n \) space, but with explicit location
  - \( z \) be an observation of \( l \) measured ‘properties’

\[
y \in \mathbb{R}^n \quad x_k(y) \in \mathbb{R}^m \quad z_k(y) \in \mathbb{R}^l
\]

- Key Result:
\[
P \left[ x(y) | \alpha \right] = \delta \left[ x(y) - f_{\alpha} (y) \right]
\]
In the example

- Height-measuring device moving along a line

\[ x_k(y) \in \mathbb{R}^m \]

\[ z_k(y) \in \mathbb{R}^l \]

\[ y \in \mathbb{R}^n \]
Approximations

• Infinite-dimensional problem!
• What is desired is an appropriate summary or sufficient statistic
• In principle any approximation is valid
However...

• Practical constraints limit effective approximations
  1. Approximation effects and artifacts
  2. Parameter sensitivity (numerical stability)
  3. Representational compactness (storage)
  4. Manipulation costs and implications (processing)
  5. Decomposition complexity/re-sampling costs
Functional forms, parameter distributions and ‘arbitrary’ distributions

- Crucial distinctions
  - A *functional-form* represents a function as a set of parameters
  - A *parameter distribution* represents the probability distribution over those parameters
  - A *arbitrary distribution* refers to constructing distributions using functional approaches
In the example

- Height-measuring device moving along a line

\[ x(y_i) \in \mathbb{R} \]

\[ z(y_i) \in \mathbb{R} \]

\[ y \in \{y_i\}, i \in \mathbb{Z}^+ \]
Sensor Models

- Follows ‘traditional’ interpretation
  - Forward/inverse models
  - Bayes’ rule

- Functional models (Transformations of Ch4)
  - Domain Transformation (deterministic)
    \[ T_{\text{det}} : \hat{x}(y) \mapsto \hat{z}(v_i) = \int f_z(\hat{x}(y), y, v_i) dy \]
  - Measurement Uncertainty
    \[ P_{\text{value}}[z_i(v_i) | x(y)] = P_{\text{value}}[z_i(v_i) - T_{\text{det}}\{x(y)\}] \]
  - Domain (‘Spatial’) Uncertainty
    \[ P[z_i(v_i) | x(y)] = \int P_{\text{value}}[z_i(v_i) - T_{\text{det}}\{x(y')\}]P[y' | y] dy' \]
In the example...

\[ T_{\text{det}} : \hat{x}(y) \mapsto \hat{z}(v) = \int f_z[\hat{x}(y), y, v]dy \]

- No domain transformation (both measured at \( y_i \))

\[ P_{\text{value}}[z_i(v_l) | x(y)] = P_{\text{value}}[z_i(v_l) - T_{\text{det}} \{x(y)\}] \]

- Value uncertainty due to measurement error
- Assume Gaussian

\[ P[z_i(v_l) | x(y)] = \int P_{\text{value}}[z_i(v_l) - T_{\text{det}} \{x(y')\}]P[y' | y]dy' \]

- Domain uncertainty is HARD
Transformative ‘Reasoning’

Summary of Information (state)
\[ P[x(y)] = P[\alpha(w)] \]

Transformation Model

- Deterministic model
- Value (noise) model
- Domain uncertainty model

Performance Measures:
- Task-agnostic information
- Task-specific information
- Task-specific cost functions
- On-line entropy contribution
Arbitrary Transformations

• The ‘Feature-space’

• Transformations

• Task-centric Approaches:
  • Correlation/Convolution (Minkoff 2002)
  • Statistical Communications Theory (Middleton 1996)
  • Regression and discriminative classifiers (Mackay 1998)
  • Generative Statistical Models (Ng/Jordan 2002)
Simplified Transformations

Transformation

- Model: \( P(\mathbf{s}(\mathbf{q}) | \mathbf{x}(\mathbf{y})) \)
- Deterministic model
- Value (noise) model
- Domain uncertainty model

Kernel: \( f_\mathbf{s}(\mathbf{x}(\mathbf{y}), \mathbf{y}, \mathbf{s}) \)

\( T_{\text{det}} \)

\( P_v[\mathbf{s}(\mathbf{q}) | T_{\text{det}}(\mathbf{x}(\mathbf{y}))] \)

\( \mathcal{P}_y[\mathbf{s}(\mathbf{q}) | T_{\text{det}}(\mathbf{x}(\mathbf{y}))] \)

\( \mathcal{P}_v[\mathbf{s}(\mathbf{q}) | T_{\text{det}}(\mathbf{x}(\mathbf{y}))] P[\mathbf{y}'|\mathbf{y}] \, d\mathbf{y}' \)

\( \hat{\mathbf{x}}(\mathbf{y}) \)
What Have We Learnt So Far?

- Infinite dimensional problem – Must use approximations to be tractable (kernel methods)
- The chosen approximation has multiple performance measures, and may need to be different depending on the downstream application
- Features extracted via transformation of data. Can be thought of as Lossy Compression
- The task/application determines the appropriate transformation (the one that is optimal for task completion)
Open Questions...

• What are the appropriate domains? Usually the sensor domain is given, and the others manually designed.

• Performance measures – how do we tell if our perception system is actually working as expected. How do we know what to expect?
A Concrete Example
UGV Operation
Intermediate Representation

max possible slope : Worst case scenario Cost
Transformed to Feature Space
Higher Order Representation
Transformed to Feature Space
Perception Failure 😞