AI rEvolution:

Author: Prof. Dr. Visvanathan Ramesh (September 2018)

Background: Artificial Intelligence (AI), intelligent software that can perform tasks that were once considered to be in the realm of human abilities, are now impacting all walks of society. The AI field had its origins in 1955 when Prof. John McCarthy organized a group to focus on the idea of building thinking machines and proposed a summer seminar project. The proposal stated the hypothesis and formal goal of AI: "every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. An attempt will be made to find how to make machines use language, form abstractions and concepts, solve kinds of problems now reserved for humans, and improve themselves." Today, terms such as Artificial Narrow Intelligence (ANI), Artificial General Intelligence (AGI), and Artificial Super Intelligence (ASI), are used to describe systems with different levels of functionality. ANI for instance is a system designed to perform a narrow function (e.g. a conversational assistant for supporting a customer in a narrow domain). AGI is an attempt to revisit Prof. McCarthy's original goal of AI to design a fundamental substrate that can emulate human intelligence. ASI is a term that is used to describe a system that is able to exceed human level intelligence. Today, AI is enabling a wide spectrum of applications such as: autonomous self-driving cars, medical diagnosis, language translation, personal conversational assistants, search, robotics, etc. AI systems are beating humans at strategy games such as Go.

What is an AI system? Before we delve more into the state of AI development and its relationship to our own experience we provide a conceptual view of an AI system. Conceptually, an AI system, embedded in a world, can be seen as one that translates sensed data and answer queries regarding the state of the world. The output can be seen as a compound decision, i.e. a conjunction of simple atomic tests that may be applied in sequence, parallel, or in combination. The AI system is thus a computer program, a complex data flow graph translating input data to output states. The output states may be refined further via iteration in conjunction with action to acquire more data to resolve ambiguities. While doing this, the system may or may not have an explicit model of the world or of its own computational process. And, the system may or may not be able to provide an explanation for why a given answer is the best for a given input data. The architecture for the program – i.e. computational structure is specified by the designer. Complementary computational representations - a decision tree or deep neural net are used in modern practice. Decision trees involve a sequence of tests that are viewable by humans. Neural nets are not inspectable but it's end to end behavior can be visualized using tools. Key questions that AI researchers have pondering about include: a) Where do the atomic tests come from? b) How can one order these atomic tests to achieve a given task in a given context? c) How can we search over alternative program architectures? d) How can the AI system diagnose itself that it is unable to do a given computation? e) How can we certify that a given AI system will perform within safe bounds in a given world? F) How can the AI system self-evolve to improve its competences and reapply its learned representation to new tasks in new contexts? A lot of progress has been made in developing learning systems but key stumbling blocks remain with respect questions (d), (e), and (f).

AI rEvolution is an Evolution: The 'AI rEvolution' is, indeed, an evolution over the last 60 years of AI research. This evolution has come in three waves (Source: DARPA 2017) – a first wave wherein human expert knowledge in the form of rules were used to enable design of expert systems. Rules were found to be brittle as they don't enable ease of handling of ambiguity and uncertainty in real-world settings. This led to the second wave of AI that used statistical machine learning methods wherein the engineer provides a learning algorithm a network architecture, large amounts of training data along with expected output labels, and a cost function to optimize. The optimization algorithm produces network parameters that minimizes the specified loss. Modern engineering practice uses deep neural networks to perform a variety of narrow AI tasks. The explosion of the second wave over the last decade is enabled by significant advances in computing power, training algorithms, ubiquitous computer networks and more importantly huge volumes of training data from which to learn from. The technical limitations of current practices in design, implementation and validation of an AI system include:

- The need for large amounts of labeled training data,
- Lack of ability of the system to explain why it arrived at its results,
- Inability to generalize –i.e. transfer learned experience from one setting to another setting, and
- Systemic bias that is due to the sampling process in the training data.

Systems Engineering Example in Practice (second wave): It was the year 1999. I was a technical manager leading the real-time imaging program at Siemens Corporate Research in Princeton, New Jersey. At this stage, our small team had Phd students and a few core scientific staff members. I was juggling my time between being a technical contributor, program and project manager and interfacing with customers to set research priorities that can help increase their product value for end-users. One of our major customers had a challenging project involving a video-based system that could monitor about 2 kilometers of highway shoulder-lanes and alert a control center if there are any pedestrians, stopped and/or slow vehicles in the lane. The system was to perform with: a) near-perfect (100%) detection of events with very low false positives, b) under all weather conditions except for heavy snow and rain and c) provide self-diagnosis about the unavailability of the system in the event of extreme input conditions involving very poor contrast due to events such as direct sunlight on the camera lens, severe glare, and other effects. The added computational constraint was that the system will perform its functions in near real-time with up to four camera streams per computer. The overall objective of such a system was to allow the shoulder lane to be opened up for traffic in the event of road congestion. A system performing such a task in the modern days could perhaps be labeled as a ANI system because the complexity of the task is at the level of a child performing a task in a narrow domain. In summary, the:

- System Context involves: traffic scenarios,
- Task involves event monitoring in the outdoors, and the system tasks are at the level of what a child can achieve (but perhaps not for extended periods as the child may get bored over longer time), and
- Performance requirements requiring a system with low computational complexity while providing high reliability and safety (i.e. with ability to self-diagnose that the system is functioning within safe boundaries).

My job as a systems engineer was to take these requirements and translate them to a design that addresses these needs. Fortunately, since the late 80's and early 90's, I was influenced by mentors, Professor(s) Robert Haralick and Thomas Binford, who were examining how vision systems can be designed systematically and analyzed in a principled way. Michael Greiffenhagen, one of my Phd students at that time, and we were exploring how to systematically model the contexts and goals of an intelligent vision system, map these requirements to the design of the system architecture (i.e. its components, and composition), and analyze its behavior in the application context. The focus of our work was on how systems engineering principles may be applied to a real-world engineering problem that had computational constraints (i.e. the system had to be real-time) as well as accuracy requirements. The essence of our philosophy was that: “Real-time Vision is possible via steps involving a sequence of statistical tests – i.e. indexing functions (a parametrized decision tree) followed by estimation”. The key aspect of our design methodology was that these statistical tests are context-sensitive (i.e. application dependent) and they involve explicit use of expectations – regularities, i.e. strong constraints that help in reducing the computational complexity, as well as invariance requirements, i.e. the system must be able to ignore variations that are not relevant for the goal. In order to be able to characterize the performance of the designed system, we postulated that the chain of statistical tests chosen must allow for model-based control-theoretic analysis that allows propagation of probability distributions of input to output so that one can quantify the overall performance of the system as a function of contextual model parameters and system parameters. These principles for systems engineering and performance modeling had been laid out in the early 90’s in a series of dissertations, including my own, under supervision of my mentors.

Meta-analysis of AI system Performance: In our work, performance modeling is essentially a meta-level analysis wherein a given system behavior is analyzed by propagating input uncertainties through the sequence of computations performed by the system to derive the output uncertainties. The philosophical viewpoint is that:

- The interpretation system is a cascade of algorithms. Each algorithm is an “estimator” that estimates quantities from its input. The system can be treated as a compound estimator.
- Characterizing deviations of the estimate from true values is “Performance Characterization”. Performance characterization is carried out through either analytical or numerical error propagation.
- Total system performance is a function of the components, their ideal input models and perturbation models, their tuning constants, and the architecture of the overall system.
- An application context essentially is specified through a probability distribution on the input space (i.e the restrictions that are placed in scene (including the sensor) properties, environmental conditions, object dynamics, etc.)
- Application context specific performance evaluation is the derivation of specific online performance measures as a function of a given system configuration and parameters.

Since the 90’s we have systematically analyzed properties of human expert driven model-based designs as well as second wave data-driven machine learning algorithms. Our design methodology and computational architecture also has relationship to efforts in the late 90’s by Prof. Donald Geman and his collaborators to address real-time visual pattern recognition through a sequence of invariance tests (i.e. decision tree) in context.

In highway monitoring system example, articulated above, the regularity in the context was that traffic, to a large extent, followed specific rules, the geometry and motion of objects were constrained, and the invariance requirement was that the computed indexing functions should be invariant to illumination and weather. These properties were exploited in our model-based design. A critical element of success in this project was that the engineer responsible had systematically collected video data for various scenarios and had devised a rigorous experimental plan for evaluation. Even though our model-based design methodology addresses learning of system parameters with small amounts of data, having a comprehensive sampling of the scenarios allowed us to validate our methodology. The resulting system had a performance of 98% detection of true events, with contrast levels between pedestrian and background of only a few gray values, and corresponding false positive rate of 1 per camera per day.

Furthermore, the principles outlined in this example were the basis for a family of systems and implementations involving safety and security of subways, highways, tunnels, airports, robotic assembly, visual inspection systems, etc. Note that this achievement in the year 1999 was based on: 1) a computer that is roughly 10000 times worse in performance than current computers, 2) a design methodology that uses context-sensitive modeling, is able to be tuned with small amounts of examples, and enabled safety by design.

Safety of intelligent systems require the ability for the system to be able to predict its limitations in a given contextual instance. In the systems engineering methodology for computer vision, that we developed in the 1990's, the performance of a system is modeled and quantified through a "white box" (analytical) or a "black box" approach. The white box approach provides a complete error analysis of an algorithm via propagation of uncertainty models through various component steps in the algorithm. Using a model of complete architecture of the system, each component is treated as an estimator whose deterministic and stochastic characteristics are derived as a function of inputs. The system is viewed as a compound estimator whose performance characteristics are derived as a function of individual component estimators. In black box analysis there is no modeling of the individual components, their error characteristics, or their interactions. Rather, the system performance is viewed as a function of its inputs and its deviation from a user specified objective. Thus, in order to characterize system behavior one specifies criterion functions (performance measures) that are to be measured through a carefully devised experimental protocol. Empirical evaluations involve the estimation of how the criterion functions change as a function of design choices and tuning parameters (e.g. degree or rate of adaptation) of the system.

Modern practice using deep neural networks, inspired by the structure of the brain, may be seen as a sequence (layered) of computational transformations with homogeneity in structure of computational blocks. Humans gather large amounts of data, annotate them with labels describing desired output, hypothesize network structures, provide optimization criterion, and use optimization tools to estimate the network parameters that perform best on the data. The core assumption in this framework is that the statistics of the data gathered in the future will be the same as the data gathered in the past. In contrast, a fundamental difference between modern practice in second wave AI system design and our methodology is that our methodology approaches the design, implementation and validation process holistically. The systems engineering methodology we follow separates the user, the modeler, implementation and validation viewpoints. Moreover, it separates the formal specification of world models and generative processes for data, the tasks and the computational pipeline that performs the task. There is an explicit map between contextual models, tasks, performance requirements to algorithmic structures that allow for the design to have properties such as transparency of assumptions, explainability of design, modularity, compositionality, scalability, adaptivity, extensibility, performance characterization etc.

Ongoing Efforts: The present focus of our research in Frankfurt is the integration of our past experience towards a trans-disciplinary approach incorporating modern AI, machine learning, applied mathematics and statistics, systems engineering, neuroscience, psychology and cognitive science. The core challenge we face is one of '**scaling**' of system complexity to allow the system to perform a wide range of tasks in diverse contexts with human-level performance or beyond. For this, we take inspiration from the human brain, as it is an evolved system with a flexible learning architecture designed by nature to solve a wide range of tasks in a class of environments that enhances the survival and reproduction of humans. Our interactions with neuroscientists, psychologists and philosophers in the past decade have shown that, at a high-level, our architectural designs inspired from systems engineering principles have close parallels to computational models of the visual brain. Our design approach exploiting invariance and regularities for hypotheses generation followed by deliberation can be seen as an analogue of dual-system models in psychology proposed by Nobel prize winner Prof. Daniel Kahneman. An analogue can also be made to that of dynamic visual architecture of Prof. Christoph von der Malsburg who advocates that the brain performs massively parallel and perform feed-forward decomposition of input visual signal into constituent modalities (e.g. color, motion, texture, shadow, reflection, contours, etc.) thus allowing for efficient indexing into a rich memory structure. Generated hypotheses can then be refined via a dynamic, recurrent process to converge to an interpretation. While both engineering and brain science views of the architectures agree at this higher level, our view is that 'scaling' can be achieved mainly through advances in cognitive architectures, AI systems theory, and software platforms that facilitate rapid design and validation of these systems. Moreover, there is a need for the design of a homogeneous visual architecture with self-learning ability that supports context sensitivity, explainability, thinking and reasoning like humans, and can learn from very few examples. Indeed, the US Department of Defense and Advanced Research Agency (DARPA) is investing in the third wave AI research that addresses these key gaps in AI.

Our research over the last years, as part of the Bernstein Focus in Neuro Technology and EU project AEROBI, is to advance our understanding of systems engineering for visual intelligent systems and incorporate them in automation tools that enable ease of design, validation and certification of AI systems. My Phd student, Subbu Veerasavarappu, addresses the role of computer graphics simulation and modeling in the design and analysis of cognitive vision systems. Simulations can play a dominant role in evaluating the behavior of alternative implementations and systematic performance evaluation and validation. Moreover, they can synthesize data for the data-hungry deep learning methods. In our work in 'simulation for cognitive vision' we address a range of questions such as: a) What is the impact on modern computer graphics engine rendering fidelity on machine learning system performance? b) How much can we reduce the amount of real data required for machine learning through use of simulated data? c) How can we bridge the gap between the deviation between simulated data statistics and real-world statistics so as to facilitate better transfer learning?. In our applied systems work, addressed by Phd students Rudra Hota, Tobias Weis, and Martin Mundt, we have combined model-based and data-driven machine-learning principles to demonstrate our cognitive architecture designs wherein expectation models in context are used for estimating world state, monitor behaviors and identify anomalies. These application examples include: video surveillance/security, brake-light on/off detection in automotive, fine crack/defect classification in bridge infrastructure, and behavior monitoring in scientific applications.

Future Perspectives: The main challenge in building a holistic theory of design of intelligent systems is that it involves harmonizing the model-based engineering and modern machine learning perspectives via the joint development of a body of knowledge leading to a comprehensive encyclopedia involving the space of contextual models, tasks and performance criterions with maps to the appropriate computational structures. Open source platforms and open data initiatives along with integrated research and development networks are accelerating the development of AI. Indeed, countries, companies, are setting up AI eco-systems to address transdisciplinary scientific research, engineering platforms development, and successful integration and transition to industry and society. Key technical gaps are in: a) platforms that can enable creation of safe and explainable AI systems, b) training, mentoring of systems thinkers and c) establishment of integrated eco-systems for rapid AI innovations. AI is ultimately a discipline that will be shaped by creativity, collaboration, communication, and critical thinking and problem solving abilities that are uniquely part of humans. The same four C's along with 'systems thinking' are the core elements that we as educators must teach the next generation in order to be able to find their roles in the pervasive AI world of the future!

References:

[Ramesh1995] V. Ramesh, Performance Characterization of Image Understanding Algorithms, Ph.D Dissertation, Department of Electrical Engineering, University of Washington, Seattle, 1995.

[Thacker2008] N. A. Thacker, A. F. Clark, J. L. Barron, J. R. Beveridge, P. Courtney, W. R. Crum, V. Ramesh, and C. Clark. Performance characterization in computer vision: A guide to best practices. Computer vision and image understanding, 109(3):305–334, 2008.

[Binford1989] T. Binford, et al., Bayesian Inference in Model-Based Machine Vision, Uncertainty in AI (3), 1989, Levitt, Kanal and Lemmer (Eds.), North Holland.

[Mann1996] W. Mann, 3D Object Interpretation from monocular images, Phd Dissertation, Stanford University, 1996.

[ZhuMumford2006] S. C. Zhu and D. Mumford, A Stochastic Grammar of Images, Foundations and Trends in Computer Graphics and Vision, Vol. 2, No. 4 (2006) 259–362

[Greiffenhagen2001] M. Greiffenhagen et al, Design, Analysis and Engineering of Video Monitoring Systems: A case study, in Proc of IEEE, Special Issue in Video Surveillance, Nov. 2001.

[Bengio2009] Y. Bengio, Learning Deep Architectures for AI, Foundations and Trends in Machine Learning Vol. 2, No. 1 (2009) 1–127.

[Malsburg2012] C. Von der Malsburg. A Vision Architecture Based on Fiber Bundles . Front. Comput. Neurosci. Conference Abstract: Bernstein Conference 2012.

[PoggioBook2016] T. Poggio and F. Anselmi, Visual Cortex and Invariance, MIT Press, 2016.

[Kahneman] Daniel Kahneman,. Thinking, Fast and Slow. Farrar Straus and Giroux, 2011.