Role of Neural Networks for Avionics

Christopher Bowman

Intelligent Reasoning Systems, Incorporated (IRSI)
8503 Sykland Dr., Longmont, CO 80503
(303)652-8102

Mark DeYong and Tom Eskridge

IRSI P.O. Box 300001, Dept. 3ARP Las Cruces, NM 88003
(505)646-7683

ABSTRACT

Neural Network (NN) architectures provide a thousand-fold speed-up in computational power per watt along with the flexibility to learn/adapt so as to reduce software life-cycle costs. Thus NN's are poised to provide a key supporting role to meet the avionics upgrade challenge for affordable improved mission capability especially near hardware where flexible and powerful smart processing is needed. This paper summarizes the trends for air combat and the resulting avionics needs. A paradigm for information fusion and response management is then described from which viewpoint the role for NN's as a complementary technology in meeting these avionics challenges is explained along with the key obstacles for NN's.

Keywords: neural networks (NN), avionics, pulse stream NN, data fusion tree, duality, neural control, multispectral NN image processing

1. Introduction

Electronic computer hardware (HW) capability per dollar cost has been doubling every two years and is expected to continue. Software (SW) capability in executable lines of code per dollar has not been improving at even near that rate. Thus DOD SW costs have far surpassed HW costs (e.g. $25 billion vs $5 billion in 1991). The growth in neural networks (NN's) has been spurred by both of these trends. First computer chip HW capabilities have enabled extensively parallel NN processing to provide timely solutions. NN architectures have capitalized on the digital and analog electronic chip technologies to define even more efficient architectures as measured in operations or connections per second per watt. Second, NN's use their hardware advantage to reduce SW programming costs. The NN architecture embeds data and goal driven learning mechanisms to solve pattern recognition and control problems without requiring the user to discover these procedures. Thus, NN's are most advantageous in rapidly changing uncertain environments where robustness and high speed per watt are needed such as avionics.

NN's have a complementary role with other computing architectures. Figure 1-1 plots the problem solving flexibility of the various architectures against their computational speed per watt. Their computing performance increases with the increase in parallelism as the HW moves from electronic digital to analog to optical. In addition their flexibility tends to decrease accordingly albeit for different reasons including coding costs, restricted parallelization levels, scalability, lack of readily available learning paradigms, lack of user tools, common interface limitations, and HW immaturity. All of these computing architectures are moving up and to the right as they mature. As the amount of information, complexity, uncertainty, and computational capacity grows, systems designers identify the cost effective partitioning of each problem for each computing type. For example well-characterized subproblems are solved with conventional HW/SW using model-based algorithms (e.g. logic models, Bayesian, optimal search.). More hearistic subproblems are solved with knowledge-based expert system HW/SW combining data and procedures to search heuristically for a solution. Unmodeled subproblems are solved with high speed NN HW/SW learning from data to find patterns that achieve desired goals. The key to success will be the hybrid architecture within which these mutually supporting computing HW/SW elements can be cost effectively integrated.

0-8194-1925-7/95/$6.00

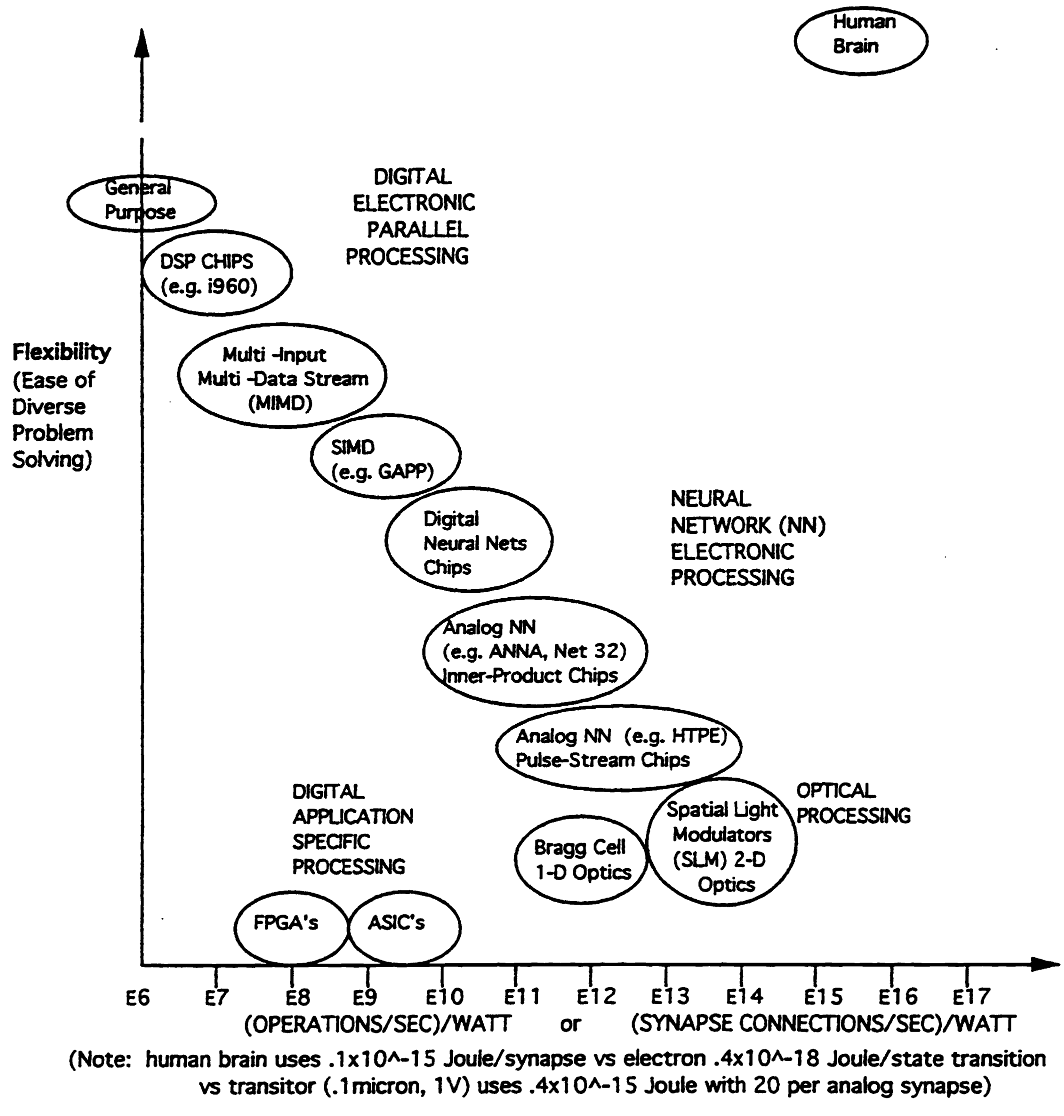

Figure 1-1. Pulse-Stream Neural Nets Have Highest Potential To Break Out Of Trend Of Flexibility Decrease and Software Cost Increase With Speed/Watt Growth

2. Avionics Needs and the Role for Neural Networks

2.1. 21st Century Air Combat Trends and Avionics Needs

The emphasis over the next two decades will be on affordable upgrades to existing military aircraft. These upgrades will leverage commercial advances in computing (HW & SW), and wireless communications as well as military advances in long range smart weapons and responsive surveillance (e.g. JSTARS, UAV's, space-based). Our worldwide competitors will also be capitalizing on these advances for enhancements in their military capability. To meet the resulting accelerated tempo of conflict, forces will be organized around information (instead of platform/weapon systems). Sensors and shooters will be digitally netted busting thru "stovepipe" C^4I systems. This will engender flatter unified command structures and shooter-level situational awareness and response strategy. C^4I architectures for Close Air Support (CAS) (via ABCCC, ASOC's, and FAC's) and for Theater Air Defense (TAD) (via AWACS, CRC's, GBR's, and SAM's) are on track for the 21st century. However our ability to efficiently adapt to the changing Air Interdiction (AI) and Offensive Counter Air (OCA) situations (after aircraft are airborne) is constrained. Theater wide JSTARS, manned, UAV, and space-based sensors promise near real-time near-perfect ground situational awareness. Future emphasis is needed on air operations execution especially inflight target handover for AI and OCA.

The projected air operations organization is shown in Fig. 2-1 where the Control and Reporting Center (CRC) commands the TAD missions and the Local Attack Control (LAC) center commands CAS, AI and OCA missions. The LAC features are similar to CRC for air operatons namely: significant two-way communication connectivity (space, air, ground); significant information fusion computing power (awareness and target prioritization); C^2 authority to rapidly retarget AI and OCA sorties inflight; capability to replan UAVs and other ground surveillance assets; enable airborne extension (e.g., JSTARS) with ABCCC functions.

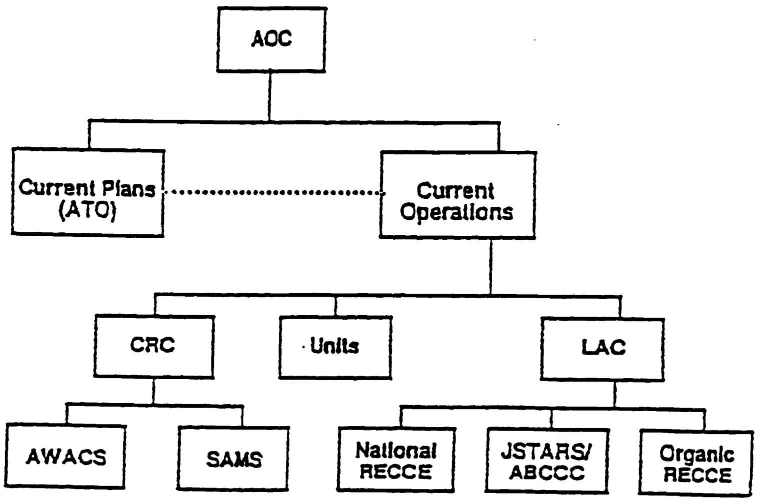

Figure 2-1. Future Emphasis Is Needed On Air Operations Execution Especially Air To Ground Target Handoff

As a result, long-range rapid retargeting precision strike capabilities will demand automated robust information fusion (IF) and resource management (RM) to include management of sensors, communications, countermeasures, crew interfaces, platforms, and their processing. The information-driven military system costs will be driven by the avionics SW timing and robustness requirements. To reduce these life cycle costs, an avionics SW architecture and a systems engineering process stst need to be developed within a common underlying paradigm.

The avionics assists the aircrew by providing robust and affordable situation awareness and response operations. Avionics is needed to detect, locate, ID, and rapidly handover time critical targets. To do this the avionics SW will internet to cooperating platforms and integrate stand-off off-board information to extend the situation awareness envelope and coordinate the response for improved weapon system performance. The avionics will provide automated response, and decision aids, as personalized by pilot, to stay inside the opponent's response cycle. To meet cost constraints these avionics SW upgrades will be developed within an open, layered, object-oriented architecture.

With this avionics architecture the cost savings will increase with level of SW reuse achievable (from domain abstractions to mechanisms to architectural layers). The (horizontal) architectural layer interface standards will be developed to maximize SW reuse to include physical (bus and memory), data link/data base, network/data base manager, presentation/formatting, and applications layers. The (vertical) functional interface paradigm will enable user partitioning to divide and conquer so as to achieve the knee-of-the-curve in performance versus cost/complexity. The object-oriented class diagrams support both the horizontal and vertical portions of the SW development matrix. In summary, stet are needed to acquire, condense, distribute, and apply information in a timely, affordable, and robust manner.

2.2. A Functional Paradigm For Avionics Development

Avionics SW can be divided into two interacting processes, data fusion (DF) and resource management (RM). Data fusion is the process dealing with the association of data and estimation of aggregate and object kinematics, attributes, and identity to achieve timely and refined products providing relevant assessments and projections of the situation. Resource management is the process utilizing the data fusion product to assign and control the available platform, countermeasure, sensor, and processing resources in support of the mission objectives. These two processes are duals in the sense that a formulation and solution for one (e.g. data fusion) can be used to formulate and solve the other (e.g. resource management) using the dual variable interpretations (e.g. the duals for data association and estimation are response planning and control, respectively)[1]. Thus the data fusion tree "fan-in" paradigm[2] used to define the batching of data for fusion node association and estimation yields a dual "fan-out" paradigm to formulate the batching of responses for planning and control. An example of a distributed DF and RM avionics tree is shown in Fig. 2-2. Each node in the tree is tailored within the fusion tree dual node paradigms to maximize its capability within the cost constraints.

2.3. Avionics Challenges and the Role for Neural Nets

Avionics performance evaluation is based upon meeting affordability and mission capability requirements. Performance evaluation provides feedback for design optimization on every level of the avionics system design process (i.e. 1) system 2) component 3) module/node 4) detailed design levels). A recommended avionics performance evaluation criterion is to maximize the confidence in meeting the requirements where increasingly detailed requirements are derived for each level of the interactive design process. The avionics SW system level requirements are of two types, namely, life cycle costs (development, P^3I/maintenance) and mission capability (information fusion, resource management). Life-cycle cost reduction is being jointly addressed by the open, layered, object-oriented architecture and testbed development at Wright Labs (WL), NAWC, and other labs. A top-level evaluation of the state-of-the-art, challenges, and desired capabilities for future avionics SW is given in Fig. 2-3.

Avionics upgrades have a long way to go to meet the 21st century needs described above. For example, the F-18 is the only current operational fighter with even limited data fusion. However, the Talon programs are demonstrating Real-Time Intelligence in the Cockpit (RTIC) capabilities. Also, real-time demos and flight tests with DF and RM on a C-130 and followed by an F-16 are planned under the Expanded Situation Awareness Insertion (ESAI) WL program. Real-time symmetric multiprocessors (RTSMP) are replacing current processors achieving 100 times the thruput per watt. For example, a 4 CPU RTSMP operates at 512 MIPS (80 MFLOPS) with 64 Mbytes of memory and a 4 Gbytes removeable disk at approximately 100 watts.

Neural nets provide over a thousand-fold improvement over the RTSMP in computational speed per watt with reduced SW costs due to their adaptivity. Thus the future role for NN's is in enhancing operational capabilities within size, weight, and power constraints especially where high speed and adaptivity are required. This includes SAR-to-IR-to-VIS pixel level fusion , automatic target recognition (ATR) with spatial/temporal reasoning, goal driven automated response, and nonlinear adaptive control. Mathematical scoring and algorithmic searches are preferable for avionics applications where the problem is more well-defined (e.g. sensor data association). For applications with a continual need for knowledge acquisition on partially modeled processes and where decisions are conditional (e.g. situation and threat assessment, top level response management) expert systems (AI)

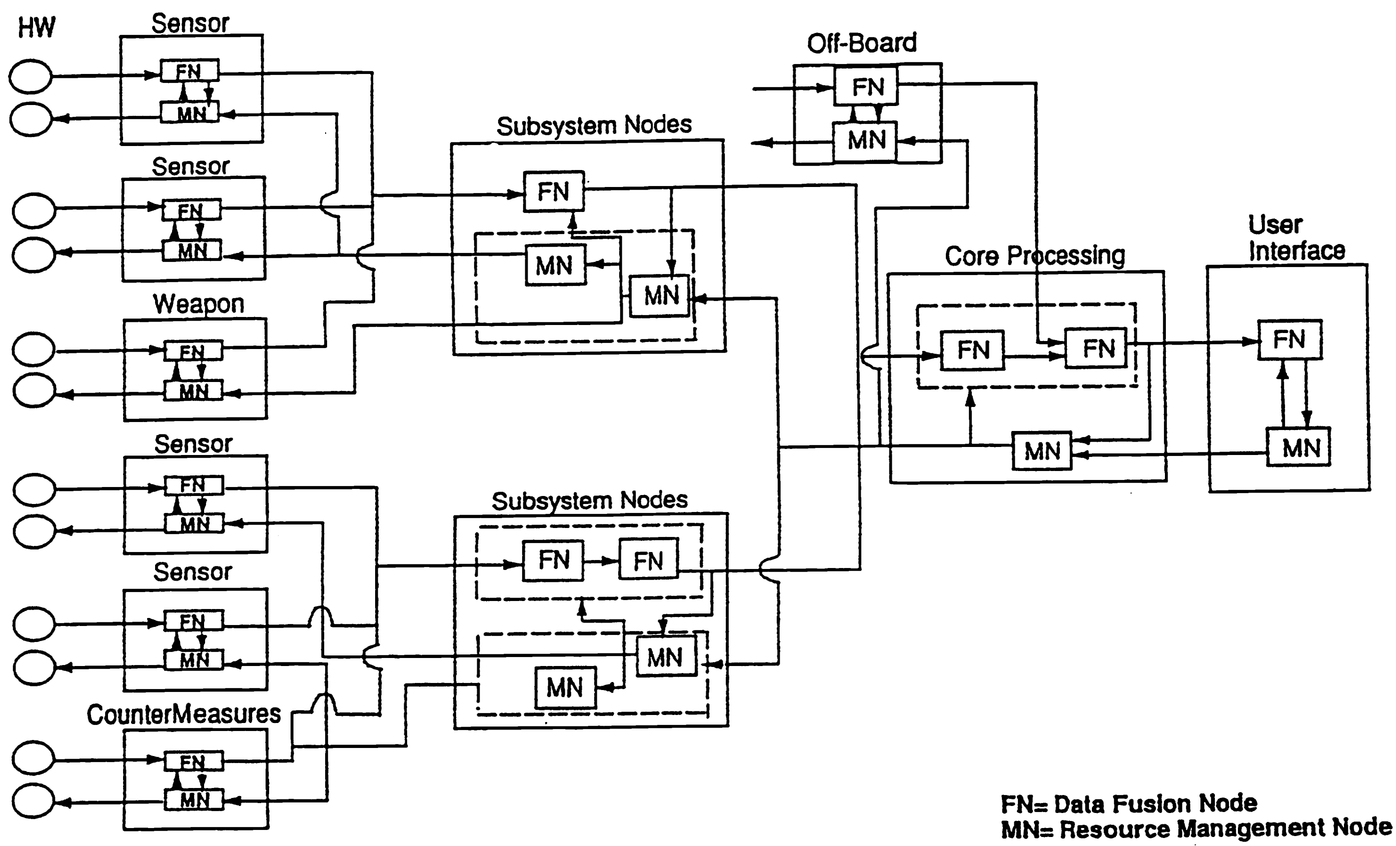

Figure 2-2. Fusion and Management Nodes Are Interlaced To Provide Local Feedback

Functions	Subfunctions	State-of-the-Art	Current Challenges	Desired Capabilities
Level I: Object Refinement (Application Layer)				
1. IF Trees	- On-board - Off-board (one-way) - Internetted (shared)	Sequential Lab SW RTIC SW Capability Limited None Operational on AF Shooters	No Standard Architecture Integrate Stovepipe Trees Distributed Consistancy	Multi-Frame Multi-Spectral Adaptive IF Trees IF Parallelization
2. IF Tree Nodes	- Common Referencing - Data Association - Object Estimation - Kinematics - Attributes - ID (IFF/class/type)	Kalman Misalignment Estimation Bayesian Scoring; Search Schemes: Lagrangian Relax , JVC, Munkres Adaptive KF's, PDAF, Passive Ranging RAD, MOP, RF, PRI, PW, Signature Edge Extraction, Wavelet Match	Robust Internetted Nav A Priori/Late Data Predetection Fusion Maneuvering Angles-Only Track Random Hoppers, Scoring Feature Level Fusion	Common Shared View Mgmt of Uncertainty Model-Based Vision BistaticTargeting SAR-to-IR-to-VIS Pixel-Level Fusion
LevelII/III: Sit/Threat Refinement (Application Layer)				
1. IF Trees	- Weapon Systems/IADS - Internetted	Order of Battle Refinement Parallel AI, Time Critical Processing	Doctrines Ill-Defined Lack of Architecture	Non-Monotonic Reasoning Distributed Info Sharing
2. IF Tree Nodes	- Common Referencing - Object Aggregation - Event Est./ Prediction	Blackboards & Symbolic Reasoning Script-Based Templating Knowledge-Based & Expert Systems	Brittle A Priori Expectation-Based Reasoning; Adaptivity	Cognitive Model Standards Automated Learning Spatial/Temporal Reasoning
Level IV: Mission/Process Mgmt (Application Layer)				
1. Resource Mgmt Trees	- Onboard - Internetted (shared)	Single Resource Unique Mgmt None Operational on AF Shooters	Lack of Architecture Affordable BW Links	Divide & Conquer Architecture Shooter Synchronization
2. RM Tree Nodes	- Common Referencing - Response Planning - Control/Tasking	Data/Event Driven Rule Based Simple Mode Mgmt Single Resource Unique Controls	Prioritized. Request Driven Lack of Resource Modes Time Critical Process Mgmt	Mission Goal Driven Automated Response Strategy Nonlinear Adaptive Control
Level I-IV: Support Layers: (for each)				
	- Architecture - Utilities/GUI/API - DBMS/Link Mgr - Operating System	Open Architectures COTS SW Dev. Tools Object Oriented DBM's Symmetric Multi-Processors	Portability via Standards Interoperability Diverse Data DBM's Multi-level Security	Standard Architecture Common Core Modules Natural Language Interface Reuseability/Affordability

Figure 2-3. The State-of-the-Art And Key Challenges For Avionics Software Technology

representations (e.g. logic, frames, scripts, production rules, semantic nets) and procedures (e.g. searches) are preferable. The NN high pay-off "double win" applications are those with speed and adaptivity requirements such as described above. These are grouped into image processing and adaptive control categories for further discussion in the next section.

3. Neural Network Approaches for Avionics Roles

3.1 What are Neural Nets?

NN architectures are biologically motivated. A NN is an adaptive data and goal driven processing system that recognizes patterns in response to inputs based upon built-in learning mechanisms. NN's contain an extensively parallelinter connected system of simple processing elements (PB's) whose information is communicated via fan-out connections to other PE's. The processing in each node is local (i.e. based only upon its recent inputs and minimal local memory). As a result NN chips provide over a thousand-fold speed-up per watt and NN's learn to recognize patterns thus also reducing SW costs. NN's can learn on-line (i.e. in real time) and can be trained off-line by the user to discover patterns via solution exemplars or objective functions. Thus the user does not need to design, code, and test solutions to new problems. NN's can also be "hard-wired" with biologically motivated architectures for up-front sensor data processing (e.g. on-focal plane adaptive gain control, on-center off-surround convolution, and temporal lowpass filter modeling photo receptors, horizontal cells, and bipolar cells of the eye, respectively). A trained recurrent NN (i.e. with feedback) settles fast into a stable local equilibrium solution. The actual NN speed depends upon the implementation HW.

Biological neurons operate at -80 to 60 millivolts with millisecond widths. The action potentials (AP) provide rapid pulses for communication with the more analog postsynaptic potentials (PSP) providing the broader signals for local computation. The traditional inn-product NN's use sigmoids of weighted sums for PSP's and activation levels for streams of AP's. Pulse-stream NN's provide spatio-temporal integration of AP pulsed signals using hybrid encoding schemes (e.g. pulse width, threshold, frequency, phase) and PSP waveforms modulated via amplitude, time delay, and duration.

The Hybrid Temporal Processing Element (HTPE)[3] analog pulse-stream NN architecture patented by IRSI is tailored for applications (vs Mead's which supports neural research). The HTPE generates 2-10 nanosec AP pulses (vs milli-seconds: in brain) at 1-5 Volts (vs millivolts). The 100's nanosec PSP analog signals modulated in the neurode are sensitive to AP temporal pulse encodings. The HTPE provides the highest electronic computational thruput per watt ($>10^{12}$ Connections Per Sec (TCPS)/watt of all NN chips architectures. A comparison of NN powers and performances is given in Figure 3-1. The biological NN motivations incorporated or planned for pulse-stream NN chips include the following:

- Center-surround and shunting to accommodate large dynamic range
- Replicated receptive fields to detect oriented spatial contrast
- Diffusion (i.e. filling in) for spatial grouping
- Temporal pulse-stream dynamics to react to changing information
- Competitive learning and Hebbian synapse dynamics for pattern recognition
- Inhibitory/excitatory local processes to filter missing contours

3.2 The Role for NN's in Image Processing

The avionics image processing requirements are summarized as follows:

- Invariant: scale, shift, rotation
- Robust: background, occlusion, lighting, noise
- Adaptive: quick learning of new object/textures
- Real-time: compression of image information in user form
- Efficient: low size, weight, and power; high performance

The traditional approaches apply low (2-D), intermediate (2 1/2-D), and high (3-D) level processing[5]. The high level automatic target recognition (ATR) deficiencies include the following:

- model-based requires too many models for invariance and robustness

Company	Joule/Synapse*	Transistors/Synapse	Type	VLSI (CMOS) Tech	Chip Speed
Adaptive Solutions (C'NAPS)	4n Joule/syn	160 K trans/syn	Digital (SIMD)	0.8 μ	1 GCPS
Mitsubishi	15 p Joule/syn	420 trans/syn	Analog (Inner product)	1.0 μ	100 GCPS
AT&T Net 32 (1 bit) ANNA (6/3 bit)	 0.5 p Joule/syn 10 p Joule/syn	 20 trans/syn 45 trans/syn	 Analog (Inner product) Analog (Inner product)	 0.9 μ 0.9μ	 300 GCPS 10 GCPS (0.5 GCPS I/O limited sustained)
KaKadu	8 p Joule/syn	70 trans/syn	Analog (Inner product)	1.2 μ	2 MCPS
Lawrence Livermore	8 μ Joule/syn	—	Digital (Systolic Array)	—	12 MCPS
IRSI	25 f Joule/syn	16 trans/syn	Analog (Pulse stream)	2.0 μ	2 GCPS (1 K GCPS w/0.8 μ)

*W/chip = col 1 x col 5. Brain is .1 fJ/syn and E15 CPS. Optical Processing is ≅ 5 fJ/operation.

Figure 3-1. NN Power and Performance Capabilities per HW Technology Highlight The Advantage of Analog Pulse-Stream NN Chips

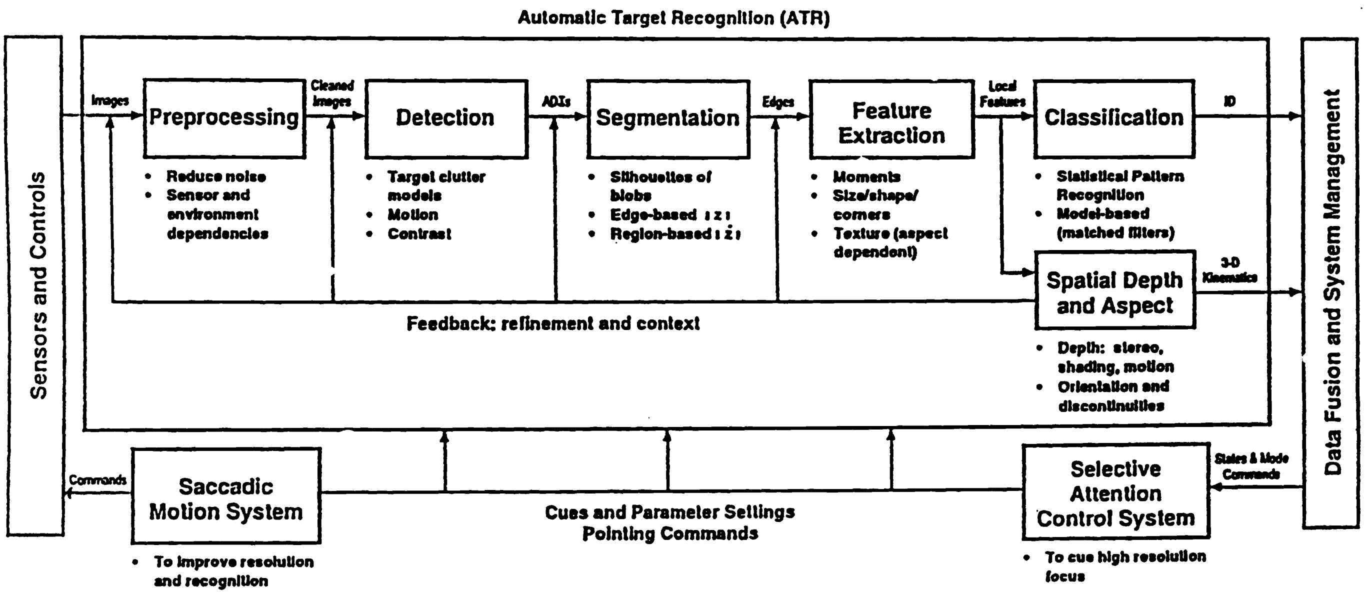

Fig. 3-2. The "Double-Win" Role For NN's Is Up-Front (near sensors and controls)

- invariance transforms (FT, Gabor, Log/Polar) are sensitive to noise and occlusion
- artificial intelligence methods rely upon good edges and are slow in other than limited domain
- statistical pattern recognition inhibits use of a prior context knowledge and degrades with number of classes

NN's are a key enabling technology which provides orders of magnitude increase in speed per size, weight, and power, as well as reduced solution development cost. This is accomplished by processing data nearer the focal plane using analog massively parallel processing which is trained on the data instead of a "programmed solution". As such, the NN automatically tailors its architecture parameters to fit each problem.

Numerous NN image processing paradigms have been developed including the neocognitron, feature contour system/boundary contour system (FCS/BCS), Le Cun constrained BPN, Seibert-Waxman, pulse-stream NN's SAHTIRN, hidden markov models with learning via Baum-Welch, higher order NN's, and replicated receptive field on log/polar transformed image.[6] Tests indicate that NN's achieve recognition comparable with traditional techniques with 300 connectings/pixed vs 300 ops/pixel. Thus an analog NN chip can achieve orders of magnitude more ATR performance per watt than a DSP chip. Such NN's are best applied where the key features need to be discovered.

The highest potential for NN's image processing is up-front. However NN adaptivity is applicable for the later functions also, see figure 3-2. The benefits for up-front on-focal plane analog signal processing include:

- Local contrast enhancement using full dynamic range ($\geq$18 bits) of detector
- User controlled adaptive spatial and temporal filtering
- Real-time pixel non-uniformity correction in log-space for gain (since offset=0)
- Pixel-level sensor fusion at the pixel rate (e.g. 60 frames/sec)
- Noise voltage is independent of detector resistance and improves with capacitance

In summary, the long-term NN image processing performance objectives include:

- Thousand-fold reduction inpower per operation
- Sensors and processing circuitry integrated together
- Extensively parallel analog computation (for speed with low power and area)
- Computations that map naturally to physical processes in silicon
- Computational methods that do <u>not</u> require high precision
- No "up-front" long-term storage (via fast processing)

3.3 The Role for NN's in Adaptive Control

The two traditional approaches to adaptive control are 1) direct control (such as performed in direct model reference adaptive controllers) and 2) indirect control (such as performed by explicit self-running regulators) Direct control techniques provide good stability however are susceptible to noise. whereas indirect control have low noise susceptibility and good convergence rate. However they require more control effort and have worse stability. Also, they are less robust to mismodeling. NN's synergistically augment traditional adaptive control techniques by providing improved mismodeling robustness both adaptively on-line for time-varying dynamics as well as in a learned control mode at a slower rate.

The NN control approaches which correspond to direct and indirect adaptive control are commonly known as inverse and forward modeling, respectively. More specifically, a NN which maps the plant state and its desired performance to the control command is called an inverse model, a NN mapping both the current plant state and control to the next state and its performance is called the forward model. When given a desired performance and the current state, the inverse model generates the control. The actual performance is observed and is used to train/update the inverse model. A significant problem occurs when the desired and achieved performance differ greatly since the model near the desired state is not changed. This condition is corrected by adding random noise to the control outputs so as to extend the state space being explored. However, this correction has the effect of slowing the learning and reducing the broadband stability.

For forward modeling the map from the current control and state to the resulting state and performance is learned and the plant sensitivity to control is passed to the inverse neural controller, see Figure 3-3. For cases where the performance is evaluated at a future time (i.e. distal in time), a predictive critic[7] NN model is learned. In both cases the Jacobian of this performance can be computed to iteratively generate the next control action.

However, this differentiating of the critic NN for back-propagation training of the controller network is very slow and in some cases steers the searching in the wrong direction due to initial erroneous forward model estimates. As the NN adapts itself the performance flattens which results in the slow halting of learning at an unacceptable solution. Adding noise to the controller's output[8] breaks the redundancy but forces the critic to predict the effects of future noise.

This problem has been solved by using a separately trained intermediate plant model to predict the next state from the prior state and control while having an independent predictor model generate the performance evaluation from the plant model predicted state[9,10]. The result is a 50-100 fold learning speed improvement over reinforcement training of the forward model controller NN. However, this method still relies on a "good" forward model to incrementally train the inverse model. These incremental changes can still lead to undesirable solutions. For control systems which follow the stage 1,2, or 3 models[11] the control can be analytically computed from a forward-only model. For the most general, non-linear (stage 4)[11] systems, an alternative is the memory-based forward model.[12] Judicial random control actions are applied to improve behavior only where the forward model error is predicted to be large (e.g. via cross-validation). Also using robust regression, experiences can be deweighted according to their quality and age. The high computational burden of these cross-validation techniques can be reduced by parallel on-line processing providing the "policy" parameters for fast on-line NN control.

For control problems which are distal in time and space, a hybrid of these two forward-modeling approaches can be used. Namely, a NN plant model is added which is trained off-line in real-time and updated as necessary at a slower rate than the on-line forward model which predicts performance based upon the current plant model. This slower rate trained forward-model NN supports learned control (e.g. via numerical inversion) whereas the on-line forward model provides the faster response adaptive control. Other NN control techniques such as using a Hopfield net to solve the optimal-control quadratic-programming problem or the supervised training of ART II off-line with adaptive vigilance for on-line pole placement have been proposed. However, their on-line robustness appears limited due to their sensitivity to a prior parameter assumptions.

A recommended hybrid approach is to use forward model recurrent NN to augments a reduced order model traditional controller for unmodeled modes and unforeseen situations. This approach has been successful in on-line learning for smart structure antenna control[13] and is extendable to other adaptive control problems such as flight control.

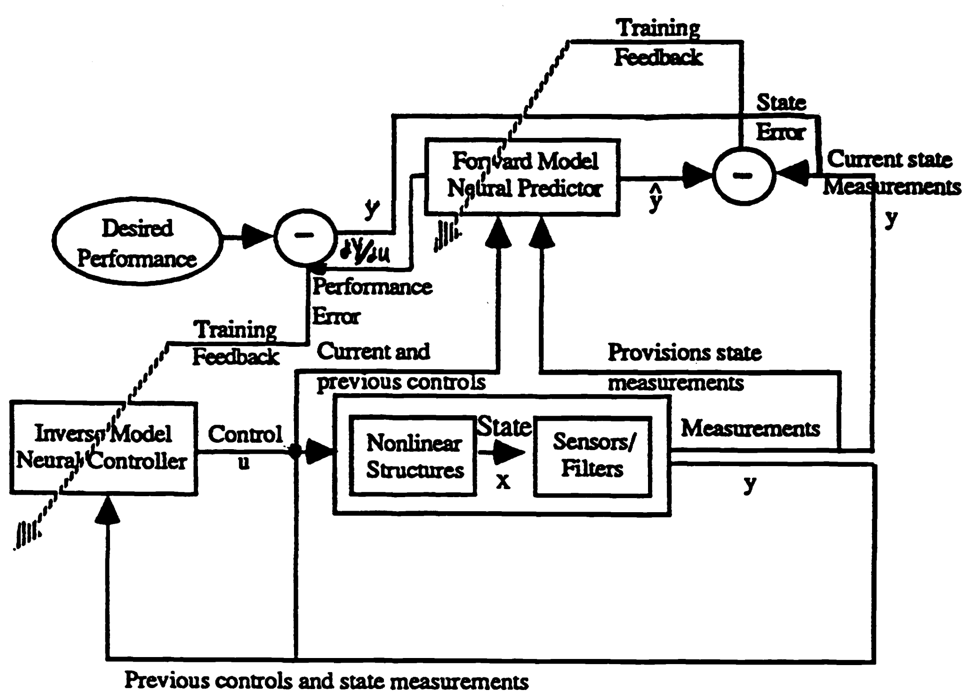

Fig. 3-3. Dual (indirect and direct) Adaptive Control Using Forward Model Plant Sensitivities To Aid Inverse Model Convergence

4. Summary

NN's provide the highest speed per watt and data/goal driven pattern recognition to augment traditional digital programmed computers. Consequently NN's have a role in satisfying avionics upgrade needs for affordable high speed multisource information fusion and adaptive response control especially near or on sensor and countermeasures hardware backplanes. Research and development (6.2) is needed especially in pulse stream NN's for on focal plane image processing and to enable NN's to program themselves to compensate for faults in its own or supporting HW.

5. References

1. C.L. Bowman "The Data Fusion Tree Paradigm and It's Dual" Proceedings of 7th National Fusion Symposium, invited paper, Sandia Labs, NM, March '94.
2. C.L. Bowman, "Data Integration (Fusion) Tree Paradigm", SPIE, Orlando FL, Proceedings, Signal and Data Processing of Small Targets, (Session 1698), April 1992.
3. C.L. Bowman and M. DeYong "Multispectral Neural Network Camouflaged Vehicle Detection Using Flight Test Imagery" World Congress on Neural Networks Proceedings, San Diego, CA, June 1994.
4. Mead, Carver, "Silicon Models of Neural Computation", Proc. ICNN '87, Caudill and Butler, eds., vol. 1, pp. 91-106, June 1987.
5. Marr, D.C.; and Poggio, T.; "A theory of human stereo vision," Proc. Roy. Soc. Longon, vol B. 204, pp. 301-328, 1979.
6. C.L. Bowman "Neural Network Image Processing" Very High Performance Processing SER 93-02-01-005 Ball Aerospace, Feb. 1993.
7. Barto, A.G., Sutton, R.S., and Watkins, C.J.C.H., *Learning and Sequential Decision Making. Univ. of Mass.* at Amherst COINS Technical Report 89-95, September 1989.
8. Jorden, M.I., and Jacobs R.A., *Learning to Control by Learning Only Forward Models*. NIPS 4, 1992.
9. Werbos, P, *Architectures For Reinforcement Learning*, in Miller, Sutton and Werbos, ed., Neural Networks for Control, MIT Press 1990.
10. Brody, Carlos, Fast Learning With Predictive Forward Models, Neural Information Processing Systems 4 (NIPS4), 1992.
11. K.S. Narendra, Adaptive control of dynamical systems using neural networks. *In Handbook of Intelligent Control-Neural, Fuzzy, and Adaptive Approaches*, Van Nostrand Reinhold, 1992.
12. Moore, A.W., *Fast, Robust Adaptive Control by Learning Only Forward Models*. NIPS 4, 1992.
13. C.L. Bowman "Neural Network On-Line Learning Control of Spacecraft Smart Structures", Advances in Neural Information Processing Systems (NIP-5), Nov. 1992, Morgan Kaufmann Publisher, San Mateo CA, 1993.