Vision in an Agent based on Fatiguing Leaky Integrate and Fire Neurons

Christian Huyck,
Dan Diaper,
Roman Belavkin, and
Ian Kenny

School of Computing Science
Middlesex University
London, UK
c.huyck@mdx.ac.uk

Abstract – *A long-term research and simulation methodology based on simulated human neurons is presented. One medium-term goal of this methodology is described; a software games agent that integrates vision and language in a biologically plausible manner will be developed. An implementation of a prototype vision system and the proposed topology of this agent is described. Part of the methodology is to address difficult neuro-theoretical problems; this proposed system will address the symbol grounding problem. Symbols will be grounded by the agent's interactions with the environment.*

Keywords: fatiguing LIF neuron, Cell Assembly, Visual Processing, Neural Agent, Retina.

For the Special Session on Biologically Plausible Neural Systems and Virtual Robotics.

1 Introduction

One of the major approaches to Artificial Intelligence (AI) is to develop a system that closely approximates the human brain at the neural level. A successful AI is often thought of being a computer system capable of passing the Turing test [21]. Initially proposed in 1950, the test involves the notion of placing a human dialogist in one room, a computer one in a second room, and a human judge in a third room communicating with the other two. The judge must determine which dialogist is human and which is the computer; if the judge cannot do this, the computer has passed the test. Passing the Turing test is a long standing target for AI researchers, but despite advances in the field of AI, no computer system has yet come close to passing the test.

At least two reasons are to blame for the failure of AI systems to come close to passing the Turing test. Firstly, most systems are based around symbols and these symbols have no firm connection to some world. This is known as the symbol grounding problem [3]. This lack of grounding makes it difficult for the system to learn from its environment, and thus learn the range of things that people are able to learn. Secondly, few systems are domain general. AI has been successful on domain specific problems so, success following success, there has been a great deal of development work on domain specific systems. The Turing test, however, has always been intended as a domain general problem; the dialogists and judge can explore any topic or topics to any depth. It has proven very difficult to build domain general systems. For example, Cyc [15] attempted to build a domain general system which boot straps a symbolic system by inputting a large amount of information from a few key domains. It was believed that it would then be able to learn about new domains based on this key information, but the system was far from successful at easily learning new domains.

Neural systems, it is claimed, can resolve both the symbol grounding and domain generality problems. The Cell Assembly (CA) hypothesis [5] states that human concepts are stored in groups of neurons that have a high mutual connection strength. These neurons can be activated directly or indirectly from their external environment causing a cascade of neural activity in the CA that can persist after the external activation ceases. The neural connections can be learned in an unsupervised manner from the environment. So, the CAs are based on environmental input and thus are grounded.

Domain generality might be solved by developing neural systems that follow a biological human model. That is, systems are based on known human neural topology, which causes the systems to perform within known psychological parameters. This neural cognitive modelling approach provides an obvious set of target parameters. The aim is to develop systems that adhere to known neural structures and perform tasks that humans perform in a way that is consistent with what is known of biological and neuropsychological processing. The more psychological phenomena that can be accounted for, the better the system.

To build a system that can pass a Turing test, the system needs to be based in a reality and from it learn symbols, new ways of combining symbols, and new mechanisms of combining symbols. Some artificial neural systems can do these things.

The authors have been constructing increasingly complex neurally based systems. The neural model is based on a type of Leaky Integrate and Fire (LIF) neuron [20] that fatigues. This paper presents initial work on an AI system that acts as an agent in a virtual environment. The agent will combine a visual system with natural language input processing and output behaviour. CAs are composed of neurons, are the neural correlate of symbols, and will be activated via vision and language. Initial work, based on a neurally and topologically plausible simulation, on the agent's visual system is described. The proposed architecture for this integration, all based on fatiguing LIF neurons, is also discussed.

2 Methodology

The vision system work described in this paper is a portion of a larger method and research agenda. The ultimate goal of this agenda is to build a system capable of passing a Turing test. The current project involves building a virtual games agent that senses its environment, behaves independently in the environment, takes commands in natural language and intelligently replies to, or acts on, them.

The overall plan is to start with relatively simple systems, then to use these systems as the basis of more complex systems that do more complex tasks. While building these systems, three basic tenets are followed:

1. Resolve neuro-theoretical problems as they arise. Basic theoretical problems, such as variable binding and symbol grounding, have been identified. These problems need to be resolved. Solve these problems as soon as possible to avoid pursuing paths that will fail in the long run. Also, identify new theoretical problems as soon as possible.

2. Pay attention to neurobiology, neuro-psychology and psychology. Try to build systems that adhere to known neuro-biological, neuro-psychological and psychological theories. The systems need not adhere to all constraints, but should identify omissions and inconsistencies that future systems should attempt to resolve. Unlike traditional cognitive architectures, this method would lead to an implementation of a cognitive architecture in the relatively well understood neural band [17]. Higher level bands, e.g. the cognitive and rational band, would be implemented in this basic band. System performance at each band can be compared with the appropriate human correlate. This tenet leads to four particular sub-tenets.

   (a) Use a neural model that is a close approximation of human neurons. It does not need to be a perfect model, indeed it cannot be perfect as it needs to be efficiently simulated on a computer.

   (b) Implement systems that are grounded in a world. Humans exist in their world, which is part of a physics universe, for example, only some frequencies in the electromagnetic spectrum are visible. Apart from the physical, other properties can exist in a system's world, for example, the probability of an event occurring. Some properties, however, such as social or ethical ones, or Gibsonian affordances [4], which can apply to any thing in a world, can either be genuine properties of the world itself or they could be properties of the model of the world constructed by a system, human or AI. Either way, this allows an AI system's symbols to be grounded and allows interactions between the AI agent and its world. Moreover, it allows the AI system to solve problems in its world.

   (c) The system should learn in a biologically plausible manner. The Hebbian Learning rule [5] is supported by biological evidence. Learning is crucial to allow the system to develop.

   (d) Components should be integrated. Subsystems should be combined and connected in a manner similar to how areas of the brain are connected. This enables the communication and sharing of information. Note that this is all consistent with Hebb's CA hypothesis; groups of neurons with high mutual synaptic strengths are the basis of human concepts. There is evidence that a given CA can span several portions of the brain [18]. So, the concept of *dog* will have some neurons from the visual areas, some from the cognitive, language and motor areas, and some from elsewhere in the brain e.g. areas associated with general emotional responses to dogs, such as loving, or being frighten of, nearly all of them.

3. Implement systems. Systems are really a great testbed. Implement them to see where the problems are, and to have things that are useful to solve problems.

These tenets form the basis of a development methodology. Systems using this methodology can be validated by comparison with other artificial systems, and, crucially, by comparison with human performance. For example, two artificial systems can be compared across a range of tasks. While one might perform better at more of the tasks, the other may compare more closely to human performance. For example, it may produce errors or biases similar to those of people. In this case, the second system may be considered a better basis for further exploration.

In theory, another advantage of this methodology is that components can be replaced or combined in new ways. For example, if someone develops a better vision model, it should be easy to replace an earlier component, and so develop a better overall system. In practice, of course, there are often interface problems between independently developed components and the "better" concept is rarely a simple one, more often being a balance between different functionality and the biases of the evaluators as to functional importance.

3 Previous Work

In earlier work, the first author proposed that neural systems could be used as an intermediate level for modelling cognition [8]. The above methodology is a refinement and expansion of that earlier proposal. This section describes some related systems, the fatiguing LIF neural model used, and some prior work that has been done using this neural model.

3.1 Related Systems

The authors are not the only researchers to build complex neural systems. Systems have been built that are both cognitive models and functioning robotic systems.

A good example of a cognitive neural model is one based on continuously variable Hopfield neurons [13]. This combines ideas from cognitive psychology and neural computation to produce an integrated model for a human-like form of working memory, noting that psychologists are not in universal agreement about the detailed properties of the one or more putative, short duration, human memory systems. Hebbian learning rules are used to set connection weights. This system provides a neurally based cognitive model of serial memory that accounts for several traditional, cognitive phenomena associated with short term information storage, including primacy and recency effects.

As another example, a robotic system has been developed [14] that takes natural language commands. It is based on model neurons with a medium degree of biological faithfulness, implements a regular grammar, grounds its semantics in its world, and resolves ambiguity dynamically.

These systems show that neural simulations can be used to implement complex systems, both as robotic systems and as cognitive models. One of the key questions in developing such a system is the type of neural model that is used.

3.2 Neural Model

The systems described in this paper are based on fatiguing LIF neurons. Neurons collect activation from other neurons via synaptic connections. If the neuron does not fire some of that activation leaks away. Equation 1 describes the activation of a neuron i at time t if it does not fire at time $t-1$.

$$A_{i_t} = \frac{A_{i_{t-1}}}{d} + \sum_{j \in V_i} w_{ji}, 1 < d \quad (1)$$

The amount of leak is d. V_i is the set of all neurons that have connections to i and fire at time $t-1$. The weight, or synaptic strength, of the connection from neuron j to neuron i is w_{ji}.

The model is based on discrete time steps. This allows the whole system to be updated simultaneously. It can be argued that each time step is roughly equivalent to 10 ms. of simulated time. This enables the system to ignore refractory periods and synaptic delay as these are all within the 10 ms.

Neurons also fatigue so that the more closely adjacent steps at which they fire, the more difficult it becomes for them to fire. While a simplification of the biological system, where fatigue is associated with ion transport, it is modelled by increasing an activation threshold θ if a neuron fires as described by Equation 2.

$$\theta_t = \theta_{t-1} + F_c \quad (2)$$

In Equation 2 the threshold θ at time *t* is set to the threshold at time *t-1* + the fatigue constant F_c. If the neuron does not fire, the threshold is reduced toward the base resting level as in Equation 3.

$$\theta_t = \theta_{t-1} - F_r \quad (3)$$

The threshold is reduced by the fatigue recovery constant F_r though it never becomes less than θ. So a neuron fires if it has more activity than the threshold plus accumulated fatigue. If it fires, it loses all activity and is reset.

In this system, neurons may be inhibitory or excitatory, but they obey Dale's principle [2] that a neuron cannot have both inhibitory and excitatory synapses leading from it.

This fatiguing LIF neural model leaves out much of the detail of biological neurons, but is intended to capture what are believed to be the most crucial details for computational purposes. However, if future simulations show such details missing, the neural model will be revised to more accurately reflect a more realistic biology. One anticipated difficulty is that adding such detail does not guarantee improved AI systems performance, indeed, it could become worse, for example, where newly added detail can only improve performance in association with other necessary things that have not been added. The general approach when adding detail, therefore, is to go through the complete model of the system, revising where appropriate each component that can provide interaction within a stage, to its output stage(s), and to preceding stages, i.e. where there are top-down processing effects.

3.3 Prior Work with fatiguing LIF Neurons Model

Extensive exploration of CA models using fatiguing LIF neurons has led to a range of simulations. Some of the more advanced simulations have led to systems with, for example, Hebbian learning mechanisms, categorisation processes and the ability to apply rules.

The Hebbian learning rule states that two neurons that have a connection and tend to fire simultaneously will have that connection strengthened. This leaves a wide range of possible ways that this can be implemented in AI systems. A compensatory learning rule has been described [9] that limits the total synaptic strength leaving a neuron. This enables CAs to form and allows individual neurons to participate in multiple CAs.

Since CAs that share neurons overlap at the neural level, this provides a taxonomic mechanism which can form the basis for hierarchical categorisation. Artificial categories have been learned [10] in an unsupervised manner and create a categorisation hierarchy.

This CA model has also been used for useful human-world categorisation systems [12]. The model has performed well on a standard categorisation task, the Congressional Voting Task, and on an information retrieval task.

Rules have also been implemented in the model and the system used for a counting task [11]. This has used a novel variable binding mechanism based on changing synaptic

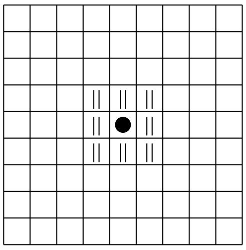

Figure 1: 3x3 On-Off Receptive Field in V1

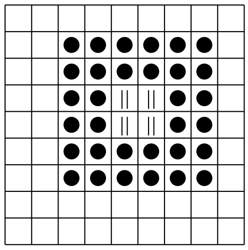

Figure 2: 6x6 Off-On Receptive Field in V1

strength. Spontaneous neural activation is used to erase bindings. The model has thus already provided a novel solution to one major problem of subsymbolic AI, the variable binding problem. The authors have investigated a system based on the fatiguing LIF model.

The latest system will be an agent in the virtual environment of a video game called Half-Life [19]. The agent is an assistant to a player in the game and will take natural language commands. The agent will sense the environment from the same data that generates the screen that the user sees. The agent will move about the environment and try to fulfil the user's commands. All processing that the agent does will be done by fatiguing LIF neurons.

4 Visual System

As a first step in developing the games agent, a prototype, proof of concept, visual system has been developed for it. The vision system is broken into three layers simulating the retina, the primary visual cortex (V1) and the secondary visual cortex (V2). While a simplification of the human visual system, the system still possesses functionality sufficient to recognise lines and simple shapes.

Visual input to the agent corresponds to a subset of pixels equivalent to those that are presented on a user's screen. The system simulates monocular vision with a stationary eye. The current simulation being used to test its general computational properties is based on a 30x30 input array of binary pixels.

Simulated retinal processing, via fatiguing LIF neurons, codes the input so that output from the retina is a set of spatial frequency tuned channels similar to those recorded in the human optic nerve. The actual simulation is based on three sets of overlapping, square receptive fields; the fields are 3x3, 6x6, and 9x9. Each field functions as having an on-centre/off-surround and vice versa. The 3x3 on-off receptive field responds optimally when the centre pixel is on, and the surrounding eight are off. Figure 1 represents this with the circle representing on, and the double bars representing off. Similarly, the 6x6 on-off field responds optimally when the centre four pixels are on, and the surrounding 32 are off, and the 9x9 on-off receptor responds optimally when the centre nine pixels are on and the remaining 72 are off. The off-on receptive fields are the reverse of their on-off mates; figure 2 represents a 6x6 off-on receptive field.

In the logical model of the retina that has been developed, which specifies the total number of possible outputs, there is one output for each receptive field, so with 900 input pixels, there are 5400 logical retinal neurons. The AI system, using the fatiguing LIF architecture, however, may require fewer neurons to achieve the number of logically possible retinal outputs.

These simulated neurons correspond to information that is passed from the retina to the optic nerve in the human visual system. It is not, however, an exact duplicate of the processing steps in the retina. For instance, this simulation has no fovea, supports only monochromatic vision, and, for example, has no horizontal cells. Future simulations may introduce some of these retinal features, but this simulation is designed to be, and does function as, a simple shape discriminator.

The output of the retina is the input to V1, i.e. processing in the Lateral Geniculate Nucleus has been ignored. The V1 consists of a series of different types of neurons.

Orientation and position specific line and arc detectors have been implemented. These are types of simple cells that are known to exist in V1 [1]. There are six orientations of line that are recognized, and lines roughly one pixel thick are recognized. This is done by the orientation and location specific V1 line neurons taking inputs from the retinal neurons that implement the logical 3x3 receptive field. A simple, repetitive expansion is being implemented so that more orientations and thicker lines are recognized.

To recognise a horizontal line, neurons in V1 have an activation threshold, θ, of four. The connections from the 3x3 on-off receptor in the retina to a particular horizontal line neuron have weights of 1.4 and each horizontal line neuron is connected to three horizontally adjacent retinal receptors. The 3x3 receptors respond suboptimally when a horizontal line passes through them, firing in the third and seventh cycles. As all three connections to the horizontal line neuron fire in a given cycle, it receives 4.2 units of activation. This surpasses θ so the neuron fires indicating a horizontal line is present.

There are four different orientations of angles that are recognized. These take inputs from the neurons that implement the 6x6 receptive fields. They only recognize angles when lines do actually meet. The angle neuron has connections from the 6x6 on-off neurons that have weights of 3. An angle corresponding to the less-than sign, for example, has connections from neurons that implement the receptive field centred on the angle, the one above and to the left, and the one below and to the right.

The output of V1 is the input to V2. V2 is set to recognize two orientations of hollow triangles. It does this both position dependently and independently. Output from the line and arc neurons in V1 is sent to neurons in the V2. Additionally, input from the retinal neurons with 9x9 off-on receptive fields is also sent to V2. Together, these neurons activate position and orientation specific triangle detectors. These specific triangle detectors also activate location independent neurons in a fashion similar to earlier hierarchical categorisation work [10].

The separation of V1 and V2 is clear in the AI system, but it probably corresponds quite poorly to human neural architectures. For example, in bigger systems, the larger receptive fields will project to V1 as well as, or instead of, to V2.

These initial implementations are just the beginning phase of a visual processing system, but we believe that they provide a sound basis for further work on the other major components. Future work on visual processing will include motion detection, simultaneous multiple object recognition, and the learning of new objects, but not, of course, in this project, stereopsis.

In the current system there is no learning as the connection weights and topology were programmed. Future work, based on previous implementations (e.g. [10]), will implement the fatiguing LIF mechanisms so that weights can be learned, changed and maintained.

5 Outline of the Proposed Agent's Architecture

The vision system is just one component of the proposed agent. The role of the agent is to support a user playing the Half-Life game and to do this it will integrate vision, language and action. The agent has to see what is in the environment and be able to understand the user's language input, be able to reply to the user, and convert user commands into actions that are performed in the game. The visual processing of the agent will be modelled as described in the prior section and both language and action will be modelled using the same fatiguing LIF architecture.

Words will be represented by CAs so they can remain active after external stimulation ceases. Language processing capabilities will have two sources. First, nouns and concrete adjectives will be presented along with visual representations of each word, so that these will become associated. Thus, the symbolic words will be grounded with their visual representations, yielding grounded symbols.

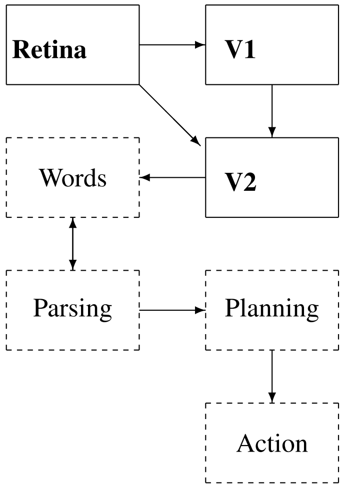

Figure 3: Simple Rule Topology

Second, while it is difficult to stimulate neurons directly in the brain, it is simple to do so in the simulation. A word can be presented directly to the model by stimulating some of the neurons associated with the word. Language processing will be a neural implementation of earlier symbolic work [6]. A mechanism for neural language processing has already been proposed [7]. Parsing rules will be derived from linguistic data and an existing rule system [11]. The application of the rules and conflict resolution will be done by a parsing system using a pseudo-stack.

In later implementations, a spreading activation net [16] will be used to give the agent goal-directed behaviour based on the user's natural language inputs, while remaining responsive to the immediate environmental context that both user and agent can see and act upon. The extent to which longer term, higher level goals can be supported by the agent, particularly when the route to such goals is indirect, i.e. as typically found in 'Tower of Hanoi' type situations, remains an empirical issue to be investigated.

The spreading activation net will be linked to a network representing the motor area in the brain. As the Half-Life game requires symbolic inputs for actions, these will be triggered by CAs in the neural agent. So, if a CA in the motor region is active, the agent will emit a symbolic command to the Half-Life game to execute the appropriate action.

Figure 3 is a diagram of the entire proposed system. Solid boxes have an initial implementation, and dashed boxes are proposed modules.

The new system will solve another major problem associated with subsymbolic AI, the symbol grounding problem [3]. Moreover, these problems will be solved with a biologically plausible neural model, fatiguing LIF neurons, and biologically plausible neural topologies. While this model and the topologies are simplifications of the brain, they should be sufficient to implement the proposed AI game agent.

6 Conclusions

The proposed agent will address the fundamental symbol grounding problem of AI. By solving this problem, the system will be able to learn natural semantics. This will open a new class of systems that can learn from the environment by interacting directly with it. This system will take advantage of all three tenets of our methodology: a system will be implemented; it will adhere reasonably closely to known neurobiology; and the system, for example, will address the long standing neuro-theoretical problem of symbol grounding.

A simple vision system has been described that adheres closely to some of the known properties of the human visual system. Based solely on fatiguing LIF neurons, a biologically plausible model, it processes and stores information.

A proposal has been made for the integration of this module with proposed language and action modules. These new modules and the integration process have been sketched.

By processing with biologically plausible neurons and using biologically plausible topologies, the system can use known neuro-biological properties. This can test neurobiological theories, but it can also draw inspiration from these theories to develop more sophisticated AI systems.

Acknowledgements: This work is supported by EPSRC grant EP/D059720 - 'Natural Language Parsing using Cell Assemblies: Computational Linguistics with Attractor Nets'.

References

[1] P. Churchland and T. Sejnowski *The Computational Brain*, MIT Press 1992.

[2] J. Eccles. "Chemical Transmission and Dale's Principle", *Prog. Brain Research*, Vol 68, pp. 3–13, 1986.

[3] J. Fodor *The Mind Doesn't Work That Way: the Scope and Limits of Computational Psychology*, MIT Press. 0-262-06212-7 2000.

[4] J. J. Gibson, *The Ecological Approach to Visual Perception*, Lawrence Erlbaum Associates, 1986.

[5] D. Hebb, *The Organization of Behavior*, John Wiley and Sons, New York, 1949.

[6] C. Huyck. "A Practical System for Human-Like Parsing", Proc. of the 14th European Conference on Artificial Intelligence, Berlin, pp. 436–440 2000.

[7] C. Huyck, "Natural Language Parsing with Cell Assemblies: A Model of Nonconscious Human Language Processing", Proc. of the AISB'01 Symposium on Nonconscious Intelligence: From Natural to Artificial, York, UK pp. 31-42 2001.

[8] C. Huyck, "Cell Assemblies as an Intermediate Level Model of Cognition", in *Emerging Neural Architectures based on Neuroscience* pp. 383–397, 2001.

[9] C. Huyck, 'Overlapping Cell Assemblies from Correlators", in *Neurocomputing* 56 pp. 435–439, 2004.

[10] C. Huyck, 'Creating Hierarchical Categories Using Cell Assemblies", in *Connection Science* in print.

[11] C. Huyck and R. Belavkin, "Counting with Neurons: Rule Application with Nets of Fatiguing Leaky Integrate and Fire Neurons", Proc. of the Seventh International Conference on Cognitive Modelling, Trieste, Italy pp. 142–147 2006.

[12] C. Huyck and V. Orengo, "Information Retrieval and Categorisation using a Cell Assembly Network", *Neural Computing and Applications*, Vol 14, pp. 282–289, 2005.

[13] M. Jones and T. Polk, "An Attractor Network Model of Serial Recall", Proc. of the Fourth International Conference on Cognitive Modelling, Fairfax, VA. pp. 121–126 2001.

[14] A. Knoblauch and H. Markert and G. Palm, "Associative Model of Cortical Language and Action Processing", Proc. of the Ninth Neural Computation and Psychology Workshop, Plymouth, UK pp. 79–83 2004.

[15] D. Lenat, "Cyc: A Large-Scale Investment in Knowledge Infrastructure", *Communications of the ACM*, 38:11, pp. 33–38, 1995.

[16] P. Maes, "How To Do the Right Thing" *Connection Science*, 1989.

[17] A. Newell, *Unified Theories of Cognition*, Cambridge, MA: Harvard University Press, 1990.

[18] F. Pulvermuller " Words in the brain's language", *Behavioral and Brain Sciences* 22 pp. 253–336 1999.

[19] Sierra, Half Life, from http://games.sierra.com/games/half-life/ 1998.

[20] D. Tal and E. Schwartz "Computing with the Leaky Integrate-and-Fire Neuron: Logarithmic Computation and Multiplication", *Neural Computation* 9:2 pp. 305–318 1997.

[21] Turing, A. (1950) Computing Machinery & Intelligence. In *Mind* 59:236 pp. 433-60