SIMULACRUM OF THE MIND'S EYE: MODIFYING THE NEOCOGNITRON FOR CONSTRUCTING OBJECT REPRESENTATIONS

by

Joseph L. Chu
Student ID #5133663

COGS499 – Research Report Option
Research Proposal
Department of Computing
Queen's University

2009

Contents

Abstract

There have been countless theories throughout history regarding the human mind from a wide range of fields of inquiry. Developments in psychology, neuroscience, and computer science led to the advent of neural networks that were used to model such things as visual perception and memory. At the forefront of this trend, the Neocognitron proved that a biologically inspired architecture could be utilized to recognize hand-written characters. Further advances have led to network models that show promise in the field of object recognition. The question that is proposed to be answered is whether such networks could also categorize and classify objects recognized. If so, the potentiality exists to combine such a network with a semantic memory network to produce a visual semantic memory model capable of object representation. This model is divided into two semi-independent stages, the first of which is object recognition, and the second is semantic representation. Following a biological development chronology, focus is placed on the first stage for further study. A prototype network is proposed and three immediate problems with implementation considered: grayscale conversion, network dimensions, and learning algorithm. After discussing the tentative results of the prototype, further research possibilities are looked at, including possible incorporation of the second stage of object representation.

Introduction

It should be possible to say with confidence that readers of this work, if they are able to understand what is written here, must obviously possess a mind to understand it with. Though not necessarily the case, it should be reasonable given the current state of the known world to further assume that such a mind would be human in nature. The

search for understanding the human mind by human beings then can be considered but an intriguing manner of self-reflection. More than a mere question of the science of matter or life, it is a question of who and what we are, to study this symbolic thought machine that philosophy calls the mind, and biology calls the brain.

Historically, the mind has been given a mystique that made it all but untouchable. Famed philosophers such as Aristotle and Descartes considered it to exist in a separate plane of existence, a world of ideas and souls. More recently with the development of the discipline of psychology, the scientific method has been applied to try to unravel the mysteries of the mind. After many false starts, psychological behaviorists such as Skinner and Watson considered the conscious mind to be too unobservable to study scientifically, but cognitive psychology has resuscitated the notion (Pylyshyn, 1998).

The functionalist perspective currently dominates Cognitive Science along with the Computational Theory of the Mind and the Tri-Level Hypothesis. Functionalism regards the difference between natural and artificial intelligence as being a superficial one, that if it functions like it can think, then for all intents and purposes it can. In effect, it demystifies the mind from Cartesian dualism's separation of soul and body. The Computational Theory of the Mind builds on this by providing the analogy of the computing machine as a way of understanding cognitive processes as a form of information processing. The Tri-Level Hypothesis expands on this by dividing the study of intelligent systems into a three level hierarchy (Pylyshyn, 1998). The first, most fundamental level is the biological or physical level, the second, intermediate level is the symbolic or syntactic level, and the third, most abstract level is the knowledge or semantic level.

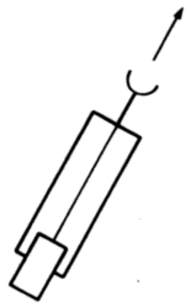
The mystical and philosophical concept of the mind's eye is something of a metaphor for the manner in which objects can be internally represented and recalled, a kind of mental imagery (Anderson, 2000, p. 111). Much debate has occurred over whether and to what extent such images actually exist in the mind, and if so, in what format. These debates call to attention the greater question of the format of mental representations in general. While the idea of mental pictures, the notion of there being a capacity in the brain to exactly reproduce a visual perception in all its original detail is regarded as an inaccurate understanding of perception, it is difficult to argue against the brain being able to recollect constructed representations of objects previously perceived (Pylyshyn, 2003). Such representations do not contain the exact pixel by pixel accuracy of the actual object, but then it is unlikely that the images that we perceive possess such accuracy either. Indeed, the common phenomenon of visual illusions is only possible because perception itself involves a degree of cognitive processing. Thus what we see in our minds is not so much a reflection of the real world as a combination of real world information with existing knowledge of a given object or objects in general. The properties of objects are thus partially projections of our memory, filling in the blanks and enabling us to identify objects without having to thoroughly investigate every angle.

The Evolution of the Anatomy of the Mind

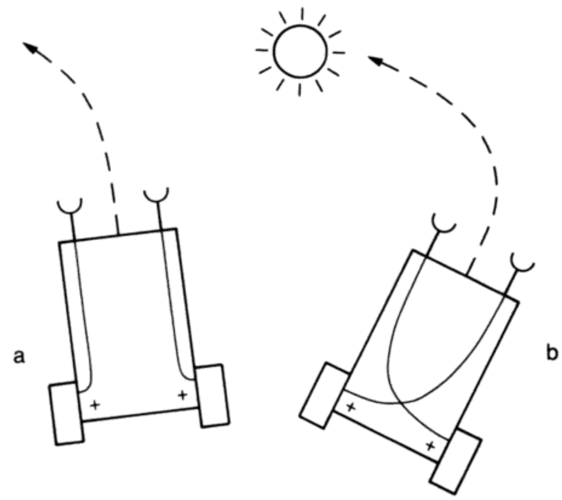
To understand this unique behaviour requires an understanding of the manner in which the brain is wired. There is a tendency in research to focus on specific angles such as perception or memory as exclusive fields of study. While this is useful as a means of narrowing down such complexity as the brain into compartments that can be studied independently and thus with greater focus and ease, it is naïve to become so narrow-

minded as to ignore anything outside of this limited scope. Any truly meaningful model of the mind should incorporate elements from the multitude of streams of psychology and neuroscience.

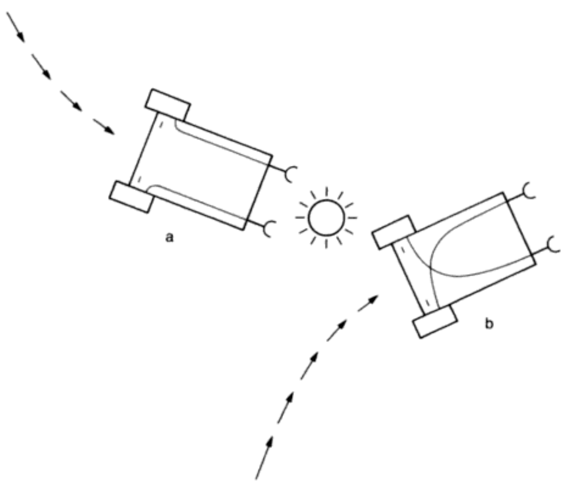
In the context of the Computational Theory of the Mind, any research into the human mind should be able to fit the hardware aspect of the theory to the biology of the human brain. The reasoning for this was possibly best expounded by Italian-Austrian neuro-scientist and cyberneticist Valentino Braitenburg in this seminal work, Vehicles – Experiments in Synthetic Psychology (Braitenburg, 1984). A former director at the Max Planck Institute for Biological Cybernetics in Tübingen, Germany, Braitenburg provides a plausible explanation for the long running mystery of why the human brain is left/right cross-wired in a simple analogy. Imagine a simple robotic agent with two sensors in the front and two motors in the back. Assuming that each sensor is connected to one motor, there are two possible configurations. Either the left sensor is connected to the left motor and the right sensor to the right motor, or the left sensor is connected to the right motor and the right sensor to the left motor. In the first case, the effect is that the agent will turn away from objects that stimulate its sensors, while in the second case the agent will turn towards objects that stimulate its sensors. While both avoidance and attraction are potentially useful actions for the agent to be capable of, it would seem that for predators and animals, not to mention light seeking algae, that the attractive layout is the one that attracts the agent towards sources of food, mates, etc. Thus, the cross-wired configuration evolves, as seen in Figure 1. This is of course a simplification, but it shows how a biological adaptive explanation allows us to understand better the architecture of the mind.



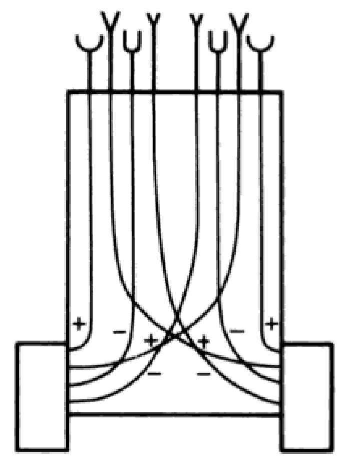
Vehicle 1, the simplest vehicle. The speed of the motor (rectangular box at the tail end) is controlled by a sensor (half circle on a stalk, at the front end). Motion is always forward, in the direction of the arrow, except for perturbations.



Vehicles 2a and 2b in the vicinity of a source (circle with rays emanating from it). Vehicle 2b orients toward the source, 2a away from it.



Vehicle 3, with inhibitory influence of the sensors on the motors.



A multisensorial vehicle of brand 3c.

Figure 1. Four diagrams showing the evolution of the sensor-motor wiring of the hypothetical vehicle from Braitenburg (1984).

Interestingly, this also suggests that the modern human fight-or-flight response is at least in part founded on the wiring of our primitive single celled ancestors. Indeed the whole tendency seen in psychology of dualistic, black and white binary thinking seen in depressed patients, the “Us vs. Them” mentality, and many other cognitive phenomena

could be sourced in this deceptively simple explanation. It suggests at the very least that to understand the mind, tracing its potential root causes is a plausible route of inquiry.

Models of the Mind

Much has been made of efforts to create comprehensive models of the human mind in cognitive psychology. Anderson's ACT-R 5.0 is one such effort (Anderson et al., 2004). While there are many others, the majority seem to share the general concept of some kind of Sensory Input module leading to a Perception module which in turn leads to a Memory Trace and eventually to Recognition Output. Memory and Perception in such frameworks are often characterised as a feedback loop that culminates in recognition.

Thus, the first stage of Object Representation is necessarily recognition. This recognition process involves the aforementioned Sensory Trace and Perceptual Feedback elements. Efforts to model and simulate this stage specifically would include Fukushima's Neocognitron neural network as well as other perceptual models (Fukushima & Miyake, 1982).

The second stage of Object Representation involves Semantic Association, which is to say, the memory system that carries meaning and non-perceptual abstract information. This stage allows the initial recognition of objects to then activate the full Semantic Representation, including associated concepts. Examples of research that could be categorized as modeling this segment would include semantic networks such as LSA, BEAGLE, and other semantic memory models (Jones & Mewhort, 2007).

While both stages are essential for a full understanding of the process of object representation in the human mind, it makes more sense to first study the first stage, as it is more inclined to have developed earlier in the evolution of the brain and would also be

more likely to have formed the basis for the architecture of the brain. Before one can attempt to seriously deconstruct the nature of the human memory storage system, one should first decipher the format of the objects and concepts that are being stored.

Connectionist models of the brain represent a unique paradigm. While deceptively simple in their basic construction, connectionist neural networks are able to perform highly complex computations, including planning, decision making, and learning, while at the same time possessing both psychological and neurological plausibility (Thagard, 1996). In keeping with the concern for maintaining grounding in the biological basis of the brain, the most obvious starting point in any effort to reconstruct the human perceptual memory system would be a connectionist model built around a specific perceptual system, such as the much studied visual system.

Modeling the First Stage

Image recognition is one of those fields that has received considerable attention for decades, but which has proven to be a deceptively complex problem. What seemed at first to be a simple mechanical perception phenomenon has turned out to require considerable cognitive processing. While efforts have been made to solve the problem mechanically as a purely algorithmic problem, a more innovative approach has been to analyze the biological basis of human and animal vision as a model for artificial vision systems.

The Neocognitron is a multilayer feedforward neural network devised for visual pattern recognition, and trained via competitive learning (Fukushima & Miyake, 1982). Its design is modeled on the way the human eye and occipital lobe process visual information, but is structured specifically for numerical and alphabetical characters. The

Neocognitron is based on the earlier Cognitron, which was an alternative neural network to the better known Perceptron. Since Fukushima's original work on the Neocognitron, other attempts have been made to extend the functionality of the Neocognitron to incorporate more complex elements of visual processing, and move beyond simple characters to actual real world objects.

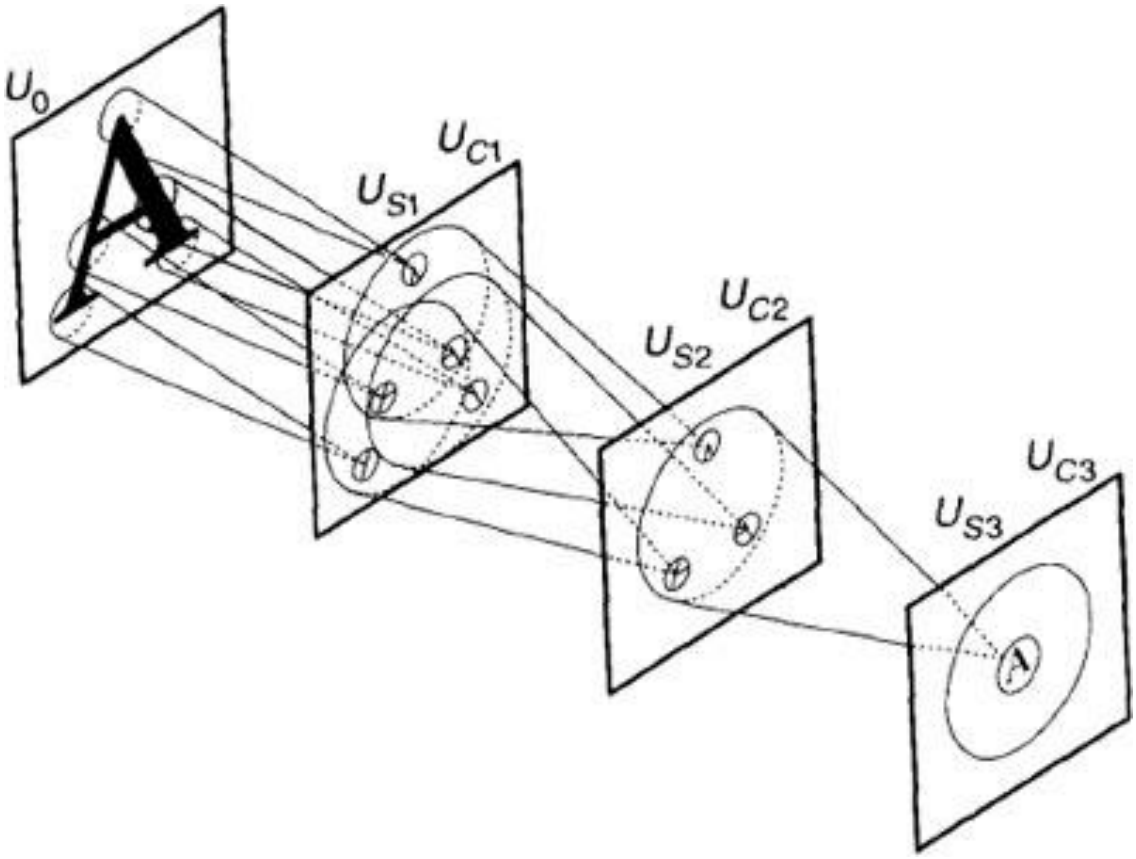


Figure 2. A commonly used depiction of the action of the Neocognitron, as it delocalizes features to identify the character 'A'.

The Biology and Evolution of the Neocognitron

As a hierarchical competitive-learning based neural network, the notion of biological basis was fundamental to the conception of the design of the Neocognitron. The architecture of the network is comprised of S-Cell Layers and C-Cell Layers based

on the Simple Cell and Complex Cell hierarchy hypothesis proposed by Hubel and Wiesel to explain the function of Ganglion cells in the Basal Ganglia (Fukushima, 2003). What this in effect does is perform pattern recognition through the delocalization of features in the visual receptive field, as seen in Figure 2.

Since its inception in the 1980s the model of the Neocognitron has been repeatedly updated and refined over the past twenty or so years. In that time however, advances in neuroscience have called into question the original hypothesis of Hubel and Wiesel. Recent evidence suggests that Complex Cells are actually a separate parallel pathway working alongside Simple Cells (Wolfe et al., 2006, p. 61). This would seem to question the validity of the whole concept except that the hierarchical model can actually still be justified by extending the analogy further along the visual information processing pathway.

Thus the hierarchy now begins with the retinal ganglion cells as they lead to the lateral geniculate nucleus, which in turn sends more delocalized information to the striate cortex in which both simple and complex cells send more condensed signals to the extrastriate cortex, which then sends signals into the inferotemporal (IT) cortex. Damage to the inferotemporal (IT) cortex in the temporal lobe is known to cause agnosia reliably in patients, suggesting the IT cortex is where object recognition is finalized (Wolfe et al., 2006, p. 94). The inferotemporal (IT) cortex is also connected to the hippocampus and is thus associated with memory formation. This appears then to be a clear neurological indicator of an interaction between perception and semantic memory, as seem in many models. Though contrary to ACT-R 5.0, there appears to be a direct connection between the occipital and temporal lobes as shown in Figure 3. That the ACT-R 5.0 model

inaccurately separates these modules is likely a result of its basis as a top-down approach in which it tries to hypothesize an appropriate arrangement for the brain. Thus it attempts to match brain regions to this hypothetical model, rather than first recognizing the existing structures in the brain and constructing the model from the ground up. If anything this shows the limitations of a top-down approach to modelling the mind and rules in favour of the bottom-up approach used by more connectionist models, which can claim to be a bit closer to reality.

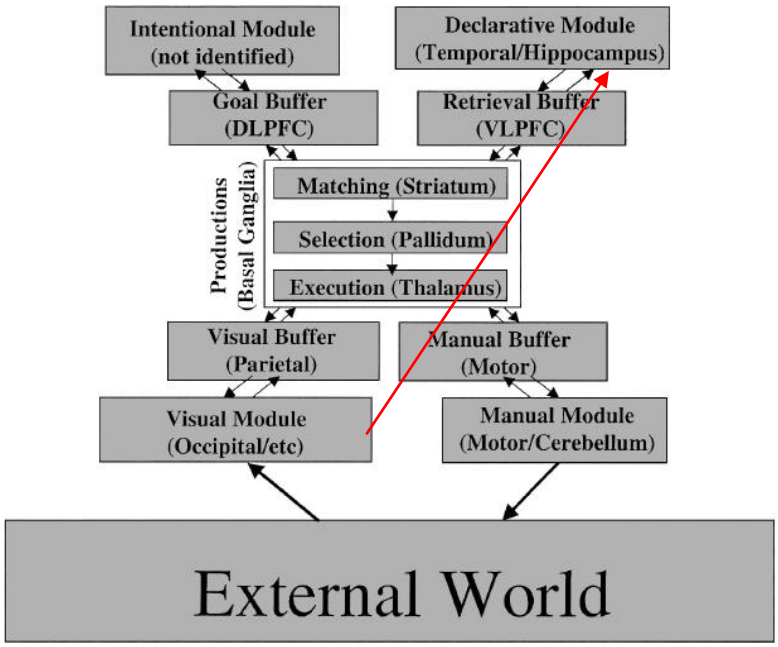


Figure 3. The original figure of the ACT-R 5.0 model (from Anderson et al., 2004) with a red arrow added showing the connection that is suggested by recent research but which it does not currently model.

Human infants, who begin life legally blind, are known to learn to see in an organic way. No one teaches them how to see, so they must learn on their own in a manner consistent with the concept of unsupervised learning. Only when labelling

objects, which is arguably at least partially a semantic memory task, is there any supervised learning involved. Indeed, the phenomenon of remembering what something looks like but forgetting what it is called, such as in the case of names and faces, suggests a separation between the two processes exists at some level. This would appear to leave two possibilities: That the arrangement of Simple and Complex cells is formed from genetic coding, or that they are formed from unsupervised learning. However it could also be the case that it is a combination of both, with the basic hierarchical structure being genetically hard coded, but the specific number of nodes and connections being adjusted during the period of infant neural and visual development. Clearly since the hierarchical architecture is found across human subjects, some degree of genetic coding is involved, and this would justify modelling the architecture as pre-arranged, even if the connections between nodes are not.

The earliest versions of the Neocognitron by Fukushima and Miyake (1982) did just this. The architecture of S-cell and C-cell layers was pre-arranged, and only the weights of the connections between nodes were trained. Initially these weights were entirely trained through unsupervised competitive learning. To improve performance, versions throughout the late 80s and 90s had the S1 and C1 layers trained through supervised learning, the middle layers through unsupervised learning, and the last layer was supervised so as to better identify characters (Fukushima, 2003). It is possible to justify using supervised learning on the last layer since it involves labelling the objects, but to train the earliest layers of the network through supervised learning is to get away from the biological rationale of the architecture, though it appears to have been a necessary shortcut to implementation.

Satoh, Kuroiwa, Aso, and Miyake (1999) worked to make the Neocognitron more rotation invariant as seen in Figure 4. Fukushima (2003) also put significant efforts into advancing a version of the Neocognitron that could perform the character recognition task with great reliability as shown in Figure 5, albeit this system required supervised learning of many of the layers in separate phases that could not be repeated once passed. Essentially this was a trade-off of biological realism for practical functionality. However, a more recent development by Fukushima (2004) restored the unsupervised learning to the initial layers so as to allow incremental learning. It accomplished this partly through a more comprehensive set of inhibitory nodes (see Figure 6), as well as a different learning algorithm. This network proved to be about as capable as its recent predecessor (see Table 1), essentially showing that the biological rationale for unsupervised learning is still applicable.

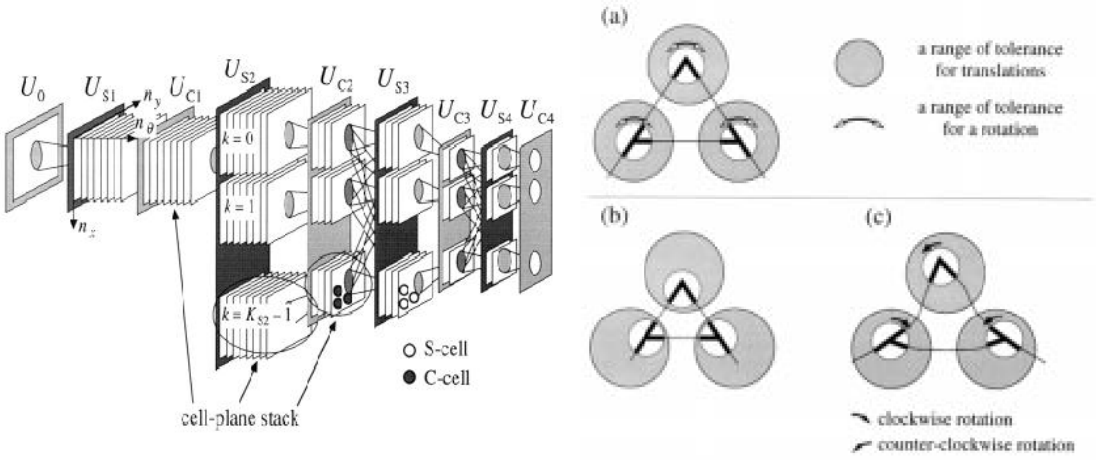


Figure 4. (Left) A visual depiction of the network from Satoh, Kuroiwa, Aso, and Miyake (1999) and examples of its deformation resist properties (right).

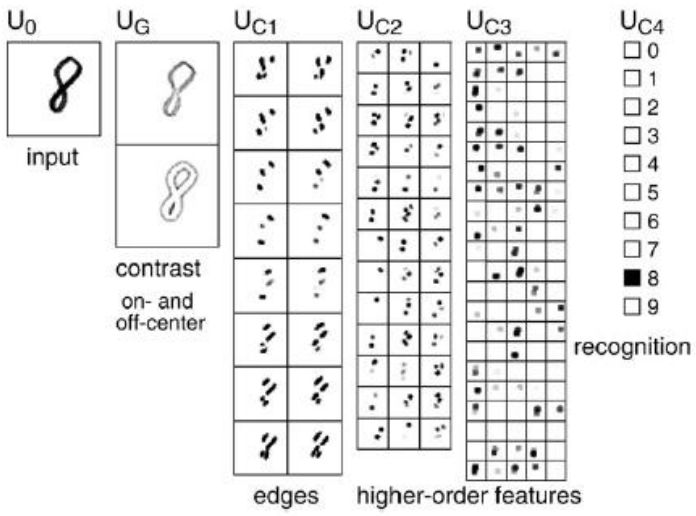


Figure 5. An example of the response of the Neocognitron from Fukushima (2003). The input pattern is recognized correctly as '8'.

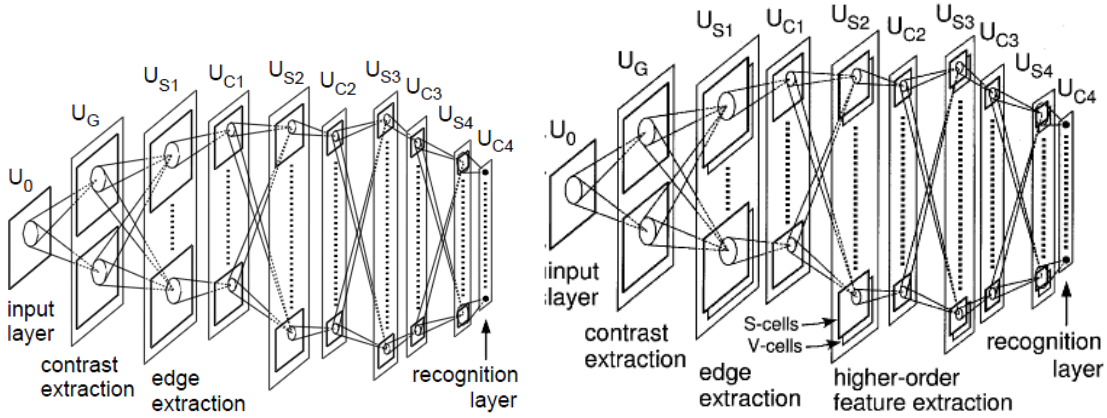


Figure 6. On the left, the modified architecture of the Neocognitron proposed by Fukushima (2003). On the right, the updated version by Fukushima (2004) that incorporates additional V-cell layers at a one-to-one correspondence to S-cell layers.

Table 1

A Comparison of the Recognition Rates of Two Recent Versions of the Neocognitron

Network	Method of constructing layers	Training patterns		Recognition rate (%) for test set (for training set)				Scale of the network		
		Method of presentation	Number of patterns	After	'0' – '4'	'5' – '9'	'0' – '9'	K_{S2}	K_{S3}	K_{S4}
New	Simultaneous	Together (experiment A)	500		96.2 (100)	97.0 (100)	**96.6** (100)	28	86	74
			1000		96.5 (99.8)	98.2 (100)	**97.3** (99.9)	36	104	103
			2000		98.2 (99.9)	98.5 (100)	**98.3** (100)	43	128	146
			3000		98.1 (100)	98.5 (99.9)	**98.3** (100)	45	140	179
		Incremental (experiment B)	500	1/2	98.3 (100)	–	–	24	64	27
				2/2	94.1 (97.6)	96.5 (100)	**95.3** (98.8)	28	78	58
			1000	1/2	98.4 (100)	–	–	26	72	33
				2/2	95.0 (99.0)	97.5 (100)	**96.3** (99.5)	32	85	76
			2000	1/2	99.1 (100)	–	–	32	82	46
				2/2	97.5 (99.3)	98.6 (100)	**98.0** (99.7)	42	108	110
			3000	1/2	99.3 (100)	–	–	35	91	57
				2/2	98.1 (99.1)	98.7 (99.9)	**98.4** (99.5)	48	129	126
Old	Sequential	Together (experiment A)	3000		98.3 (100)	98.9 (100)	**98.6** (100)	39	110	103
	Simultaneous		3000		98.4 (100)	98.2 (99.7)	**98.3** (99.9)	45	241	325

Note. From Fukushima (2004).

Similar work on hierarchical networks for character recognition has also been done at AT&T labs, in the form of Convolutional Neural Networks (CNN) such as the LeNet series (LeCun, Bottou, Bengio, & Haffner, 1998). Convolutional Neural Networks share the hierarchical S-layer and C-layer structure of the Neocognitron, and generalize the delocalizing behaviour as a mathematical convolution. Essentially, the convolutional layers are equivalent to simple cell layers, and the subsampling layers are equivalent to complex cell layers. Thus the lower layers of these two network architectures are almost identical. Where they differ is in the higher layers of the network and in the learning algorithm used (Shouno, 2008). The higher layers of a Convolutional Neural Network are fully connected, making models like LeNet 5 essentially hybrids of the Neocognitron and the more traditional fully connected networks. Furthermore, the learning algorithm used in a Convolutional Neural Network is based on energy optimization, making it more

similar to back-propagation (Shouno, 2008). In any case, Convolutional Neural Networks have been successfully implemented commercially in the banking industry with NCR Corporation's line of cheque recognition systems.

Neocognitron as an Object Recognition Module

Sato and Hagiwara (2003) effectively incorporated the Neocognitron into their Parallel-Hierarchical Neural Network for 3D Object Recognition shown in Figure 7. This network model took the dramatic step of moving away from mere black and white character recognition in favour of identifying specific real-world objects using training data consisting of greyscale still images of said objects taken at different viewing angles (see Figure 8). In essence they extended the Neocognitron to recognizing the two dimensional projection images of three dimensional objects, even after these images were manipulated with affine transformations and obscured by occlusions, as well as when the objects were photographed at different angles.

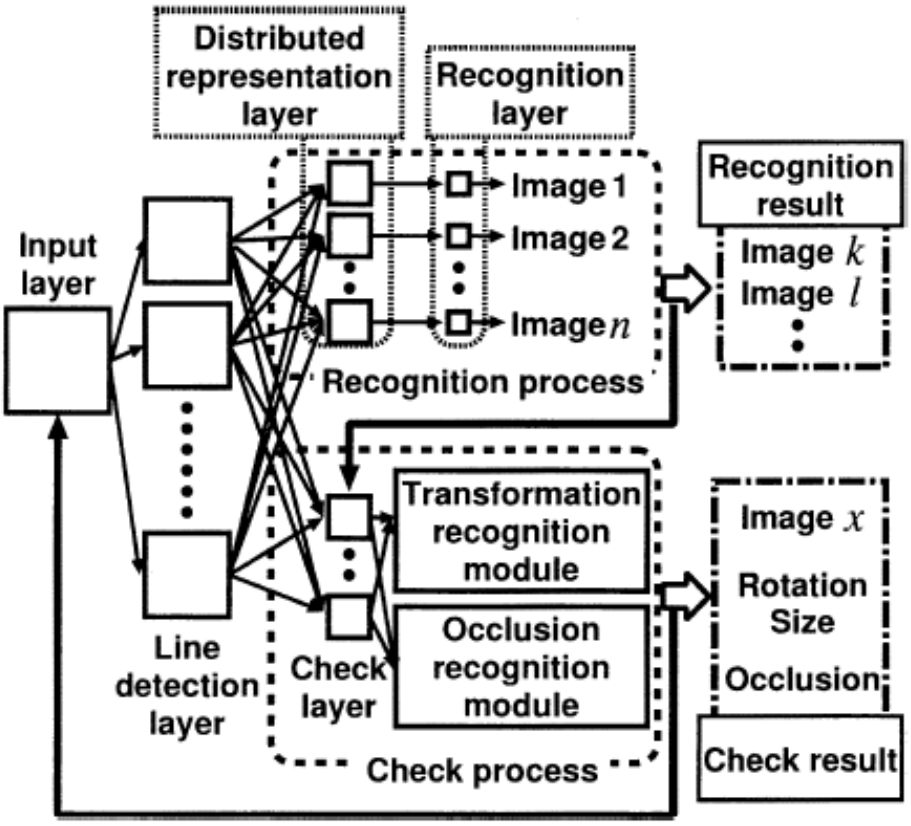


Figure 7. The Parallel-Hierarchical Neural Network for 3D Object Recognition from Sato and Hagiwara (2003). The Neocognitron appears to have formed the basis of the Recognition layer.

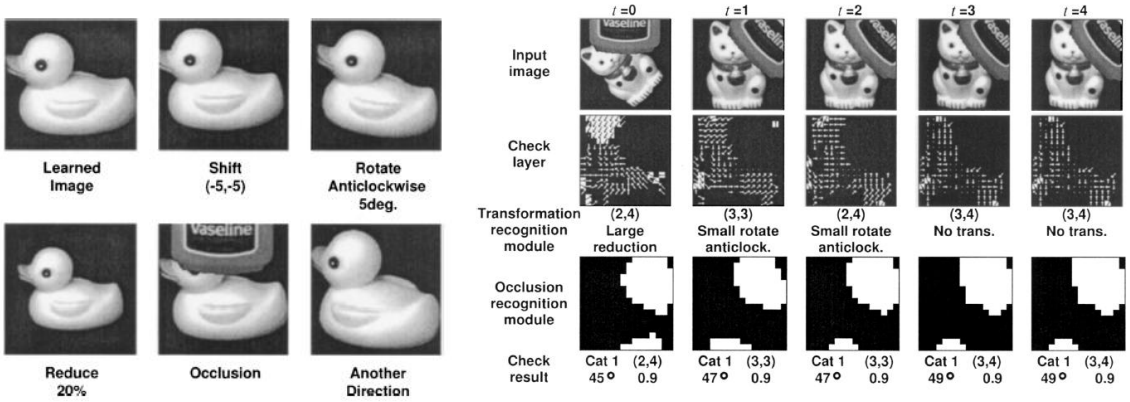


Figure 8. Example input images (left) and corresponding outputs (right) of the network from Sato and Hagiwara's (2003).

Bax, Heidermann, and Ritter (2005) performed a similar feat with their Hierarchical Feed-Forward Network for Object Detection Tasks, which showed that a similar hierarchical network model could be used to identify learned three dimensional objects (see Figure 9) located within a natural environment scene (see Figure 10). They also applied the use of Gabor kernels to convert the greyscale data into an input format that the network could more easily recognize, as shown in Figure 11.

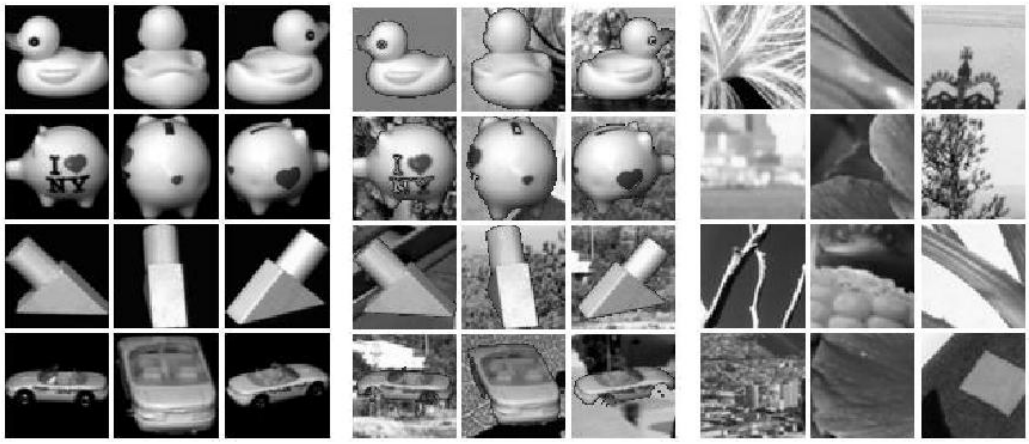


Figure 9. Example images from Bax, Heidermann, and Ritter (2005).

Figure 10. Test images from Bax, Heidermann, and Ritter (2005).

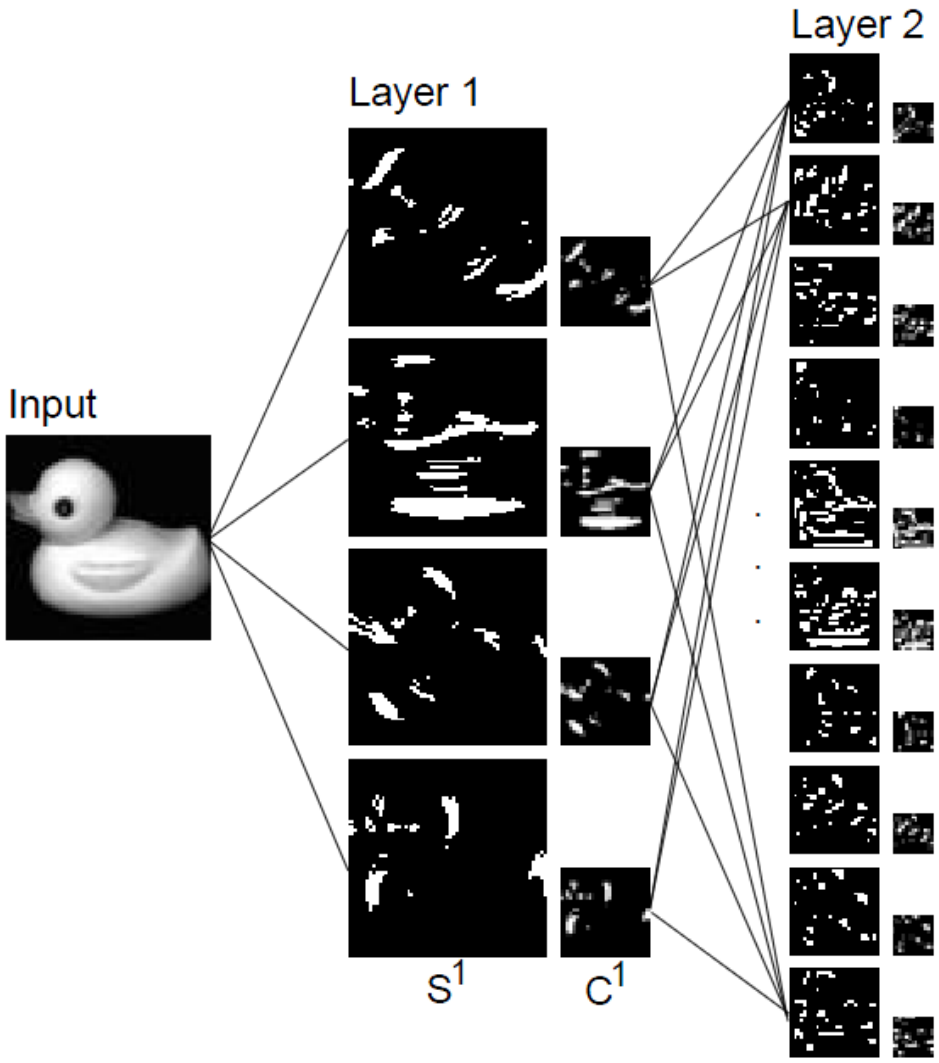


Figure 11. Use of Gabor kernels to convert the greyscale data, from Bax, Heidermann, and Ritter (2005).

In both of these papers the authors show it is possible to identify a 3-D object from a 2-D image, even when somewhat distorted or obscured. In both cases however, the focus is on specific individual objects that the network has been trained to learn and recognize. In reality human visual recognition extends not only to recognition of previously seen objects, but an ability to generalize features so as to categorize similar but not identical objects as being of the same classification. For instance, all apples share

visual features that make them identifiable as apples rather than oranges. This would entail a network that can not only identify the apple object that it was trained on, but also other similar apples, or even 2-D caricatures of apples. Thus, the network could classify a picture of any apple as an apple, or any cat, car, toaster oven, or human face, as an object of that classification.

Kume, Osana, and Hagiwara (1999) provide an example of this in the form of a simple 2D shape recognizing network based on Feature Integration Theory. Essentially it can take simple coloured shapes like triangles and squares and identify them. It combines a Neocognitron for feature recognition, with an LVQ network for colour recognition. While the form recognition of shapes is less impressive in light of the other networks described, it does take things one step further with the ability to recognize colour. An eventual goal would be to have a network able to recognize 3-D objects in all their colourful glory, as colour can be an important feature in identifying objects. At this time however it is not a necessary initial component of our basic network concept, though it should be considered as a possible expansion of the project in the future.

Huang and LeCun (2006) have also shown that Convolutional Networks can perform generic object classification in combination with Support Vector Machines. This is very similar to what we are about to propose, except with a noticeably different approach. It should be noted that error rates remain significantly higher for objects than for characters, so there is room for improvement. Nevertheless, Huang and LeCun's network is one that we will have to compare any successful proposed design to.

Thesis Proposal

It is already proven that a Neocognitron, Convolutional Network, or Hierarchical Network can differentiate between categories of non-identical black and white characters, and can identify identical real world objects in greyscale. Thus it makes sense to hypothesize that a design is possible that can combine the two and differentiate between categories of non-identical real world objects in greyscale. It is therefore proposed that we expand on the Neocognitron and develop a network capable of identifying and classifying images of real world objects as well as artistic depictions categorically. Obviously such a project has the potential to be exceedingly complex. To scale it down to something reasonable, the focus will first be on designing a network that, after being trained on examples of a single type of object, can then detect whether a given input picture contains an object of the type it is designed to classify, something like a simplified Neocognitron with a Perceptron classifying the output, and similar in concept to the work by Banarse and Duller (1997) where they attempt to correctly classify simple iconic hand gestures, albeit using a Neocognitron-based architecture. More complex features such as the occlusion solving methods seen in recent iterations of the Neocognitron (Fukushima, 2007) will not be implemented. Instead the focus should be on a simple, streamlined network that can answer the hypothesis.

As previously mentioned, Huang and LeCun (2006) have already performed a similar feat by combining a Convolutional Neural Network with a Support Vector Machine. While the task they set out to accomplish is very similar, their implementation is fundamentally different from our proposed network in that they depend on the SVM to perform the proper final classification. Our model on the other hand should be simpler

and perhaps not depend on a separate multi-layer supervised learning based classifier module.

A research project is thus proposed and divided into four possible stages of development, with time constraint factors in consideration. Each stage is proposed as a separate subproject, with each subsequent stage building on the previous stage at increasing degrees of complexity.

Proposed Method

To begin this proposed project requires first that an appropriate neural network model be conceptualized. This entails researching in detail the existing work extending the Neocognitron architecture for object recognition, as well as other possible neural network and machine learning alternatives, such as the LeNet series of Convolution Neural Networks. From this an assessment of the limitations of the existing models and potential interactions and similarities can be made. At that point it will be possible to design an initial basic neural network based on the Neocognitron, Convolutional Neural Network, and subsequent parallel-hierarchical models, while also mapping out the more complex modifications necessary for later stages, to be researched in depth when appropriate. This process will also formalize the conceptual documentation into a report. The actual project is divided into the following stages:

Stage 1 – Basic Implementation – Single Greyscale Object Recognition

1. Data Set Construction
    a. Find twenty example images using Google Image Search of generic apples, four oranges, and four cars (for the control images)

b. Standardize the size of each image, cropping so that the object fills the image, and removing external or background noise

c. Convert each image to greyscale (keeping a colour version for future use)

2. Programming the Model

    a. Take the basic design for Stage 1 and write it as a Java application

    b. Test the basic functionality with a single image from the dataset

3. Experimentation

    a. Train the network on sixteen images from the dataset (see Figure 12)

    b. Test the network on the test dataset (see Figure 13):

        i. Four images from the training subset of the dataset

        ii. Four images from the dataset not in the training set

        iii. Four images of objects not in the dataset that are similar (oranges)

        iv. Four images of very dissimilar objects (cars)

Figure 12. Example images from the 'Apple' category training data set.

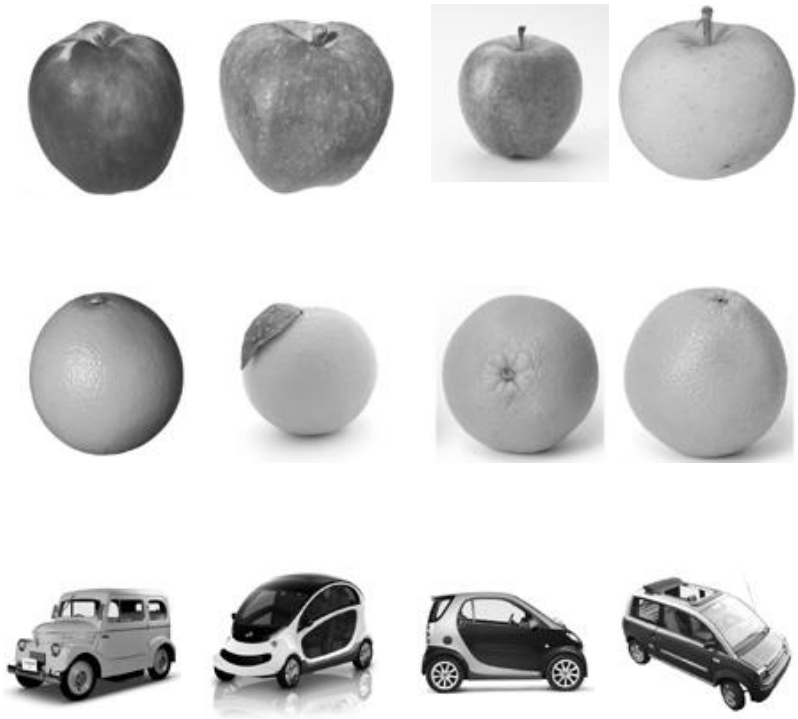


Figure 13. Example images from the 'Apple', 'Orange', and 'Car' test data sets.

Stage 2 – Advanced Implementation – Multiple Greyscale Object Recognition

1. Data Set Extension
    a. In addition to the existing dataset, search for an additional sixteen oranges, sixteen cars, twenty cats, twenty dogs, and four houses
    b. Standardize the size of each image, cropping so that the object fills the image, and removing external or background noise
    c. Convert each image to greyscale (keeping a colour version for future use)
2. Modifying the Model
    a. Expand on the existing stage 1 Java application, reintegrating the functionality to include the four or more other objects
    b. Test the functionality with the stage 1 dataset, to ensure that the previous functionality remains

3. Experimentation
    a. Train the network on sixteen images from each of the five image categories of the dataset
    b. Test the network on:
        i. Four images from the training set of each image category
        ii. Four images not included in the training for each image category
        iii. The four house images (control)

Stage 3 – Colour Implementation – Multiple Full Colour Object Recognition

1. Dataset Recovery
    a. Take the previously saved colour versions of each image of the dataset from Stage 2 and group as a new dataset
2. Modifying the Model
    a. Expand on the Stage 2 Java application, restructuring colour processing elements into the network
    b. Test the functionality with the Stage 1 dataset
3. Experimentation
    a. Repeat the Stage 2 experimentation step using the new dataset

Stage 4 – Real World Implementation – Object Recognition w/ Scene Noise

1. Dataset Recovery
    a. Take the previously saved post-cropping versions of each image with background scenery or noise still present as new dataset
2. Modifying the Model

a. Expand on the Stage 3 Java application, adding noise cancellation processing elements to the network
b. Test the functionality with the Stage 1 dataset

3. Experimentation
    a. Repeat the Stage 3 experimentation step using the new dataset

Results

A proof of concept prototype was written up in Java to attempt to map how the actual implementation of the design would have to be handled, and identify some of the difficulties involved**.** Image sets were selected for the most basic implementation, and the initial network architecture developed. A Java Class was written up in which the network simulation and algorithms for importing inputs, training the network, and testing the network would be included as methods within the program. The network was simulated as an object instantiation of the Network subclass, within which the Plane subclass represented the arrays of nodes, and the nodes themselves implemented in the Node subclass. This design was meant to adhere to the object-oriented paradigm of the Java programming language, as a useful means of emulating a neural network conceptually, an approach which we have had success with in the past. After some experimentation, several fundamental issues were identified. These would require additional testing prior to implementation of the originally proposed method in order to validate the design. Only after such uncertainties are resolved would it be possible to move forward with an optimal design for the network.

Discussion

Conceptual Difficulties and Design Challenges

After initial experimentation with the prototype, it appears that before the proposed prototype model can be successfully implemented and tested there are three major hurdles that must be overcome. These three problems represent design and engineering issues that may require significant revisions to the existing network models of the Neocognitron and the Convolutional Neural Network, which the network design incorporates features of. They are the greyscale conversion problem, the network dimensions configuration problem, and the effective learning algorithm problem, respectively.

The Greyscale Conversion Problem

Greyscale images are something of a problem for the original Neocognitron, which has input parameters based on very simple binary black/white brightness discrimination (Banarse & Duller, 1997). Such a brightness contrast methodology is sufficient for the relatively simple problem of recognizing the two-dimensional silhouettes that alphabetical and numerical characters are constructed as, but real world images possess complex optical interactions such as depth and shading (see Figure 14). Whereas a single pixel cell in a black and white image has only two possible outputs, a greyscale sensitive cell would have almost a continuum of values. Such a continuum can be made discrete computationally as a range of anything from 16 to 16 million values. A very common choice is 256, which is used in most basic RGB representations. The main issue with this is that a value range greater than 0 and 1 implies a potential multiplication

of the number of features which must be differentiable by the network. Whereas a single 3 x 3 receptive field has only a certain number of combinations of identifiable features given two possible values per pixel, 16 or 256 possible values produces a combinatorial increase in features, and thus a corresponding increase in the size of the array in each layer of the network.

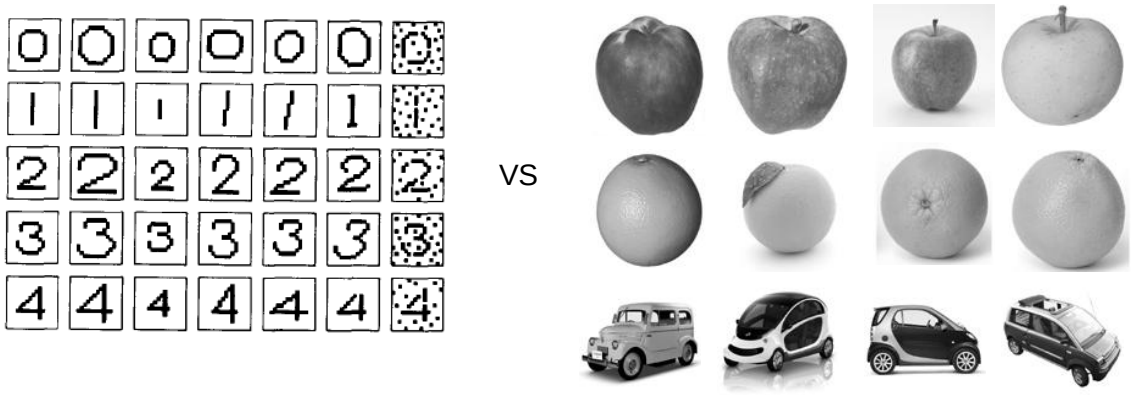


Figure 14. A comparison of the black and white input data from the traditional Neocognitron (left) from Fukushima and Miyake (1982) with the greyscale input data of the proposed network (right).

A possible solution to this issue is to actually take a page once again from neuroscience research, where the concept of on-centre and off-centre receptive field cells is well documented (Anderson, 2000, p. 41). In effect, rather than trying to identify features of shading, an intermediate layer between the input layer and the S1 layer can be inserted that functions as an edge detection layer, converting the greyscale into a brightness discrimination pattern. The edge detection layer uses the center-surround organization that is responsible in humans for our ability to focus on contrast rather than absolute brightness, which can easily vary due to environmental conditions such as the day-night cycle. Regardless of how much light we shine on an apple, it should be identifiable as long as there is enough to discern differences in intensity on the surface of

the object. This kind of intermediary layer has been tried in the most recent iterations of the Neocognitron (Fukushima, 2003), but not tested on greyscale objects, only the usual character recognition tasks.

Alternatively, Bax, Heidermann, and Ritter (2005) have proposed and shown successfully the use of Gabor kernel functions to perform a similar feat. As Gabor filters are known to be based off of mathematical convolutions, and it has already been shown with Convolutional Neural Networks the biological plausibility of performing convolutions in the connectionist paradigm, it is possible that Gabor kernels could hypothetically be implemented as part of the architecture of such a network.

Testing both of these adaptations for their effectiveness is beyond the original scope of the project, but likely necessary for an optimal design.

The Network Dimensions Configuration Problem

Another concern is the logic behind the number of nodes in each layer and each array of the network. At first glance the dimensions of these layers and arrays can seem almost arbitrary, but they are actually based on a rationale. The degree of separation between receptive fields of cells affects the possible layout configurations, and in many cases the question becomes what the appropriate Receptive Field shape and size ought to be. The original Neocognitron used 16 x 16 pixel b/w characters, limiting the size of the associated layers, but more advanced networks have used inputs consisting of the COIL-20 and COIL-100 image sets, which use 64 x 64 pixel and 128 x 128 pixel greyscale images respectively. It is not possible to arbitrarily set a network to use, for instance, 100 x 100 pixel test images, rather the input layer dimensions will depend heavily on the configuration of connections in the subsequent layer.

There are several possible ways to understand this. From a top down perspective, we could simply view the hierarchical structure as an exponential pyramid, with a spreading receptive field of 3 x 3. Since $3^4 = 81$, this suggests a hierarchical structure of 81 x 81 in the input layer, 27 x 27 in the first processing layer, 9 x 9 in the second layer, and finally 3 x 3 in the third processing layer, before coalescing into a straightforward 1 x 1 output layer. Despite getting us to an 81 x 81 input layer, which is close to the desired 100 x 100 size, this example is an unrealistic oversimplification. It assumes no overlap between the receptive fields of nodes, and is simply not how the Neocognitron, Convolutional Neural Networks, or most Hierarchical networks work. Rather, the Simple Cell or Convolutional layers tend to have considerable overlap between receptive fields, while the Complex Cell or Subsampling layers do not have overlap (see Figure 15). In nearly all hierarchical networks studied S-cell layers and C-cell layers alternate for each level of the hierarchy.

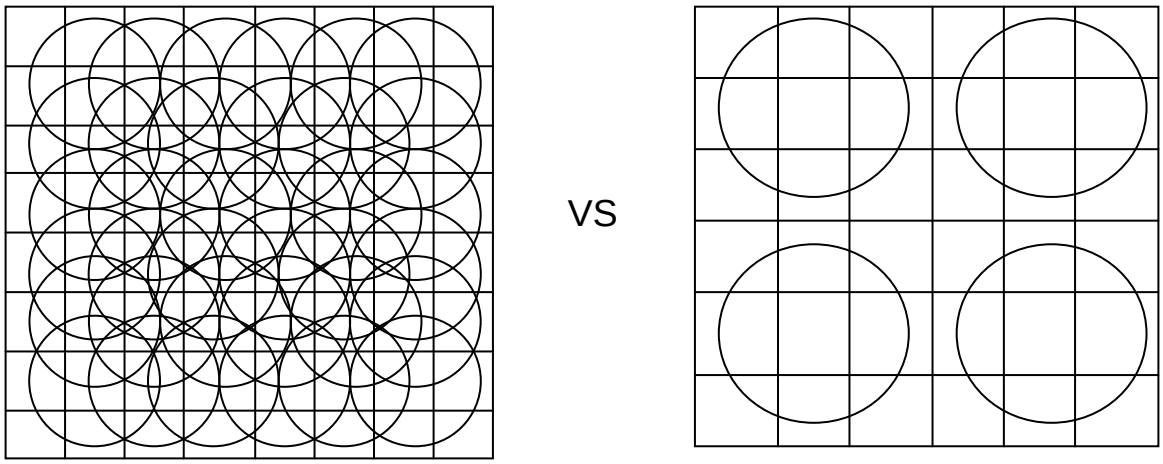


Figure 15. Circles represent the receptive fields of the cells of the layer subsequent to the one represented by the square lattice. On the left, an 8 x 8 input layer feeds into a 6 x 6 layer using receptive fields of size 3 x 3 with an offset of 1 cell. On the right, a 6 x 6 input layer feeds into a 2 x 2 layer using receptive fields of size 3 x 3 with an offset of 3 cells.

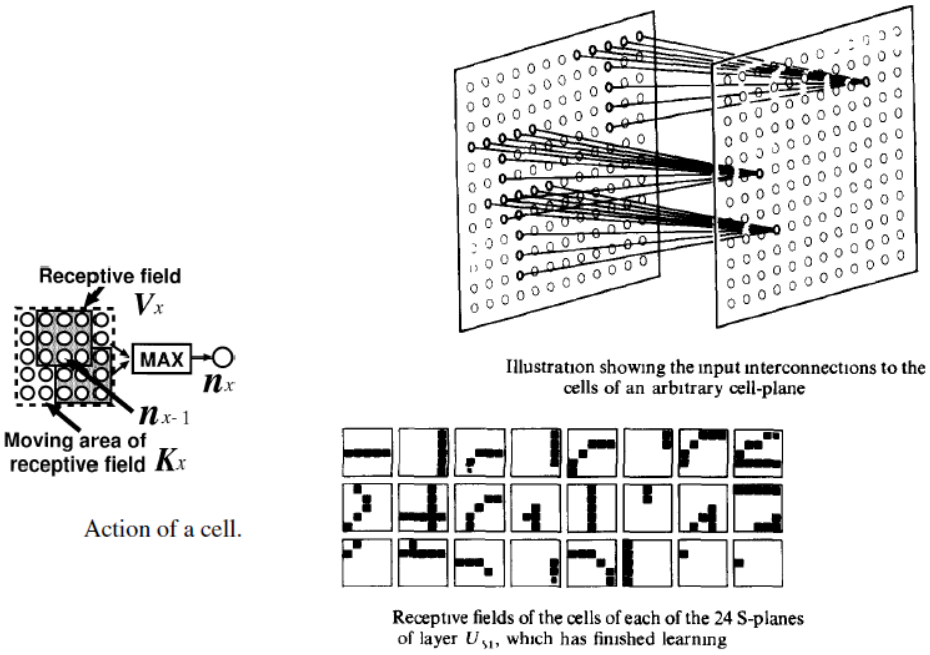


Figure 16. Several figures representing various arrangements of receptive fields from different network models. Left figure from Sato and Hagiwara (2003). Right figures from Fukushima and Miyake (1982).

Thus the relationship between layers can be described as follows. To calculate the reasonable dimensions of a square layer from either its previous layer (or next layer) in the hierarchy requires at least some of the following variables to be assigned. Let x be the width of the previous (or current) square layer. Let y be the width of the current (or next) square layer. Let r be the width of the square receptive field of nodes in the previous (or current) layer to each current (or next) layer node, and f be the offset distance between the receptive fields of adjacent nodes in the current (or next) layer. The relationship between these variables is best described by the equation:

$$y = \frac{x - (r - f)}{f}$$

Where, $x \geq y$, $x \geq r \geq f$, and $f > 0$

This equation allows us not only to determine a reasonable configuration given the input layer, but enables us to reverse engineer some of the previously used arrangements to determine variables that weren't mentioned explicitly in the papers. Though the equation is limited to immobile square receptive fields, which are not always the choice of some of the fancier designs (see Figure 16). It is also limited to offsets greater than zero, though an offset of zero is logically equivalent to a one-to-one correspondence network, which obviously has no offset, a receptive field of 1, and equal sized x and y layers. Furthermore, many designs function such that the input square is edged with inactive pixels in order to centre the receptive field on each input node. This explains why many of the earlier builds have layers that do not reduce into a smaller array in the subsequent layer, but retain identical dimensions. In those cases, the input layer is effectively two nodes larger in width, except that these edge nodes are always outputting zero. To make our equation work with these cases, we add the receptive field width minus one to the input layer width.

Fukushima and Miyake (1982) originally designed the network along the lines of a 16 x 16 input square. Layer S1 was 16 x 16 x 24, with 24 representing the size of the array. Layer C1 was 10 x 10 x 24, S2 was 8 x 8 x 24, C2 was 6 x 6 x 24, S3 was 2 x 2 x 24, and C3 represented the output layer of 1 x 1 x 24. Receptive fields were mostly 5 x 5 squares. The offset distance was not mentioned but can be calculated with the help of our equation. Since the receptive field size is 5, the offset must equal an integer between 1 and 5. This model ultimately was not effective at recognizing ten characters due to the 24 array limit imposed by the authors due to memory constraints turning out to be insufficient. Our equation furthermore is able to account for all the layer size reductions

except for that between S1 and C1, with which it only makes sense if layer C1 is actually 12 x 12 (with the outermost edge of the square being blank). As in 12 = (16 – (5 – 1)) / 1.

An updated and somewhat more successful design was published in 1988 that increased the input layer to a 19 x 19 square (Fukushima, 1988). In this design, Layer S1 was 19 x 19 x 21, C1 was 11 x 11 x 21, S2 was 11 x 11 x 27, C2 was 11 x 11 x 27, S3 was 7 x 7 x 5, and C3 was 1 x 1 x 5. At this point the total number of nodes in the network was 41,399. All that was required for a network that was only trained to recognize the five characters 0, 1, 2, 3, and 4. Getting these dimensions to mesh with our equation again requires a considerable adjusting of values. First, it would have to be assumed that the receptive field of nodes in layer C1 is a 7 x 7 square (which makes a bit of sense from the fact that S3 and C3 go from 7 x 7 to 1 x 1 in a single step), and that the C1 layer is actually 13 x 13 including an empty margin of zero nodes along the edges.

One of the more recent updates to the Neocognitron that actually showed a very high recognition rate was designed with the following dimensions: U0: 65 x 65; UG: 71 x 71 x 2; US1: 68 x 68 x 16; UC1: 37 x 37 x 16; US2: 38 x 38 x KS2; UC2: 21 x 21 x KC2; US3: 22 x 22 x KS3; UC3: 13 x 13 x KC3; US4: 5 x 5 x KS4; UC4: 1 x 1 x 10 (Fukushima, 2003). This composition is exceedingly irregular in that some layers actually increase in size from the lower ones. Our equation in its current form has exceptional difficulty explaining the dimensions chosen in this version of the Neocognitron, since the network seems to violate the principle of the offset never exceeding the size of the receptive field, which would cause gaps in input. We are at a loss to explain this, which is particularly unfortunate considering how much more effective this version of the network apparently is.

LeNet-5 has its own set of configurations with the initial input layer being a 32 x 32 square of input nodes, convolutional layer C1 (equivalent to the Neocognitron's S1) being 28 x 28 x 6, subsampling layer S2 (equivalent to the Neocognitron's C1) being 14 x 14 x 6, convolutional layer C3 (equivalent to the Neocognitron's S2) being 10 x 10 x 16, and subsampling layer S4 (equivalent to the Neocognitron's C2) being 5 x 5 x 16 (LeCun et al., 1998). The remaining layers of the network were fully connected with C5 being 120 nodes, F6 being 84 nodes, and the output layer being 10 nodes. Our equation actually has no difficulty predicting this arrangement given the variables described in the paper. LeNet-5 consistently follows a pattern of using a 5 x 5 receptive field and offset of one node for convolutions, and a 2 x 2 receptive field and offset of two nodes for subsampling. Thus the dimensions of layer C1 are explained as $28 = (32 - (5 - 1)) / 1$, layer S2 as $14 = (28 - (2 - 2)) / 2$, layer C3 as $10 = (14 - (5 - 1)) / 1$, and layer S4 as $5 = (10 - (2 - 2)) / 2$.

So for our prototype network, given our equation, an alternative arrangement was devised by calculating the hierarchical connection pyramid from the top, and assuming a receptive field of 3 x 3 as a general rule. Layer C4 was proposed as a 1 x 1 square, representing the top layer output node that would output the similarity value of a test input to the training data, essentially a simple Perceptron. Layer S4 was calculated as a 3 x 3 square by supplying the width of its output layer as 1, with a receptive field width of 3 and an offset of 3, giving us the formula $1 = (3 - (3 - 3)) / 3$. Layer C3 was calculated as a 5 x 5 square using an output layer width of 3, a receptive field width of 3, and an offset of 1, to give us $3 = (5 - (3 - 1)) / 1$.

Layer S3 was calculated as a 15 x 15 square based on the formula 5 = (15 – (3 – 3)) / 3. Layer C2 was calculated as a 17 x 17 square based on the formula 15 = (17 – (3 – 1)) / 1. Layer S2 was calculated as a 51 x 51 square based on the formula 17 = (51 – (3 – 3)) / 3. Layer C1 was calculated as a 53 x 53 square based on the formula 51 = (53 – (3 – 1)) / 1. Layer S1 was calculated as a 159 x 159 square based on the formula 53 = (159 – (3 – 3)) / 3. The Input Layer was calculated as a 161 x 161 square based on the formula 159 = (161 – (3 – 3)) / 3. Given this arrangement, the input images should be 161 x 161 pixel squares. Testing this configuration however requires that the other problems with implementation be solved before proceeding.

The Effective Learning Algorithm Problem

An additional concern involves the question of what manner of learning algorithm is appropriate. As mentioned previously, early versions of the Neocognitron were designed with an algorithm that performed simultaneous unsupervised learning on the entire network, but at a relatively slow rate of learning. More recent versions have used a much faster algorithm that required layers to be trained in sequence, and limited the capacity of the network to learn in increments. This sequential training meant it was necessary to train the S1 layer on separate edge extraction trial sets, which is problematic because it doesn't fit with the biological model, in which the human visual system learns directly from a single set of real world stimuli. Children aren't trained to see with pictures of lines after all. They are somehow able to extract features from fully featured objects such as faces.

An alternative design that restored the functionality for simultaneous learning was one of the most recent designs by Fukushima (2004) and the one that will be looked at

most closely as our algorithm of choice. The advantages of simultaneous learning must be offset by the additional complexity required to make it viable. In the aforementioned network, V-cell inhibitory nodes were increased from one V-cell array per layer of S-cells to one V-cell per S-cell, a dramatic increase in complexity. It is also somewhat problematic that this version of the Neocognitron is also the one that our previously described equation for calculating reasonable network dimensions is unable to verify.

This arrangement is also increasingly complex in comparison to Convolutional Neural Networks, which do not use inhibitory nodes. Instead, LeNet-5 operates using a Radial Basis Function network as the output layer, and its Mean Square Error function is minimized by the learning algorithm used by LeCun et al. (1998), which performs competitive learning using a Maximum A Posteriori (MAP) criterion to compare target classes. MAP is a commonly used approximation technique in A.I., used as a simpler alternative to Bayesian learning (Russell & Norvig, 2003, p. 714). It is similar in function to a Winner-Take-All network, which is a category of neural networks used for unsupervised learning (Mehrotra, Mohan & Ranka, 1997, p. 161). However, this algorithm depends on the top layers of the Convolutional Neural Network being a fully connected RBF network. It would be preferable to maintain the hierarchical simplicity of the Neocognitron for the sake of biological rationality, though it could be argued that the RBF network layers represent part of the semantic memory system and are therefore biologically plausible as simulating the part of the hippocampus connected to the inferotemporal (IT) cortex. In any case, deciding which learning algorithm to use will greatly impact the overall architecture of the network, and must be considered carefully.

The Little Problem of Complexity

In addition to these three problems, a potential conundrum exists with regards to realistic implementation of the proposed project. If the network is too simple it may not be able to perform the task as intended simply due to a lack of resources, but if the network is too complex, it may be beyond available computational resources, or see a combinatorial increase in terms of possible configurations that must be tested. The original Neocognitron had approximately 11,320 nodes and about 424 individual weight parameters. Assuming the same basic configuration our proposed network could have over 1,370,000 nodes and 27,000 individual weight parameters. Thus, for the purposes of the initial project, it makes the most sense to start off with as simple a model as feasible.

Considerations Regarding Current and Future Research

In a sign of the increasing rapidity of scientific progress in this day and age, the originally proposed project may already be potentially trivial or obsolete due to recent research. Kirstein, Wersing, and Korner (2008) in a rather interestingly entitled paper: A Biologically Motivated Visual Memory Architecture For Online Learning of Objects, showed that the proposed research is well within the grasp of a lab with sufficient resources to undertake it. Not only are they able to implement an object recognition network in full colour, but they are also proposing a form of object memory representation (see Figure 17). Though, like many of the network models previously discussed this network identifies specific objects rather than object categories.

It should be noted that the computational model by Kirstein, Wersing, and Korner (2008) uses and presupposes the outdated Modal Model of memory. The Modal Model

was a popular group of theories that described memory as consisting of separate short term and long term stores, also known as primary and secondary memory, but the reality is that this model has been abandoned by the majority of memory researchers due to irreconcilable problems (Neath & Surprenant, 2003, p. 66). It would be advisable then to update the Kirstein, Wersing, and Korner architecture with more up-to-date memory theories, such as rebasing their model around theories of Working Memory or the Feature Model. For that matter, why not incorporate research done on BEAGLE and render the semantic memory system as a semantic network-based holographic lexicon (Jones & Mewhort, 2007).

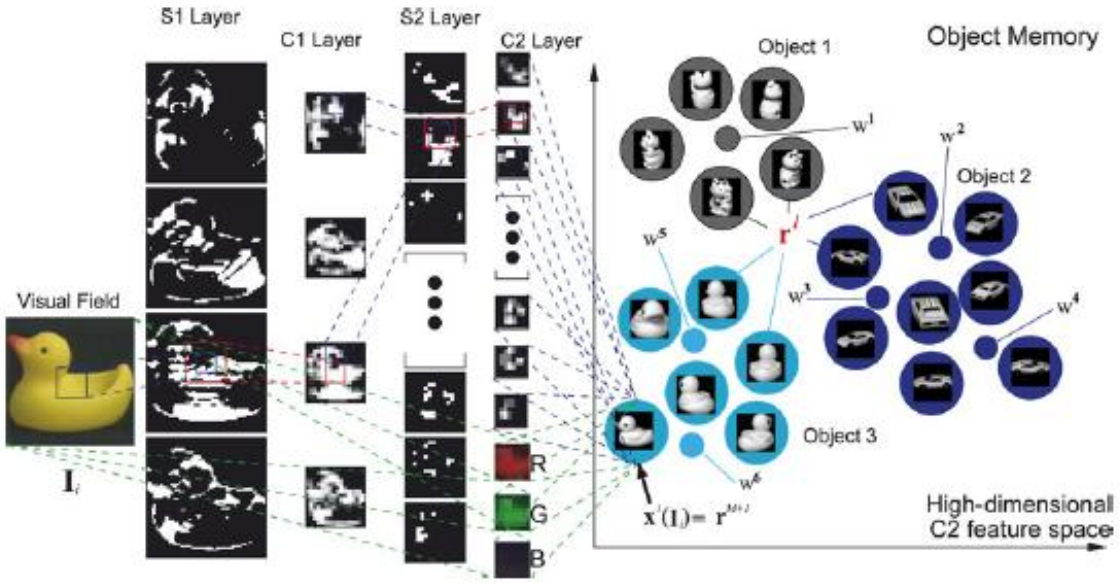


Figure 17. A diagram of the model from Kirstein, Wersing and Korner (2008) that utilizes an architecture similar to the proposed network and also implements the second stage of Object Representation in the form of a semantic memory space.

Furthermore, where LSA and BEAGLE are based on word association, a visual semantic memory would be based on visual similarity of objects. Visual semantic memory might therefore cluster the concept of an apple more closely with an orange but

also with a baseball because of their similar shape. These objects would in turn all cluster around the perceptual concept of a sphere, a hypothesis perhaps worth testing in the future.

Also notable is the fact that both Convolutional Neural Networks, which are used in object recognition, and BEAGLE, which is a semantic memory model based around words, perform mathematical convolutions as part of their various methods of encoding information. This suggests that convolutions may be a general architectural principle used throughout the brain for information compression and memory storage. This is yet another hypothesis to consider for future research.

And even while the original project set out now appears to be somewhat less remarkable, it is nevertheless apparent that the Recognition of Object Categories remains an under-evaluated concept and the prototype proposed could still be useful at some point to compare with and possibly improve upon the performance of other architectures such as the design by Huang and LeCun (2006).

Evolving Neural Networks and Other Possibilities

The Neocognitron, LeNet 5, and most of the hierarchical neural networks discussed contain a fixed number and arrangement of nodes and connections at creation. Yet from a biological and developmental psychology perspective, it would make more sense that neural networks would possess the capacity to grow and expand. As mentioned prior, human infants, who begin life legally blind, are known to learn to see in an organic way. They also develop the ability to remember things around eight months into development, in effect learning to remember. Clearly, neuronal growth is an important part of the biological model of the human brain, but few neural networks

bother to model this phenomenon. The Growing Neural Gas network is a major exception that coincidentally is relevant as well because it also performs unsupervised competitive learning. Recent work has looked at using this algorithm in image classification in combination with Convolutional Neural Networks (Dong & Izquierdo, 2008). This is almost certainly an area worth looking at for potential improvements to the network that are also biologically reasonable.

Having mentioned the idea of growing neural networks that can change the actual structure of themselves, it seems appropriate to propose we take this a step further. While neural network growth in real life animals and humans is decided partly by experiential stimulus, it is fundamentally influenced by the genetic base from which the human brain is based. That the brains of the vast sample of humans studied have roughly the same shape and structure suggests this. An interesting experiment then would be to attempt to simulate the evolution of the brain by way of creating a sea of individual networks attached to simulated agents in a simulated world. Perhaps genetic algorithms could be applied to this simulation, similarly to the manner in which they have been applied to other networks such as Backpropagation (Browse, Hussain, & Smillie, 1999). This would attempt to see whether networks can adapt and evolve for particular tasks, perhaps even tasks that we consider to require conscious thought to execute. The mind is ultimately a device that adapts and learns. It can adapt at many different levels. At the level of the daily experience, it adapts by the modification of connections between neurons. At a much more long-term level, it can adapt by modification of individual elements of the structure, by growth of neurons. And at the highest intergenerational, de-individualized level it can adapt by genetic changes that affect the very blueprints of the

mind. Where the first two processes are somewhat internally directed while influenced by external stimuli, this last process is a result of the completely external process of evolution, and as such is much slower and less certain.

Other conceptual developments should also be incorporated into future research. Khan and Yun (1997) have shown that holographic associative memory has the potential to enfold the massive search space of images into a more manageable dimensionality. The properties of holographic memory are also known to resemble the mathematical principles of light holography (Jones & Mewhort, 2007), which among other things means that loss of information will result in a gradual deterioration of the entire image, rather than loss of specific information. This seems to be equivalent to the gradual degradation of memory seen in the human aging process, suggesting biological plausibility. Given the holographic behaviour of BEAGLE, which is a semantic network that uses convolutions to encode word order information, it may be an interesting to inquire as to whether Convolutional Neural Networks also show holographic phenomena.

Similarly, other perhaps more unconventional recent research into neural networks with potential implications in object recognition, such as the work on Wavelet Neural Networks by Pan and Xia (2008) should be kept in mind when looking at ways to improve on the model. Siegelmann (2003) has shown that Analog Recurrent Neural Networks are hyper-computational, that is to say, they are able to perform beyond the traditional mathematical limitations of a Turing Machine. This appears to be done practically by using real numbers of linear precision as weights. Though not necessarily relevant to our immediate goals, exploration into the potential advantages of this could be worth pursuing. Recurrent networks in general bear a striking resemblance to the

Selective Attention Model also worked on by Fukushima (1986), which could be considered a subtype of recurrent network. Compared to the computer vision problem, the computer attention problem is far less explored and could yield interesting developments relating to memory and decision making theory in artificial intelligence.

Conclusion

Proposed here was a neural network that would merge the categorizing capabilities of the Neocognitron and Convolutional Neural Networks with the real world object recognition capacity of recent advances in parallel hierarchical network design to create a network model that could successfully discriminate between different types of objects. While a working prototype was not ultimately completed, several issues surrounding its potential development were considered, including an equation for specifying the logical dimensions for a hierarchical network architecture given appropriate parameters.

In addition several papers showing promising research have been discussed, and possible directions for further efforts in this area pointed towards. The overall idea of using biological plausibility as a criterion for research focus remains consistent with the successful work in the literature. While much of the research has focused on the first stage of object representation, there has been some effort to expand into and combine research with work done on the second stage. A working visual semantic memory connectionist model appears to be a very real possibility in the near future.

The attempt by humans to understand the mind is essentially a form of metacognition. While historically such efforts have been the purview of philosophers rather than scientists, recent advances and developments have enabled researchers to

explore, test and validate models of the human mind, and in so doing, demystify what has become one of the final frontiers of human scientific understanding.

References

Anderson, J. R. (2000). *Cognitive psychology and its implications* (5th ed.). New York: Worth Publishers.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, *111*(4), 1036-1060.

Banarse D. S., & Duller, A. W. G. (1997). Deformation Invariant Visual Object Recognition: Experiments with a Self-organising Neural Architecture. *Neural Computing & Applications*, *6*, 79-90.

Bax, I., Heidemann, G., & Ritter, H. (2005). A Hierarchical Feed-forward Network for Object Detection Tasks. *Proceedings of SPIE, 5818*, 144-152.

Braitenberg, V. (1984). *Vehicles: Experiments in Synthetic Psychology*. The MIT Press.

Browse, R. A, Hussain, T. S., & Smillie, M. B. (1999). Using attribute grammars for the genetic selection of backpropagation networks for character recognition. *Proceedings of SPIE - The International Society for Optical Engineering*, *3647*, 26-34.

Dong, L., & Izquierdo, E. (2008). A topology preserving approach for image classification. *2007 8th International Workshop on Image Analysis for Multimedia Interactive Services*, 25-28.

Fukushima, K. (1986). A neural network model for selective attention in visual pattern recognition. *Biological Cybernetics*, *55*, 5-15.

Fukushima, K. (1988). A neural network for visual pattern recognition. *Computer*, *21*(3), 65-75.

Fukushima, K. (2003). Neocognitron for handwritten digit recognition. *Neurocomputing*, *51*, 161–180.

Fukushima, K. (2004). Neocognitron capable of incremental learning. *Neural Networks*, *17*, 37–46.

Fukushima, K. (2007). Recent advances in the Neocognitron. *Neural information Processing: 14th international Conference, ICONIP 2007, Kitakyushu, Japan, November 13-16, 2007, Revised Selected Papers, Part I*, 1041-1050.

Fukushima, K., & Miyake, S. (1982). Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position. *Pattern Recognition*, *15*(6), 455-469.

Huang, F. J., & LeCun, Y. (2006). Large-scale learning with SVM and Convolutional Nets for Generic Object Categorization. *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *1*, 284-291.

Jones, M. N., & Mewhort, D. J. (2007). Representing word meaning and order information in a composite holographic lexicon. *Psychological Review*, *114*(1), 1-37.

Khan, J. I., & Yun, D. Y. Y., (1997). A parallel, distributed and associative approach for searching image patterns with holographic dynamics. *Journal of Visual Languages and Computing*, *8*, 303-331.

Kirstein, S., Wersing, H., & Korner, E. (2008). A biologically motivated visual memory architecture for online learning of objects. *Neural Networks*, *21*, 65-77.

Kume, H., Osana, Y., & Hagiwara, M. (1999). Solving the Binding Problem with Feature Integration Theory. *IEEE*, *6*(1), 200-205.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324.

Mehrotra, K., Mohan, C. K., & Ranka, S. (1997). *Elements of artificial neural networks*. Cambridge, MA: The MIT Press.

Neath, I., & Surprenant, A. M. (2003). *Human memory: An introduction to research, data, and theory* (2nd ed.). Belmont, CA: Wadsworth.

Pan, H., & Xia, L. Z. (2008). Efficient Object Recognition Using Boundary Representation and Wavelet Neural Network. *IEEE Transactions on Neural Networks*, *19*(12), 2132-2149.

Pylyshyn, Z. W. (1998). *What is Cognitive Science*?, Retrieved from: http://ruccs.rutgers.edu/ftp/pub/papers/ruccsbook.PDF.

Pylyshyn, Z. W. (2003). Return of the mental image: Are there really pictures in the brain?. *Trends in Cognitive Science*, *7*(3), 113-118.

Russell, S., & Norvig, P. (Eds.). (2003). *Artificial intelligence: A modern approach* (2nd ed.). Upper Saddle River, NJ: Pearson Education.

Sato, N., & Hagiwara, M. (2003). Parallel-Hierarchical Neural Network for 3D Object Recognition. *Systems and Computers in Japan*, *35*(1), 1-12.

Shouno, H. (2008). Recent studies around the Neocognitron. In M. Ishikawa, K. Doya, H. Miyamoto, and T. Yamakawa, (Eds.), *Neural information Processing: 14th international Conference, ICONIP 2007, Kitakyushu, Japan, November 13-16, 2007, Revised Selected Papers, Part I,* (Lecture Notes In Computer Science, vol. 4984, pp. 1061-1070). Berlin: Springer-Verlag.

Siegelmann, H. T., (2003). Neural and super-Turing computing. *Minds and Machines, 13*, 103-114.

Thagard, P. (1996). *Mind: Introduction to cognitive science*. Cambridge, MA: The MIT Press.

Wolfe, J. M., Kluender, K. R., Levi, D. M., Bartoshuk, L. M., Herz, R. S., Klatzky, R. L., & Lederman, S. J. (2006). *Sensation & Perception*. Sunderland, MA: Sinauer Associates.