Overview
As part of the Bernstein Focus in NeuroTechnology initiative in Frankfurt, a trans-disciplinary project involving neuroscience, psychology, computer science and engineering, we have developed engineering platforms, simulation tools, and application case studies in cognitive vision following systems engineering principles.

System Engineering Methodology
Our engineering platform effort lies on the modeling, implementation and validation views of the system. Mainly on core data structure support for algorithms, distributed run-time execution of modules, enhanced debugging of system pipeline execution states, contextual modeling and computer graphics simulation of data, machine learning, optimization, and performance characterization of systems [1, 2, 3]. Given the rapidly emerging open-source efforts in Machine Learning and AI, the platform effort is planned to be shifted to a startup in engineering platforms.

Inspirations from Cognitive Architecture
- Our effort is to realize the proposed cognitive architecture [1], with its major components: hypothesis generators, deliberation and knowledge update.
- Visual intelligence is seen as context and task sensitive indexing followed by detailed estimation or deliberation (Indexing involves decomposition of input data into quasi invariant features as suggested in [4]). The indexing sub-modalities (e.g. color, texture, motion, illumination, etc.) are complementary in nature.
- Detailed state estimation can be implemented by a variety of schemes - distributed fusion, belief propagation, markov-chain monte carlo methods, or deliberation and reasoning [1, 5, 6, 7].
- Model based system design thus involves translation of appropriate scene priors and task requirements to perform quick hypothesis generation and fusion.

Case studies
In this presentation, anomaly detection from video for an automotive application is used as a case study to demonstrate our engineering workflow and platform capabilities [8]. We highlight how to translate user requirements to graphical models for contexts that are in turn translated to approximate inference engines. We also illustrate how our platform is used for evaluating the performance of the inference engine on real as well as synthetic data.

Evaluation
Based on the implementation, large-scale experiments have been conducted and analyzed.

Table: Evaluation of system-estimation without temporal information.
<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>StdDev</th>
<th>Median</th>
<th>95% CI</th>
<th>2.5% CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>D/N</td>
<td>99.3%</td>
<td>11.0%</td>
<td>99.1%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D/F</td>
<td>99.1%</td>
<td>9.1%</td>
<td>99.0%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D/P</td>
<td>96.9%</td>
<td>7.7%</td>
<td>97.0%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MSE</td>
<td>54.9%</td>
<td>3.3%</td>
<td>54.5%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Case study: brake light detection in automotive video sequences

- **User view**
  - Task: brake light transition detection system on monocular video sequences. Detect brake light state changes including confidence level of detections.
  - Performance: Near-human level performance with self-diagnostics. Real-time implementation on embedded platforms. Academic prototype’s focus is on demonstration on methodology.

- **Modeler view**
  - The modeler postulates the low entropy factors in the OOBN, what factors require invariance (i.e. what sub-modalities are relevant for task) and chooses invariant modules.

- **Implementer view**
  - As described in [1], we implemented a cognitive system that consists of multimodal quasi-invariant filters that map to the variables identified in the modeling phase. A memory storage contains prior hypotheses on the world and is indexed by context and current state estimates, selecting and tuning parameters of the estimators.

Simulation
Simulation is utilized during the modeling stage to populate prior memory representations. C.T.P. – 3D description → Simulation → Memory
These memory representations are used by the inference engines of the realtime system.

Integration and Fusion of perspectives
The systems engineering process that links explicitly context, task, performance specs to designs allows for explainability, context sensitivity.

Results
Principled analysis and modeling of involved variables allows for the selection of suited estimators for a given task context. Priming of estimators from given memory representations allows to set parameters that adapt to a given context.

Acknowledgments
This work was funded by the German Federal Ministry of Education and Research (BMBF), project 01GQ0884 and 01GQ0884I (Bernstein Focus: Neurotechnology Frankfurt).

Bibliography

Figure: Graphical model for the task of brake-light detection. The World plate contains latent variables on which the observed variables in Obs are conditioned at time t.

Figure: Two frames from a braking event (top: time t, bottom: t+1). First column: input images, second column: map of θ, third column: magnified transition estimations (T-ON transition from on to off, T-No transition detected.)

Figure: Simulation for expected flow given scene geometry and velocity.