The split-fovea model of visual word recognition and the viewing position effect.

Gil Orazi

Master of Science

School of Informatics

University of Edinburgh

2003

Abstract

Connectionist networks have been used extensively in the domain of visual word recognition and reading. This paper presents a connectionist model that builds on the split-fovea model (Shillcock and Monaghan, 2003). The split-fovea model integrates into its architecture the facts that the brain and the fovea are both split. To simulate the split fovea, the model has 2 input layers. This allows for words to be presented to the network at different fixation positions. The paper compares two networks: 1. A control net for which words are presented with the same frequency at all fixation positions during training. 2. A fixation net for which the frequencies for words at different fixation positions are determined by actual data of people while reading. The aim of the paper is to establish any differences occuring in the fixation net due to the different fixation positions.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Gil Orazi*)

Table of Contents

Chapter 1

Introduction and Background

The first section of this chapter will provide a brief introduction and overview of the current paper, while the second section will give the background needed for the current work.

1.1 Introduction

Connectionist networks have been used extensively in the domain of visual word recognition and reading. There have been a number of connectionist networks in recent years which model the reading behaviour of people. An interesting model is the split-fovea model (Shillcock and Monaghan, 2003). This model integrates the facts that the brain and the fovea are both split into its architecture by having two separate inputs for the right and left visual fields and two separate hidden layers for the two hemispheres. Doing this allows one to present words to the network at different fixation positions. Because of this possibility the network is well suited for experimenting with words being presented at different fixation positions. The current work involves comparing two networks. The first one is a control network similar to the network used by Shillcock and Monaghan (2003). The second network (the fixation net) on the other hand involves using real fixation data obtained by people reading newspaper articles. The frequency with which words are fixated at each position is entered into the second network. The aim of the paper is to establish any differences occurring in the fixation net due to the different fixation positions.

The next section will give a brief overview of both the Dual-Route model of reading and some connectionist models. In particular it will concentrate on the Seidenberg and McClelland (1989) and Harm and Seidenberg (1999) models since the former was a very influential model in the domain and the latter is an improvement of it. This will be followed by a brief description of the split-fovea model (Shillcock and Monaghan, 2003). Another important part of this paper is the reading behaviour of people, namely at which position they fixate words while reading. This is described in the last section of this chapter.

The next chapter will then present the architecture of the networks used for the current work. Additionally it will explain the training regimes used before going on to give the results of the networks after training. Finally, in chapter 3, there will be a brief discussion of the the results and a presentation of possible future work.

1.2 Background

1.2.1 Models

A lot of research has been done on visual word recognition (for an overview see for example Seidenberg (1995)). There are two main types of models of visual word recognition and reading. The more traditional Dual Route model (Coltheart et al., 1993; Weekes and Coltheart, 1996) and the Connectionist models (Seidenberg and McClelland, 1989; Harm and Seidenberg, 1999; Plaut, 1999; Shillcock and Monaghan, 2003). Both of these types of models are concerned with the processes by which the visual

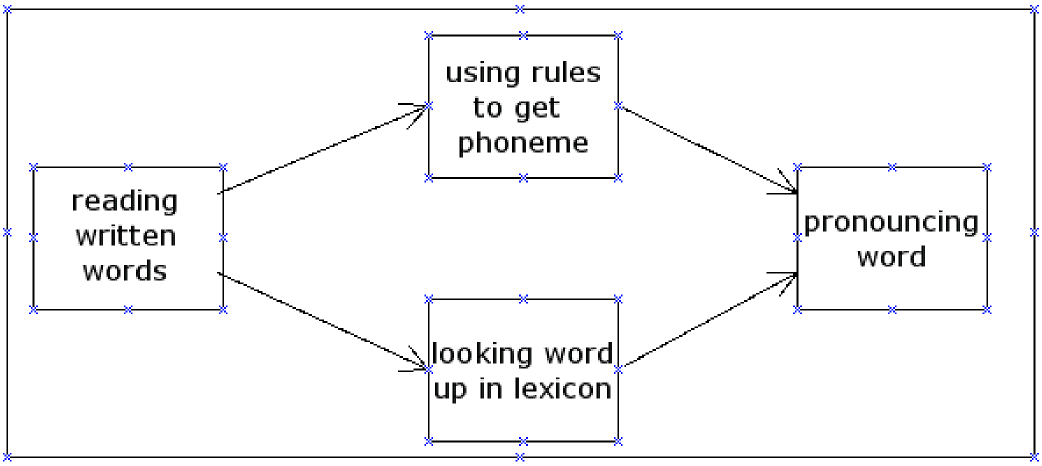

Figure 1.1: Basic Dual Route Model: Words can either be read using the lexicon or by the rules mapping from orthography to phonology

form of words (or graphemes) is translated into the corresponding phonological form (or phonemes). However they differ considerably in their approach to the problem. First, in the next section, the Dual Route model will be briefly reviewed in order to set the connectionist models into context. After that, the next section will present a review of current connectionist models and their applications.

Dual-Route Model

The Dual-Route model is the more traditional model of reading(Coltheart et al., 1993; Weekes and Coltheart, 1996). The model relies on the assumption that words are stored in a *lexicon*. According to the model, there are two such lexicons, an orthographic one which stores the orthographic forms of all the words known and a phonological one, which stores the phonological forms. The name of this model comes from the assumption that there are two distinct routes by which words can be recognised and read out aloud. The first of these routes, called the *lexical route* makes use of the lexicons to read words. If a word is read, its orthographic form is looked up in the orthographic lexicon and if a matching word is found, this is used to look up how to pronounce the word in the corresponding phonological lexicon. The second route on the other hand, uses a set of *rules* to map from the orthographic to the phonological form, without having to refer to the lexicon at all (see Fig. 1.1). The rules are used to convert the letters of the written word to the associated phonetic sounds.

From this, it is clear that exception words can only be read using the lexical route, since their spelling-to-sound correspondence is highly irregular. Since exception words do not obey any rules for mapping orthography to phonology, they must be looked up in the lexicon. If the non-lexical route would be used for these words, the result would be *regularisation errors*, where the word would be read according to the rules for regular words. On the other hand, new words that have never been seen before, and non-words, can only be read using the non-lexical route, since they are not present in the lexicon. These words are read according to the rules for mapping regular words from orthography to phonology. Defendants of the Dual-Route theory often maintain that it is not possible for any single route model to correctly read both words and non-words (or novel words).

One of the tests of this model that is often cited are the two main forms of dyslexia. In surface dyslexia, patients have problems reading exception words, whereas performance on non-words is normal (see for example Coltheart et al. (1983)). On the other hand, phonological dyslexia is characterised by difficulties in non-word reading but normal exception word reading (see for example Howard and Best (1996)). In the Dual Route model, surface dyslexia can be explained as being caused by an impairment of the lexical route, whereas phonological dyslexia is caused by an impairment of the rule-based route. A number of studies have been done with dyslexic patients to test the dual-route model (Castles and Coltheart, 1993; Coltheart et al., 1993; Weekes and Coltheart, 1996).

Connectionist Models

Connectionist models are an alternative to the Dual-Route model for visual word recognition and reading. Connectionist networks are computational models that try to be more brain-like than the more traditional box-and-arrow models (e.g. the Dual-Route model). The most basic components of these networks are abstract models of neurons called *units*. The connections between neurons are modelled by *weighted connections* between the units. The units are usually grouped into different layers. A typical feed-forward network generally has an input layer, a hidden layer and an output layer. Each unit in the input layer is connected to each unit in the hidden layer, which in turn is connected to each unit in the output layer (see Fig. 1.2). Activation from the input units spreads to the hidden and then to the output units. For an overview of connectionist modelling see for example O'Reilly and Munakata (2000).

A lot of research has been done in using connectionist networks for visual word recognition (for an overview see Christiansen and Chater (1999). The networks for reading are generally structured as that in figure 1.2 with an input, an output and a hidden layer. It is important to note that unlike the Dual-Route model of reading, the connectionist models have to use the same mechanism to read novel words and exception words.The models show that it is *possible* to have a reading system that uses the same system for *all* words, which is contrary to the claim made by Coltheart, who says that there *have* to be two routes. One of the big challenges for early connectionist models was to prove that the networks could replicate the behaviours of surface and phonological dyslexia if they were damaged in different ways. One of the first successfull connectionist models of reading was the one by Seidenberg and McClelland (1989), which was subsequently improved (Plaut et al., 1996; Harm and Seidenberg, 1999). These connectionist networks were not only able to correctly pronounce normal, exception and even non-words, but they also managed to simulate dyslexic behaviour. The fact that they manage to replicate surface and phonological dyslexia is an important part in

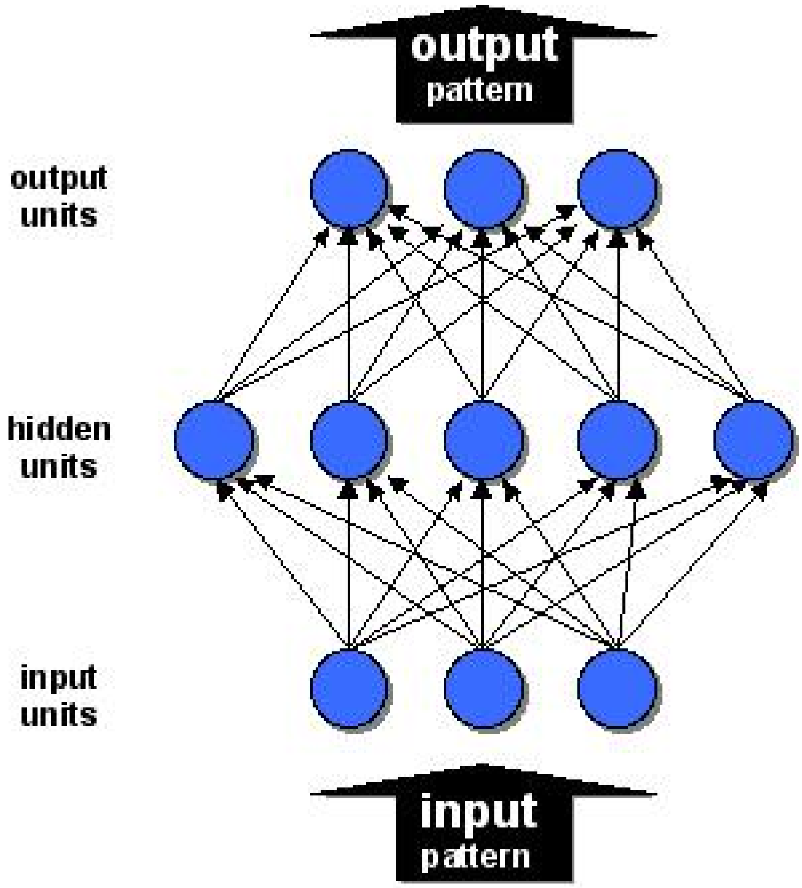

Figure 1.2: Basic Feed-Forward Connectionist Network: Wach unit in the input layer is connected to each unit in the hidden layer, which in turn is connected to each unit in the output layer. The network has to learn to map from given input patterns to the correct output patterns.

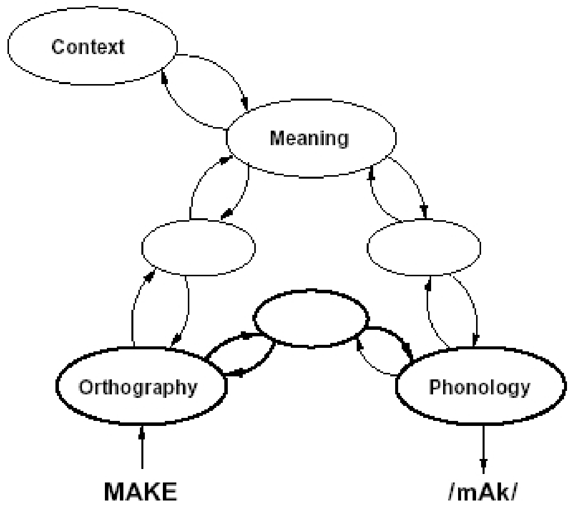

Figure 1.3: The Seidenberg and McClelland (1989) Model: Triangle model with layers representing orthography, phonology, meaning and context. Activation spreads between layers via the hidden layer between them

the success of connectionist models, since every serious reading model should be able to simulate the two different forms of dyslexia.

The Seidenberg and McClelland (1989) Model The Seidenberg and McClelland (1989) model was one of the first successfull connectionist networks used to model reading processes. The whole model (also called 'triangle model') consists of four different layers (excluding the hidden layers) as shown in figure 1.3:

- Orthography: The orthographic form the words.
- Phonology: The phonological form of the words.
- Meaning: The semantics of the words.
- Context: The context in which the word appears.

Only the bottom part of the whole model, i.e. the mapping between orthography and phonology was implemented computationally. The model was implemented as a feed-forward connectionist networks, which did not include feedback from the phonological to the hidden units as in figure 1.3. The assumption here was that a feed-forward architecture was enough for a net that only has to learn how to pronounce words and which does not have to worry about the semantics and the context in which the words appear.

For the orthographic as well as the phonological representations in the network, Seidenberg and McClelland (1989) chose a Wickelfeature scheme (Wickelgren, 1969). In this scheme the representation is of a word is made in terms of *triples*. For example the orthographic form of the word PINT would be represented by _PI, PIN, INT, NT_. This representation was necessary in order to avoid the problem that words like TUB and BUT would have the same representation since the same letters are activated for both of them. Note that these orthographic and phonological representations of the words are distributed activation patterns of the orthographic units and as such are fundamentally different from the orthographic lexicon as employed in Coltheart's Dual-Route model. There is no such lexicon in this model but rather the orthographic representation of a word is *distributed* over several units. When a word is presented to the network, activation spreads to the hidden layer and from there to the phonological layer. For example the activation of a hidden unit is determined by the activation and weights of all the input units connected to it.

The network was trained on 2897 monosyllabic English words for 250 runs, with words being presented at random to the network at each run. The words were randomly selected according to the frequency with which they appear in the English language. However, rather than using the actual frequencies, logarithmic frequencies were used to reduce the time of training[1]. Learning was done by the standard *error back-propagation* algorithm (Rumelhart et al., 1986; O'Reilly and Munakata, 2000). This algorithm compares the computed output to the the correct target output and uses the distance between the two as an error score.

[1] For further details on logarithmic frequency see chapter 2

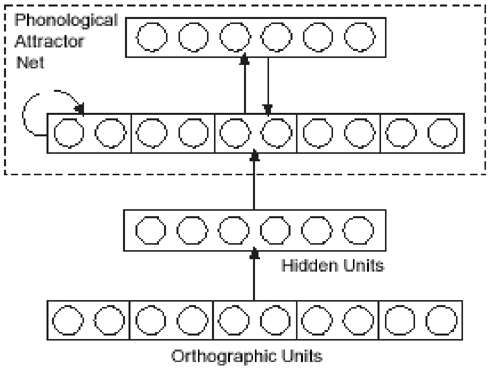

Figure 1.4: This figure (taken from Harm and Seidenberg (1999)) shows the basic architecture of the model.

After training, the network computed the correct output for 97,3% of the words in the training corpus. In total there were 77 errors, 14 of which arose from wrong coding by the experimenter. Errors made were mostly on low-frequency words, and 14 of the errors were regularisation errors, where an irregular word is pronounced in a regular way. However, an important criticism of this model was that it performed much less well on non-words than human subjects (Coltheart et al., 1993). For example on one set of nonwords, the networks performance was 59% correct whereas humans typically get 94% correct. The problem is that Seidenberg and McClelland (1989) claimed that it was meant to read regular, exception words and non-words at a level comparable to human subjects.

The Harm and Seidenberg (1999) Model Harm and Seidenberg (1999)'s model builds directly on the previous model. There are several improvements to the previous model, with the main difference being the phonological output component (see figure 1.4).

As the previous net, this one has a phonological output layer. The difference now is that the whole phonological output component is made up of the phonological units and a layer of so-called *clean-up* units. These clean-up units can be thought of as a second set of hidden units with connections *from* and *to* the phonological units. Networks using this architecture are called *attractor* networks. The advantage is that the clean-up units help the phonological output layer to settle into the correct pattern. Attractor networks have an important property. They can form so-called *attractor basins*. This means that if the network is in state that it has not encountered before but that is close to a known legal state, then the current state will be attracted to the legal state and the network will eventually settle in that legal state. Figure 1.5 is an example of this. Note that the figure is simplified, using only 2 phonetic features. An attractor basin in this net has in reality 11 dimensions since it has 11 phonetic features. This enables the network for example to repair noisy representations since a particular noisy output will always be attracted to a valid state.

Another key difference between the two models it that Harm and Seidenberg (1999) used a phonological representation that is much closer to actual phonetic features. A single phoneme was represented by 11 units, each of which corresponded to a specific phonetic feature (e.g. voice, nasal, round...) and whose activation could vary between -1 and 1 (see figure 2.2 in chapter 2).

This model was trained on 7839 monosyllabic words. Performance on words included in the training set was high, scoring 99% of words correctly. For nonwords, the network's performance was significantly better than the older network's, scoring 84% of words correctly. In this respect, the model achieved to eliminate one of the major criticisms of the older model, namely the low scores on nonword reading.

It has to be noted that the non-implemented parts of the complete triangle model as shown in figure 1.3 nontheless constitute an important part of the reading process. For example Plaut et al. (1996) suggest that there are two different ways in figure 1.3 to pronounce words. The first one is the direct route from orthography to phonology used by the Seidenberg and McClelland (1989) model and the second one is the indirect

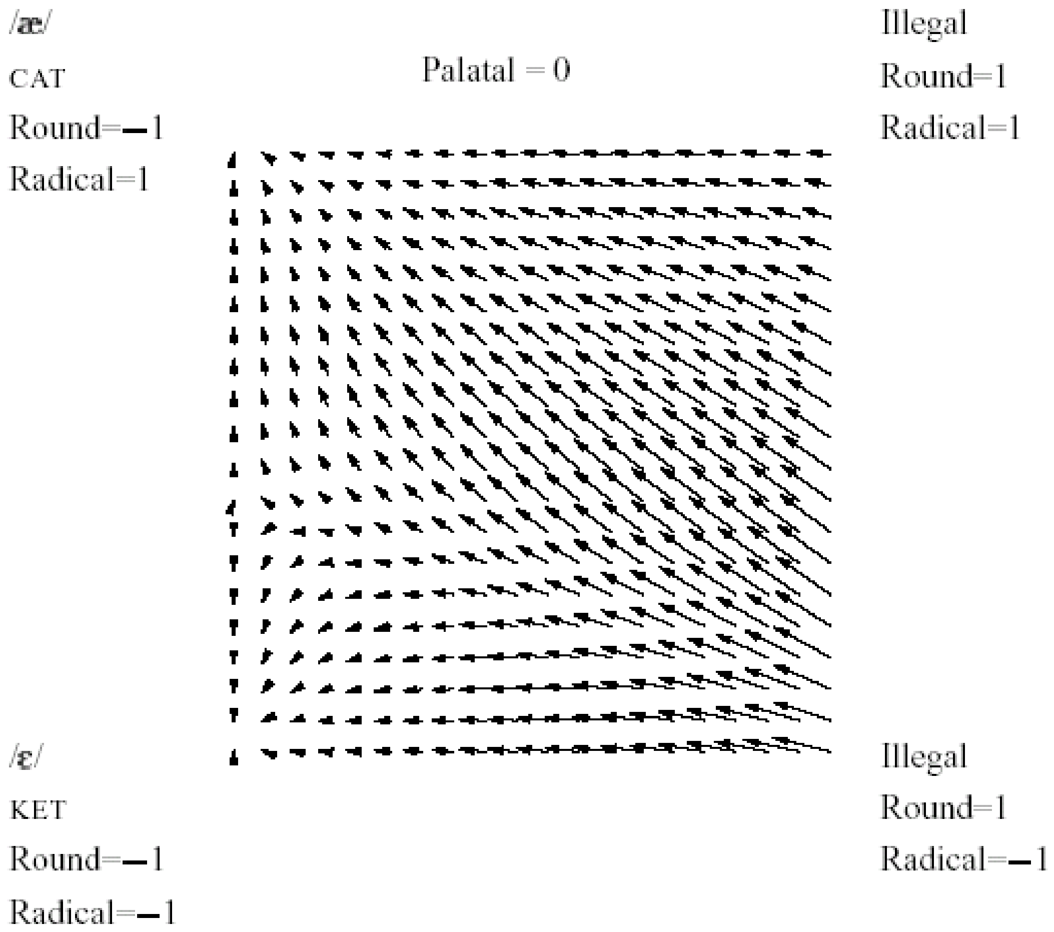

Figure 1.5: This figure (taken from Harm and Seidenberg (1999)) shows an attractor basin.

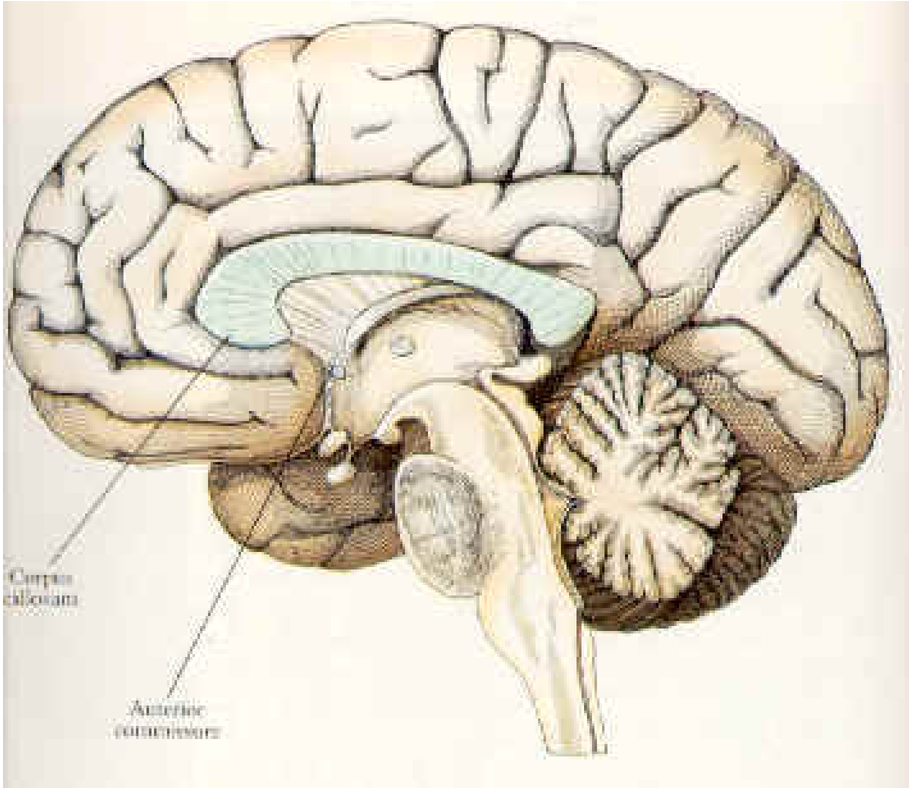

Figure 1.6: Side-view of the Corpus Callosum

route from orthography via meaning to phonology. Note that this is different from the Dual-Route model discussed before. In the present case, activation always spreads across both routes, whereas in the Dual-Route model, words are read using either one or the other route. With two routes providing the input to the final phonological output, Plaut et al. (1996) assumed that exception words would rely more heavily on the route via meaning whereas the direct route was dominant for regular words.

1.2.2 The Split-Fovea Model

The Corpus Callosum and the Split Fovea. It is a well-known fact that the brain is divided into two hemispheres which are joined together by the Corpus Callosum (see figure 1.6 for a side-view and 1.7 for a top-view). For an overview of the Corpus Callosum see for example Gazzaniga (2000). An often ignored fact but equally

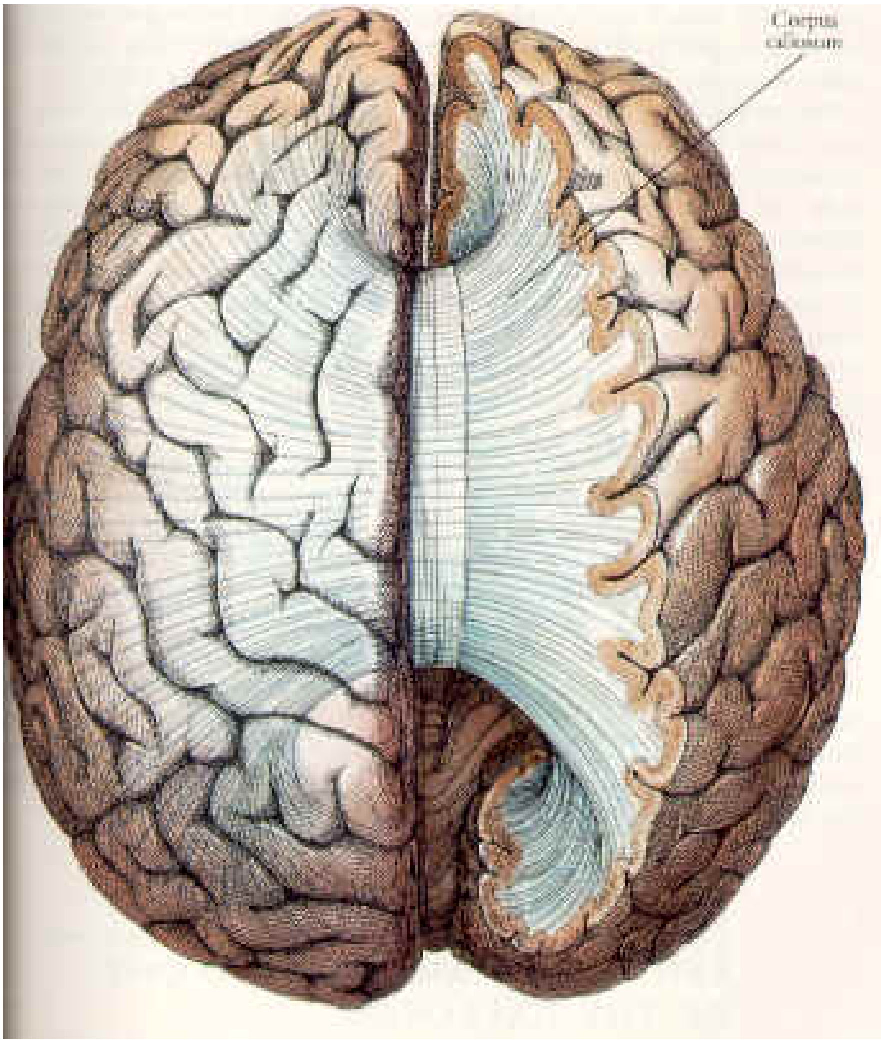

Figure 1.7: Top-view of the Corpus Callosum. It can clearly be seen that the Corpus Callosum connects the two halves of the brain.

important in the area of visual word recognition is that the fovea in the eye is also precisely vertically split (Leff, in press). Thus a fixated picture is vertically divided, with the left part being sent to the right hemisphere (and the right half of the fovea) and vice-versa. In the case of reading and visual word recognition, this means that the word is divided into the two hemispheres. Everythin left of where the word is fixated is sent to the right hemisphere and everything right of the fixation point is sent to the left hemisphere. In order to be able to recognise the word and read it out aloud, it is necessary for the information in the two halves of the brain to be recombined. This is done via the Corpus Callosum. Only after the information about the word in the two hemispheres is recombined can the word be recognised.

Split-Fovea Model Shillcock and Monaghan (2003) use the fact that both the fovea and the brain are divided and integrate it into a connectionist model which they call the split-fovea model. The model is similar to the Harm and Seidenberg (1999) model with some crucial differences. Most notably, the input as well as the hidden layer is split into two, reflecting the division in the fovea and in the brain respectively. The input representing the right visual field is connected to the hidden layer representing the left hemisphere and vice-versa. Both hidden layers are connected to each other. These connections represent the communication between the two halves of the brain via the Corpus Callosum (see figure 1.8). It has to be noted however that this is only a very abstract model and cannot be seen as an accurate model of the Corpus Callosum. One important difference for example is that in the model each unit in the left hidden layer is connected to all units in the right hidden layer and vice-versa. This is contrary to what happens in the Corpus Callosum, which constitutes a connection between the same areas in both hemispheres. However, this connection between the hidden layer is a crucial part of the model since it allows the model to compute a single correct output pattern from two input patterns as shown in figure 1.8.

Because of the two input layers, the model allows for words to be presented to the

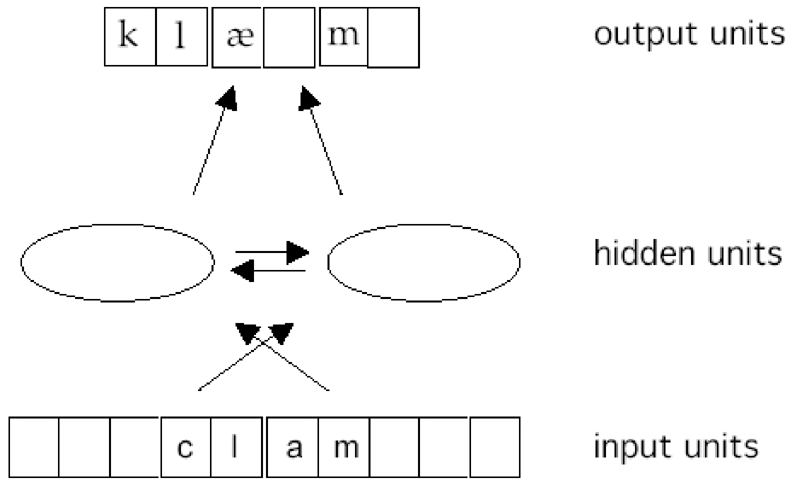

Figure 1.8: The split-fovea-model. Here the word clam is presented with a fixation point in-between the letters 'l' and 'a'. Activation spreads into the hidden layers and from there to the output layer which produces the correct output /klaem/.

network in different fixation points. For example Figure 1.9 shows the different inputs for the word clam. The network has to learn to map all the different possible inputs for a word to the same correct output target. To be able to do this task correctly for all the words and non-words presented to it, the network has to learn a *shift-invariant* mapping from the input to the output, so that shifting the input across the viewing positions doesn't make a difference to the final computed output.

1.2.3 The Viewing Position Effect

The fact that words can be fixated at many different positions (once or more than once) during reading is an important aspect of visual word recognition. There is evidence that the probability of recognising a word depends on where it is fixated (O'Reagan and Javobs, 1992; Montant et al., 1998; Nazir, 2000). Words were presented to people while controlling the exact position at which the word was fixated by the person. This is done for words with different lengths and the words that were recognised correctly

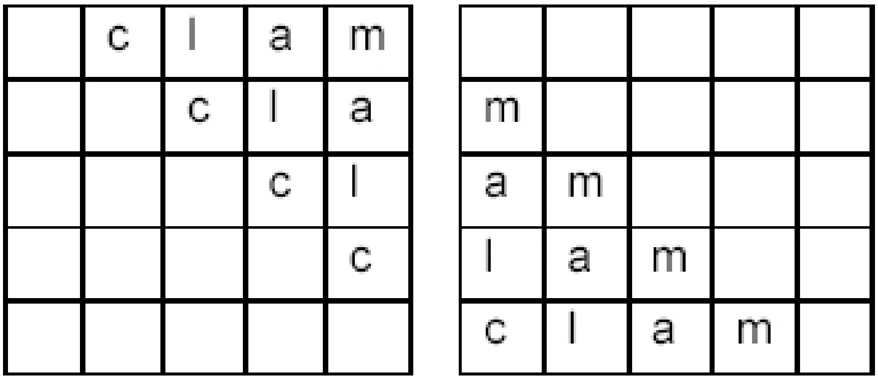

Figure 1.9: Possible fixation positions for the word clam in the split fovea model.

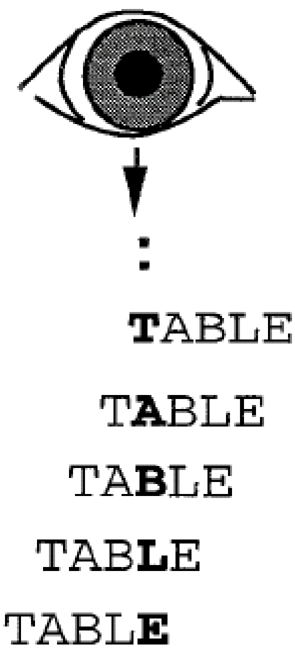

Figure 1.10: Different fixation positions for the word table.

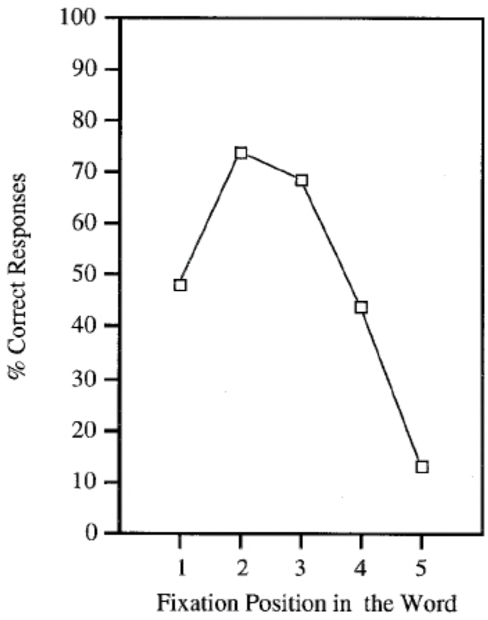

Figure 1.11: Probability vs. fixation position for seven letter words.

at the different fixation positions are recorded. All words of different length were divided into five zones into which a fixation could fall (see for example figure 1.10). By comparing the probability that a word was recognised to the location at which it was fixated, it can be seen that there is a preferred viewing position slightly left of the centre of the word (figure 1.11. This effect is fairly small for short words (i.e. four letter words), but increases with the length of the word. Figure 1.11 shows a curve of the probability of correct pronunciation against the fixation position for seven letter words. It can clearly be seen that words are more easily recognised if they are fixated left of centre and the probability drops sharply on both sides of this optimal viewing position. Note that for a four letter word, their is only a very slight difference in the recognition probability. This might at first look counterintuitive because if the word were presented exclusively to one hemisphere, it could be processed immediately since all the information would be present. But according to Shillcock et al. (2000) the division of labour between the two hemispheres due to the Corpus Callosum is beneficial for the task of visual word recognition rather than hindering it. They show

Fixation Position	Example	Frequency
1	-WORD	0.177
2	W-ORD	0.196
3	WO-RD	0.206
4	WOR-D	0.221
5	WORD-	0.200

Table 1.1: Frequencies with which four-letter words are fixated at different fixation positions during reading. In the example, '-' indicates the fixation position. The data is taken from McDonald and Shillcock (submitted).

that the optimal viewing position slightly left of centre allows each hemisphere to receive the same amount of information about the word.

The finding that there is a preferred viewing position is interesting for the split-fovea connectionist model. What will happen to the model if this fact is incorporated into the network and will this network behave differently from a normal control network? The network presented in the next chapter integrates the findings about the fixation positions during reading with the split-fovea connectionist network. Table 1.1 shows the data used for frequency of fixations at different positions for four-letter words. The frequency data is taken from McDonald and Shillcock (submitted) and the frequency of the fixations was evaluated with data from the EMBRA corpus which has eye movement data from people reading newspaper articles. Note that in table 1.1, the words are fixated slightly right of centre, which is contrary to the claim made before about the preferred viewing location. This is due to the fact that the data in this table is collected during reading and not with single word presentations. However, this is only the case for short words. For longer words the highest frequency of fixations will again be slightly left of centre (McDonald and Shillcock, submitted). One explanation for this behaviour is that while reading, the next word (the one after the currently fixated word) will already be in parafoveal vision. Because of this, the first few letters of the next word will already been known and the next saccade will be made so as to fixate the word slightly right of where it would have been fixated otherwise. For small letter words, like four letter words, this means that they will be fixated slightly right of centre.

This chapter provided an overview over connectionist models of visual word recognition as well and the viewing position effect. The next chapter contains details about the networks used in the current work and the results of the current simulation.

Chapter 2

The Simulations

This chapter starts off with an overview of the networks used in this simulation. It will then go on to describe the training regime and will finally present the results of the simulations.

2.1 The networks

Two networks are used in the current simulation. The first one is a control network, similar to Shillcock and Monaghan (2003) where the words have the same frequency at each fixation position. The second one, the 'fixation' network, has different frequencies for the different fixation positions as in table 1.1 from previous chapter. The control network is essential in order to be able to compare the performance of the fixation network and to see if there is any difference in the behaviour of the two networks.

2.1.1 Network Structure

The networks both have two input layers, one corresponding to the right visual field and the other one to the left visual field. Both of these are connected to separate

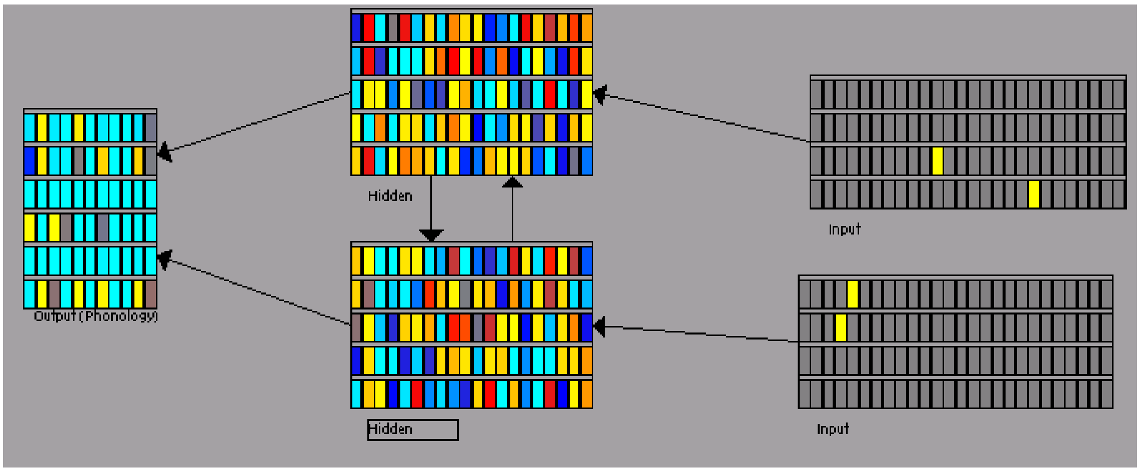

Figure 2.1: A graphical overview of the PDP++ network used. The two input layers are connected to the two hidden layers which are connected to the output. The two hidden layers are connected to each other. This example is for the word desk fixated between the letters 'e' and 's'. A unit with an activation of 1 is yellow and a unit with an activation of -1 is blue. Any other colors are somewhere in-between this range.

hidden layers corresponding to the left and right hemisphere respectively. The two hidden layers are connected to each other and both are connected to the output layer (see figure 1.8 in chapter 2).

Like in Shillcock and Monaghan (2003), both the orthographic input and the phonological output representations ared slot-based representations. For the orthographic input, this means that there are four slots per input layer, one for each letter (since we are dealing with four letter words). Each slot has twenty-six units, so that each unit in a slot corresponds to a single letter. Thus, the two input layers each have 104 units. When a word is presented to the network, the unit corresponding to the letter in each slot is activated. For those slots where no letter is presented, no unit is active. Both hidden layers have 100 units, which is enough for them to solve the task (Shillcock and Monaghan, 2003). The network was implemented in PDP++ (Dawson et al., 2001) and figure 2.1 shows the graphical 'display' of the network. The figure shows a presentation of the word 'desk' after training was completed. The network is able to 'read' the word correctly and produces the correct output. Yellow units stand for an activation of +1, whereas blue units stand for an activation of -1. Any other colors are somewhere in-between[1].

Phonological Representation. The phonological output layer consists of 66 units in total. The layer is made up of six slots containing eleven units each. These eleven units correspond to the eleven different phonemic features which were used by Shillcock and Monaghan (2003) (see figure 2.2). As can be seen from the figure, each phoneme has a very specific set of features associated with it. In the output layer, these features are represented by the activation values of the different units, ranging between -1 and 1. Some more special English phonemes (for example 'x' as in the Scottish 'loch') are not represented in this table and words containing these were discarded. The phonological representation was of the onset-nucleus-coda (CCVVCC) form (e.g. Plaut et al. (1996)). The first two slots are reserved for the onset consonants, the middle two for the nucleus vowels and the last two for the coda consonants. For words that have less than 6 features (e.g. back → b&k) the remaining empty slots have all their features set to -1. The network's task therefore is to map the slot-based orthographic input into the just described phonological representations at the output.

Training The training corpus for both networks consisted of English four letter words taken from the CELEX corpus (Baayen et al., 1995). The words were taken from the CELEX word form corpus rather than the word lemma corpus since the word forms are what is actually read and the fixation position data used thus applies to the word forms rather than the word lemmas. After eliminating the words that did not fit the ONSET-NUCLEUS-CODA structure, there were 1988 words left. Since there are five different positions in which the words can be presented to the net, the total number of events presented to the network is 9940.

[1]Note there are no clean-up units in this architecture. This network does not need clean-up units to learn its task

Phoneme	Sonorant	Consonantal	Voice	Nasal	Degree	Labial	Palatal	Pharngeal	Round	Tongue	Radical	Example
/p/	-1	1	-1	-1	1	1	0	-1	1	0	0	pink
/b/	-1	1	0	-1	1	1	0	-1	1	0	0	bike
/t/	-1	1	-1	-1	1	-1	1	-1	-1	1	0	tone
/d/	-1	1	0	-1	1	-1	1	-1	-1	1	0	dust
/k/	-1	1	-1	-1	1	-1	-1	-1	-1	-1	0	kid
/g/	-1	1	0	-1	1	-1	-1	-1	-1	-1	0	gale
/f/	-0.5	1	-1	-1	0	-1	1	-1	1	0	0	far
/v/	-0.5	1	0	-1	0	-1	1	-1	1	0	0	vote
/th/	-0.5	1	-1	-1	0	-1	1	-1	-1	0	0	think
/dh/	-0.5	1	0	-1	0	-1	1	-1	-1	0	0	this
/s/	-0.5	1	-1	-1	0	-1	1	-1	-1	1	0	sip
/z/	-0.5	1	0	-1	0	-1	1	-1	-1	1	0	zone
/h/	-0.5	1	0	-1	0	-1	-1	1	-1	-1	-1	hall
/sh/	-0.5	1	-1	-1	0	-1	0	-1	-1	0	0	ship
/zh/	-0.5	1	0	-1	0	-1	0	-1	-1	0	0	beige
/tch/	-0.8	1	-1	-1	1	-1	0	-1	-1	0	0	chink
/dzh/	-0.8	1	0	-1	1	-1	0	-1	-1	0	0	jug
/m/	0	0	1	1	1	1	0	-1	1	0	0	mop
/n/	0	0	1	1	1	-1	1	-1	-1	1	0	nit
/ng/	0	0	1	1	1	-1	-1	-1	-1	-1	0	ring
/r/	0.5	0	1	0	-1	-1	-1	1	1	-1	-1	rope
/l/	0.5	0	1	0	-1	-1	1	-1	-1	1	0	lair
/w/	0.8	0	1	0	0	1	-1	-1	1	-1	0	wave
/j/	0.8	0	1	0	0	-1	0	-1	-1	0	1	yolk
/i/	1	-1	1	0	0	-1	0	-1	-1	0	1	seat
/I/	1	-1	1	0	0	-1	0	-1	-1	0	-1	pit
/e/	1	-1	1	0	-1	-1	0	-1	-1	-1	-1	pen
/ae/	1	-1	1	0	-1	-1	0	1	-1	-1	1	fat
/a/	1	-1	1	0	-1	-1	-1	1	-1	-1	-1	hot
/uu/	1	-1	1	0	0	-1	-1	-1	1	0	-1	look
/u/	1	-1	1	0	-1	-1	-1	-1	-1	-1	-1	tug
/@/	1	-1	1	0	-1	-1	0	1	-1	-1	1	bird

Figure 2.2: Phonological features used by Harm and Seidenberg (1999)

For the control net, the frequency with which the words appear in text was also taken from the CELEX database. However, using the actual frequencies as found in the CELEX database would require too long a training time until every word present in the training corpus has been presented to the net at least once. This is because there is a very large difference between very high and very low frequency words (e.g. 'that' appears 217376 times and 'thaw' only 45 times in the CELEX corpus). For this reason, the frequencies were compressed into log-frequencies according to the formula:

$$p_i = \frac{log((f_i/100)+1)}{log(m/100)} \tag{2.1}$$

where p_i is the log frequency of word i, f_i is the CELEX frequency of word i and m if the frequency of the most frequent word in the training corpus (see Harm and Seidenberg (1999) for the formula and Plaut et al. (1996) for a discussion of the log frequency). Essentially this expresses the frequencies in terms of the most frequent word. The most frequent word in the corpus gets a frequency of one, all the other words have frequencies between zero and one. Words with a final log-frequency less than 0.05 were given a frequency of 0.05 in order that they appear in the training corpus enough times for the network to learn. Thus the frequency of the most frequent word is 20 times the frequency of the least frequent words. It also means that in the training regime, the most frequent word will appear on each epoch and the least frequent words should appear once every 20 epochs. Note that in the control network, each one of the five different possible positions has the same frequency. Note also that during training the network selects events at random (according to the frequency) from the 9940 possible events, so that it is not the case that a word is presented in all its possible fixation points one after the other.

For the second net, each one of the possible fixation positions has a different frequency according to table 1.1. The original frequencies from the CELEX database were changed according to the table and then the log-frequency was taken. Thus every fixation position has a slightly different frequency.

Training was done using the recurrent back-propagation algorithm supplied by the PDP++ software. The software allows the user to specify the exact timesteps that the simulation goes through when an event is presented. The total number of timesteps per presented event was set to seven. At timestep one, only the input is presented to the network. Activation is then allowed to spread through the network until at timestep four, the correct output target is also presented to the network. This is used for error-backpropagation. The network essentially compares its output to the target output and changes the weights in such a way as to reduce the error (see e.g. Dawson et al. (2001)). Training was ended after 2750 epochs, corresponding to about 4 million words presented to the network. An important thing to note is that both networks started with the exactly same initial small random weights. This is important in so far as that it can be excluded that any observed effects can be explained by different random weights of the two networks at the start of the training process. The only differences between the two training regimes are that different events are presented at random for each epoch according to their frequency.

Testing After training finished, the networks were tested to see what they have learned and which words they were unable to learn. Testing was done in the following way. PDP++ produces a text file containing the activation of the output units that it produced for any given event. That is, it produces a string of numbers between -1 and 1. To be able to analise the networks behaviour, this string of numbers has to be converted into the phonetic form of the word. To do this, each individual feature was converted into the corresponding phonetic feature by calculating to which of the possible 11 features in figure 2.2 it had the shortest distance. After doing this for each of the 6 different slots, the phoneme that the network calculated emerged.

	Control Network	Fixation Network
Total Events	9800	9800
Correct	9661	9664
Errors	139	136
Percentage Correct	98.58	98.61
Total Words	1960	1960
Correct	1894	1886
Errors	66	74
Percentage Correct	96.63	96.22

Table 2.1: Overall results of the two networks after 2750 epochs (or about 4 million words) of training. The upper table is for all the events so that each word is represented five times: once per fixation position. The lower table is only for individual words.

2.2 Results

This section presents the results of the networks after training for 2750 epochs. It will start off by discussing the overall results about of the networks' performance and then go on to list more detailed results in order to compare the two networks.

2.2.1 General

In this section some general results about both networks will be discussed, showing that there is virtually no difference in the general overall performance of both networks. A more detailed analysis will follow in subsequent sections. Note that all of the following results only include non-homographic words. An analysis of the 14 homographs (70 events) in the training corpus follows in a separate section. The homographs were left out from the current (and subsequent) analysis because the networks will inevitable get one of the possible pronunciations wrong. This is because there is nothing in the model that could differentiate between two different contexts or meanings and so the network is unable to learn which pronunciation to apply.

The overall results for both networks can be seen in table 2.1. The table shows the networks' performance after 2750 epochs ($\sim$4 million words). The upper part of the table shows the statistics relating to the number of *events* that the network was trained on. In overall performance, both networks are very similar, with the control network getting 98.58 percent of the events in the training corpus correct and the fixation network getting 98.61 percent correct. This very slight difference of 0.03 percent (or three words) between the networks is so small that it is negligible and one can say that both networks performed the task equally well. The fact that the control network only made three errors more than the fixation network can not be considered as evidence that the fixation network performed better than the control. This first result is not surprising as both networks were trained on exactly the same training corpus and with exactly the same initial conditions, with only a very small difference in fixation position frequencies.

The bottom part of table 2.1 shows how many words the networks got correct in *every* position. The statistics in the upper part of the table relates to the individual events and so includes each word five times, once in each of the different possible fixation positions. The bottom part on the other hand only considers a word to be correct if the network computed the right output for each of the fixation positions. Hence, for example the control network only made a mistake on the word 'lewd' when it was fixated at the second position (network: 'lUUd' - Correct: ljUUd). So 'lewd' is counted as an error in the bottom part of the table since it has not been learned correctly at all of the fixation positions. Again, it can be seen from the table that both networks are closely matched in their overall performance. However there is a small difference in that the fixation network gets 8 words more wrong here than the control.

To test if the network was able to learn both regular and exception words, it was tested *a. Regular*

	Control Network	Fixation Network
Total Events	460	460
Correct	460	458
Errors	0	2
Percentage Correct	100	99.6

b. Irregular

	Control Network	Fixation Network
Total Words	305	305
Correct	304	301
Errors	1	4
Percentage Correct	99.7	98.7

c. Ambiguous

	Control Network	Fixation Network
Total Words	95	95
Correct	95	95
Errors	0	0
Percentage Correct	100	100

Table 2.2: Overview of both networks' performance on the regular, irregular and ambiguous word list (see appendix A).

a. Regular

Word	Frequency	Fixation Position	Computed Output	Correct Output
dots	0.074	5	/dVts/	/dOts/
toad	0.062	2	/s@Ud/	/t@Ud/

b. Irregular

Word	Frequency	Fixation Position	Computed Output	Correct Output
chic	0.080	5	/Jiik/	/Siik/
chic	0,080	1	/Tiik/	/Siik/
hind	0.113	4	/h&Vnd/	/h&Ind/
mule	0.083	3	/mjUIl/	/mjUUl/

Table 2.3: Errors on the regular and irregular words for the fixation net.

on a list of regular, ambiguous and exception words taken from Plaut et al. (1996). Note that for this test only words that appeared in the training corpus were taken. The lists of the words can be found in appendix A. The list of regular words contains 92 elements (corresponding to 460 events) , the list of irregular words contains 61 elements (305 events) and the list of ambiguous words contains 19 elements (95 events). Table 2.2 shows the results of these tests. The control network only made one error on the irregular word list, getting all the regular and ambiguous words correct. The error was made on the 0.08 frequency word chic, pronouncing it as /Jiik/ instead of /Siik/ at fixation position 5. The fixation net made 2 errors on the regular words, 4 on the irregular words and none on the ambiguous words. The details of the errors can be found in table 2.3. All but one of the errors are made on low frequency words with a frequency less than 0.1. These tests show that both networks have learned to pronounce both regular and exception words.

	Control Network	Fixation Network
Total errors	139	136
Errors with frequency ≤ 0.1	129	116
Errors with higher frequency	10	20
Average Frequency	0.056	0.066

Table 2.4: Errors by frequency.

2.2.2 Frequency

Table 2.4 shows the network's performance in relation to the frequency of the incorrect words. Not surprisingly, for both networks, most errors were made on low frequency words. The control network made 129 of its 139 errors on words with a frequency lower than 0.1 and he fixation network 116 of 136. The average frequency of the errors was 0.056 for the control and 0.066 for the fixation model. The error with the highest frequencies were 'coup' at all positions for the control with a frequency of 0.110 and 'shed' at position 1 for the fixation net with a frequency of 0.223 (see table 2.5 and 2.6. This result also means that the networks have correctly learned *all* the higher frequency irregular words that were in the training corpus.

Tables of the errors with a frequency ≥ 0.1 have been included (2.5 for the control network and 2.6 for the fixation network). Note that a complete list of the errors made by each network can be found in appendix B and C respectively. As can be seen from the tables, most pronunciations computed by the networks, even though they are wrong, are nevertheless pretty close to the actual output. Thus, for example both networks pronounce the word 'dose' as /d@Uz/ instead of /d@Us/ in most positions. Clearly this error is only a very slight error in the sharpness with which the coda of 'dose is pronounced. In fact, looking through the errors in the appendix, this error of the network substituting an /z/ instead of an /s/ is very common. Although this seems not to be a very big error, the pronunciation /d@Uz/ would correspond to the word 'doze' (which also appears in the training corpus) and hence will have a completely different meaning than the original input word 'dose'. Thus the network does not make a difference between the words 'dose' and 'doze'.

Word	Frequency	Fixation Position	Computed Output	Correct Output
coup	0.110	all	kUUk	kUU
dose	0.109	1,3,4	d@Uz	d@Us
dose	0.109	2	dEUz	d@Us
khan	0.107	1	JVrn	kVrn

Table 2.5: Errors with frequency $\geq$ 0.1 for the control network.

Word	Frequency	Fixation Position	Computed Output	Correct Output
coup	0.110	all	kUUJ	kUU
dose	0.109	1,2,3	d@Uz	d@Us
heir	0.110	all	I@r	E@r
hind	0.113	4	h&Vnd	h&Ind
hymn	0.106	1	gIm	hIm
khan	0.107	1	gVrn	kVrn
nude	0.104	4	njUUz	njUUd
shed	0.223	1	SVd	SEd
sigh	0.161	1,4	T&I	s&I

Table 2.6: Errors with frequency $\geq$ 0.1 for the fixation network. Note that here the frequencies are the log frequencies as calculated from the CELEX database. Hence there is a slight difference between the frequency shown and the actual frequency at the different fixation positions.

2.2.3 Errors by Phoneme Length

Table 2.7 shows the errors by phoneme length for both networks. This table is included in the report because it is interesting to see for which target length the network has the most problems. Intuitively one would imagine that the most errors would occur with long phonemes since the there are more phonetic features to be learned and hence more possibilities to make a mistake. To be able to compare the networks' performance at different phoneme length, the table contains a column labelled percentage, which gives the percentage of errors of a particular length. For example for the control network there are 1775 events of phoneme length 3. The number of errors is 41 which is 2.31% of the total number of events for length 3.

However, the results indicate that learning words which have a larger feature bundle associated with them is not necessarily harder than learning words with a shorter phoneme. Thus, for both networks the highest percentage of errors (2.31% for control and 2.99% for fixation) were made on words with three phonetic features, whereas the 4 and 5 letter words have comparably low error percentages.

2.2.4 Different Output Positions

Table 2.8 show the errors the network made at the different output slot positions. As described before, the phonological output of the network is divided into six slots in total with an onset-nucleus-coda structure. The onset consists of two consonants, the nucleus of two vowels and finally the coda of two consonants again. For a word that has only one phonetic feature in:

- the onset position, the first slot is occupied (i.e. the onset is left aligned)
- the nucleus position, the first slot is occupied (i.e. the nucleus is left aligned)
- the coda position, the second slot is occupied (i.e. the coda is right aligned.

a. Control Net

Phoneme Length	Number of Words	Number of Events	Errors	Percentage
2	3	15	0	0%
3	355	1775	41	2.31%
4	1502	7510	92	1.22%
5	100	500	6	1.2%

b.Fixation Net

2	3	15	0	0%
3	355	1775	53	2.99%
4	1502	7510	73	0.97%
5	100	500	10	2%

Table 2.7: Errors by phoneme length for both networks. The percentage column gives the percentage of errors of the events of that length. This ensures that the performance of the networks on words with different phoneme length can be compared.

Phonetic Feature		Control Network	Fixation Network
Onset	1	53	51
	2	0	0
Total		53	51
Nucleus	1	29	29
	2	14	12
Total		43	41
Coda	1	2	3
	2	31	37
Total		33	40

Table 2.8: Errors made by the network in the different slots of the output pattern. The output is subdivided into the onset, nucleus and coda positions. Each of these positions has two slots associated with it.

So a word like ’back’, whose phoneme is ’b&k’ will have a target output b.|&.|.k, where ’.’ indicates an empty slot and ’|’ separates onset|nucleus|coda. The reason for left aligning the the onset but right aligning the coda is to have a clear mapping between the exterior letters and the exterior phonetic features. In this way, the first letter of the word (if it is a consonant) will always correspond to the first phonetic feature and the last letter (if it is a consonant) will always correspond to the last phonetic feature. Right aligning the coda in this way gives more structure to the input-output mapping. If for example the coda was left aligned the last letter would sometimes be mapped to the first coda slot (when there are two consonants in the coda) and sometimes to the second coda slot (when there is only one consonant in the coda).

Again, the results are not conclusive as to a difference between the two networks. The fact that the fixation network was trained on more words fixated to the right of centre and hence with most of the word in the left visual field and the right hemisphere does not seem to have an effect on the performance in the different output regions. The only noticeable difference is in the coda slot, where the fixation network makes 7 more errors than the control network. This can be explained by the fact that the fixation net receives proportionally more words in the input that are fixated at the third position. This means that the last letter of the word is the only letter in the right field (left hemisphere). This makes the task faced by the left hemisphere of getting the right pronunciation for the letter more difficult because it does not have information about the previous letters. For the network to compute the correct sound nevertheless, the connections between the two hidden layers are crucial. Because the net sees the last letter of a word proportionally more often alone in the right visual field, the exact pronunciation of this letter is harder to learn, which is reflected in table 2.8.

Table 2.8 also shows that both network make the most errors for phonetic features occurring in the onset. An obvious difference between the onset/coda slots and the nucleus slot is that the nucleus slot consits entirely of vowels whereas the other two consist entirely of consonants. Since there are only 8 different phonetic features for the vowels compared to 24 for the consonants (see figure 2.2). It should therefore be easier to for the network to learn the vowels since they appear at a much higher frequency. In this case the expected results should clearly be noticeably less errors in the nucleus position versus the other two positions. The reason for the high error rate in the nucleus slot lies in the unpredictability of the pronunciation of vowels in the English language (e.g. hint, mint **but:** pint). This makes it a very difficult task for the network to learn the correct pronunciation of the vowels in a word, especially since their pronunciation depends on the preceding and the following letters, Thus the hidden layers of the networks have to encode bigrams and trigrams to correctly compute the output for the vowels. The pronunciation of the consonants on the other hand is much more predictable. The difficulty for the consonants lies in the fact that there are so many different ones (24). The difference between errors in the onset slot (53 for control; 51 for fixation) and the coda slot (33 for control; 40 for fixation) arise the fact that the

Fixation Position	Control Network	Fixation Network	Change
1	29	24	-4
2	23	27	+4
3	24	27	+3
4	30	25	-5
5	33	33	+0

Table 2.9: Errors by fixation position. The last column, labelled 'change' gives the difference in errors between the control and the fixation network (errors of fixation - errors of control).

Fixation Position	Control Network	Fixation Network	Change
1	0.177	0.199	+0.022
2	0.166	0.212	+0.046
3	0.177	0.202	+0.025
4	0.179	0.187	+0.008
5	0.170	0.190	+0.020

Table 2.10: Mean Sum Squared Error for the whole training set by fixation position. Again the last column gives the difference in mse between the two networks.

beginning of English words are less predictable and thus harder to learn than the more predictable ending of words. Thus the higher error rate for the onset versus the coda.

2.2.5 Errors by Fixation Position

An interesting statistic for comparing the two different networks to be able to tell if they actually behave differently is to look at how the networks behave at the different fixation positions. This has been tested in two different ways: table 2.9 shows the number of errors the networks made at the different fixation positions while table 2.10

gives the mean sum-squared error for the whole training set at each of the fixation positions.

Control Network As can be seen from table 2.9, the control network makes less errors if the word is fixated slightly left of centre than it does when the word is fixated right of centre. According to this table, the best fixation positions at which to present the word to the network would be positions 2 or 3 - fixated before the 2nd and 3rd letter respectively. Comparing this to the mean sum-square errors given in table 2.10 one can see that once again, the error score for fixation position 2 is lowest of all. However, for example fixation position 5 has the 2nd lowest mse, although the network makes the most errors at this position.

Taken together, these results (lowest error score and lowest mse at fixation position 2) point to a preferred viewing location on or slightly left of centre. At a first glance this is surprising since the words were presented with the same frequency at each of the positions. If there is a preferred position, one would rather expect that it is easier for the network to learn words that are entirely in one of the two hemispheres. This is because in such a case all of the information needed to identify the word is transmitted to the corresponding hidden layer. Thus in theory there would not need to be a correspondence with the other hidden layer. One would expect that if the word is divided into the two layers, the process of 're-uniting' the word would be a harder process.

However, a preferred viewing position slightly left of centre corresponds exactly to Nazir (2000) and Shillcock et al. (2000) as described in section 2.3. It is interesting that this preferred viewing position should emerge from the network without any initial conditions that would bias it towards this behaviour. It suggests that, at least in the network, this behaviour emerges solely from the structure of English words and seems to support Shillcock et al. (2000) in that a fixation position slightly left of centre is

Fixation Position	Example	Frequency
1	-WORD	0.177
2	W-ORD	0.196
3	WO-RD	0.206
4	WOR-D	0.221
5	WORD-	0.200

Table 2.11: Frequencies with which four-letter words are fixated at different fixation positions. In the example, '-' indicates the fixation position (same as table 1.1).

beneficial for visual word recognition since it allows equal amounts of information about the word to go into each hemisphere.

Fixation Network For the fixation network, the situation is slightly different than for the control network. Compared to the control, it comits less errors at fixation positions 1 and 4, more errors at fixation positions 2 and 3 and the same number of errors at position 5. Similarly, the mean sum-square error given in table 2.10 for the fixation network has the lowest score at position 4 with 0.187 and the highest at position 2 with 0.212. Note that, overall, the average mse for the fixation postions is higher in this network (0.198) than in the control (0.174). This seems to suggest that it was harder for this network to learn when a condition on the fixation positions is imposed. However in total the fixation network has 3 errors less than the control (see table 2.1). Hence the added condition resulted in basically the same number of words being learned, however with less 'certainty'.

In order to be able to compare this data to the frequency used for each fixation position in this network, table 1.1 from section 2 is reproduced here as table 2.11. Looking at table 2.11, most of the changed statistics in the fixation network make sense. In this network, words have been presented with the highest frequency slightly right of centre at position 4. Consequently, this position has the lowest mse and less errors than in the control network whereas for example positions 2 and 3 have higher mse's and higher error rates. Lookin at the change in the errors as well as in the mse, the degree of change between the networks mostly matches the relative fixation frequency at these positions. The only position which is clearly contrary to what would be expected from the fixation data is position 1. This position has been the least frequent to be presented to the network. Nevertheless, the fixation network makes less errors and the mse change is less than would be expected for this position.

2.2.6 Homographs

The training corpus contained 14 homographs (70 events since there are 5 different fixation positions for each word). Homographs are words with the same orthography but a different phonology (e.g. read in 'to read' → /riid/ or 'to have read' → /rEd/). These have not been included in the discussion so far. The reason for treating the homographs separately is the following. Since the network has no way of knowing in which context the homographs appear, it is impossible for it to learn the correct pronunciation for both the phonemes of the word. For this reason these words were excluded from the previous analysis because they would inevitably introduce some errors because the network simply could not learn both pronunciations. However, it is interesting to look at the homographs on their own. More precisely, the question is whether the network is able at all to learn at least one of the possible pronunciations correctly since the two different possibilities will compete and interfere with each other during training. For words where one of the pronunciation has a much higher frequency than the other one this shouldn't be a problem and the network is expected to learn the pronunciation with the higher frequency since that is the one that will dominate during training with little interference from the low frequency one. On the other hand, when both frequencies are similar, both possibilities will be presented roughly the same number of times during training and it will be hard for the model to settle for one of the two at the end of

Homographs

	Control Network	Fixation Network
Total Events	70	70
Correct	67	60
Errors	3	10
Percentage Correct	95.7	85.7

Table 2.12: Results for the homographs in the training corpus for both networks.

training.

Results The results in table 2.12 show that the networks have been able to learn one of the pronunciations for most of the homographs. The control network gets 95.7% correct, whereas the fixation net gets only 85.7%, making 7 errors more. The details of the errors made can be found in table 2.13. As expected almost all of the errors made on the homographs were made for words where the frequencies of the two different possible pronunciations are similar, with only a few exceptions (e.g. ’lead’ where /liid/ has a frequency of 0.408 and /lEd/ has one of 0.235). It can also be seen from the tables that errors occurred because the two possible target outputs interfered with each other. Thus for example the control net pronounces the word ’dove’ as /d@Vv/ at fixation position 1. The two possible correct phonemes for ’dove’ are either /dVv/ with a frequency of 0.054 or /d@Uv/ with a frequency of 0.050. Clearly, the computed output is a combination of both the correct outputs, having the /@/ from the second and the /V/ from the first output. Similarly the fixation net pronounces for example the word ’used’ as /jUUzd/ at fixation positions 4 and 5. Again this is a combination of the two possible correct phonemes for this word: /jUUst/ with a frequency of 0.585 and /jUUzd/ with a frequency of 0.457.

An point to make is that the networks are able to learn homographs which have similar frequencies. Thus for example the two phonemes for the word ’read’ are /riid/ and

a. Control Net

Word	Possible Targets (Freq)	Fixation Position(s)	Output
dove	/dVv/ (0.054); /d@Uv/ (0.050)	1	/d@Vv/
		2	/dOv/
lead	/liid/ (0.408); /lEd/ (0.235)	2	/lEd/

b. Fixation Net

Word	Possible Targets (Freq)	Fixation Position(s)	Output
bass	/bEIs/ (0.078); /b&s/ (0.070)	1,2,4,5	/bEVs/
dove	/dVv/ (0.054); /d@Uv/ (0.050)	2,4	/dEUv/
poll	/p@Ul/ (0.128); /pOl/ (0.050)	3	/pOVl/
		5	/pOUl/
used	/jUUzd/ (0.585); /jUUst/ (0.457)	4,5	/jUUsd/

Table 2.13: Errors on homographs for both nets. The upper table is for the control net and the lower table is for the fixation net. The second column gives both possible pronunciations for the networks with their respective frequencies. The two pronunciation are separated by a line. The third column gives the fixation position at which the error occurred and finally the last column gives the network's output.

/rEd/ with frequencies 0.443 and 0.473 respectively. Although these frequency values suggest that it would be hard for the models to learn the word because of the competition, both networks mangaged to learn it correctly. Interestingly however, the control learned the word as /rEd/ at all positions whereas the fixation net learned it as /riid/, also at all positions. The reason that it was possible for both nets to learn a different pronunciation for the same word might be the following. The bigram ’ea’ appears with different phonemes in the training corpus, with the most frequent phonetic feature being /ii/ (e.g. neat). This explains the /riid/ output of the fixation net. The combination ’ead’ however appears most often and in higher frequencies as /Ed/ (e.g. dead). Thus the output /rEd/ from the control net.

This is an interesting example that demonstrates a possible difference between the two networks. The control net has seen the word in each position an equal number of times whereas the fixation net has seen it most often at position 4 (rea - d). This would cause a preference in the network to learn the combination ’ea’ in ’read’ as a bigram rather than the trigram ’ead’ and pronounce it as such. This is because when the word is fixated at position 4, ’rea’ will be sent to the left hemisphere and only ’d’ to the right hemisphere. Note that this network is still able to learn words like ’dead’ (/dEd/) where there is only one possible phoneme. But when there are two different possibilities, the pronunciation ’ii’ is easier.

2.2.7 Nonwords

Background. An important part of connectionist modelling of visual word regonition has traditionally been to test the network on a series of nonwords. This is important in order to show that the network has actually learned how to pronounce words rather than having merely stored the correct outputs for the words in the training corpus without having learned anything more general about the words. A connectionist model needs to be able to read nonwords in order to justify the central claim of connectionist visual word recognition, namely that a single system can learn to pronounce both exception and nonwords at a level comparable to human subjects. If the networks were not able to pronounce nonwords, this would validate the claim of the Dual-Route model whereby there have to be two separate processes, one rule-based one that can pronounce nonwords but not exception words and one lexicon-based one that can pronounce exception words but not nonwords (Coltheart et al., 1993; Weekes and Coltheart, 1996). Thus for example the poor nonword performance of the Seidenberg and McClelland (1989) network was one of the major criticisms for that model because the network was far below the average performance of human subjects. This model was subsequently improved and achieved levels of nonword reading comparable to humans (Plaut et al., 1996; Harm and Seidenberg, 1999), showing that it was not a problem inherent of the architecture of this connectionist model. One of the problems of the earlier model which contributed to the poor nonword reading was the use of Wichelfeatures as input and output representations. Since, for both the current networks, a slot-based representation was used, it is expected that the networks will not suffer from the same problem regarding the nonword reading.

As testing corpus, Glushko's nonwords were used (Glushko, 1979) as seen in Plaut et al. (1996). All nonwords used with their acceptable pronunciations can be seen in appendix D. The non-words are divided into two types: consistent and inconsistent nonwords. A consistent word is a word whose orthography-to-phonology correspondence is consistent with the one of its orthographic neighbours. If this is not the case, then the word is inconsistent. Note that in the present study this difference has not been made for the words in the original training corpus. However, since the nonwords were already subdivided in this way in Plaut et al. (1996), this division was kept for the nonwords. For testing, a nonword was deemed to be correct if the output of the network matched one of the possible pronunciations of the nonword.

a. Consistent Nonwords

	Control Network	Fixation Network
Total Events	155	155
Correct	132	130
Errors	23	25
Percentage Correct	85.2	83.9

b. Inconsistent Nonwords

	Control Network	Fixation Network
Total Events	175	175
Correct	149	143
Errors	26	32
Percentage Correct	85.1	81.7

Table 2.14: Results for the nonword test on both networks. The nonwords have been subdivided into consistent and inconsistent nonwords. The complete set of the nonwords is listed in appendix D.

a. Consistent Nonwords

Fixation Position	Control Network	Fixation Network	Change
1	4	6	+2
2	6	4	-2
3	4	4	+0
4	3	4	+1
5	6	7	+1

b. Inconsistent Nonwords

Fixation Position	Control Network	Fixation Network	Change
1	4	7	+3
2	5	8	+3
3	2	7	+4
4	10	4	-6
5	5	6	+1

Table 2.15: Nonword errors by fixation position for both consistent and inconsistent nonwords. The last column indicates the difference between the control and the fixation network.

Results The overall results of the nonword testing can be seen in table 2.14. The upper part of the table shows the errors made on the consistent nonwords for both networks and the lower part shows the errors made on the inconsistent nonwords. Both networks were able to pronounce most of the presented nonwords, both consistent and inconsistent, with the control scoring 85.2% on consistent and 85.1% on the inconsistent nonwords. The fixation net scored 83.9% and 81.7% respectively on the consistent and inconsistent nonwords. Again both networks are quite evenly matched with the fixation network getting only slightly more words wrong on both accounts (2 and 6 errors more on the consistent and inconsistent nonwords respectively).

Fixation Position	Example	Real Frequency	New Frequency
1	-WORD	0.177	0.177
2	W-ORD	0.196	0.367
3	WO-RD	0.206	0.467
4	WOR-D	0.221	0.617
5	WORD-	0.200	0.407

Table 2.16: Changed fixation frequencies. The 3rd column shows the real fixation frequencies as used in the previous nets and the 4th column shows the new enhanced frequencies.

Table 2.15 shows the errors made by the networks on nonwords at different fixation positions, with the last column giving the difference between the two networks. As the upper table shows, there is no real difference between the two networks as far as the consistent nonwords are concerned. More interesting for the current paper is the bottom table. There is clearly a tendency for the fixation network to get about 3 or 4 more words wrong at the first three fixation positions. However on position 4 the network makes 6 errors less than the control. Compared to the positive difference at the previous three positions, this can be considered a significant fact. Position 4 is exactly the position where the fixation network was the most exposed to due to the highest fixation position frequency, confirming the results from the words in the training corpus. Even nonwords are easier to pronounce slightly right of centre since the networks was exposed to more words at that position.

2.3 Two additional networks.

From the previous analysis of the two networks, it is clear that the difference between the two networks is very small. To try and get a more significant difference, two more networks were trained for which the differences in the fixation position frequencies were enhanced by 10 (see table 2.16). To calculate the new frequencies, the following formula was used:

$$f_i = x_i + 10 * (x_i - m), \tag{2.2}$$

where f_i is the new frequency, x_i is the old frequency and m is the lowest old frequency value (0.177 in position 1). Note that these new frequencies have no relationship to any real data of fixation positions while people are reading. It is purely introduced to enhance and examine possible difference between the two networks. To reduce training times, the training corpus consisted of 200 words (1000 events) taken from the original corpus chosen at random, corresponding to about 10% of the original corpus. Again training was stopped after 2750 epochs, although tests on the models were done throughout the training to monitor the progress of the networks.

2.3.1 Results

Overall results. After training was complete at 2750 epochs, the control network only made errors on 10 events, pronouncing 'boor' as /bUrr/ instead of /bU@r/ at all positions and 'chic' as /SiIk/ instead of /Siik/ at all positions. Both are low frequency words having a frequency of 0.050 and 0.080 respectively. The control network thus got 99% of the events correct.

The fixation net made errors on the same two words, pronouncing 'boor' as /bUOr/ and 'chic' as /SIIk/ at all positions. Furthermore the fixation net makes one more error on one other word, the word 'pint' (freq 0.145) pronouncing it as /pEInt/ instead of /p&Int/ in all positions. Thus, in total, the network had 15 events wrong, which corresponds to 98.5% correct. The reason why the fixation net gets 'pint' wrong might be the same as the reason why it got the homograph 'read' wrong (see section 3.2.6). Similar words in the smaller corpus all pronounce /pin/ as 'pIn' (e.g. pins). The end letter 't'

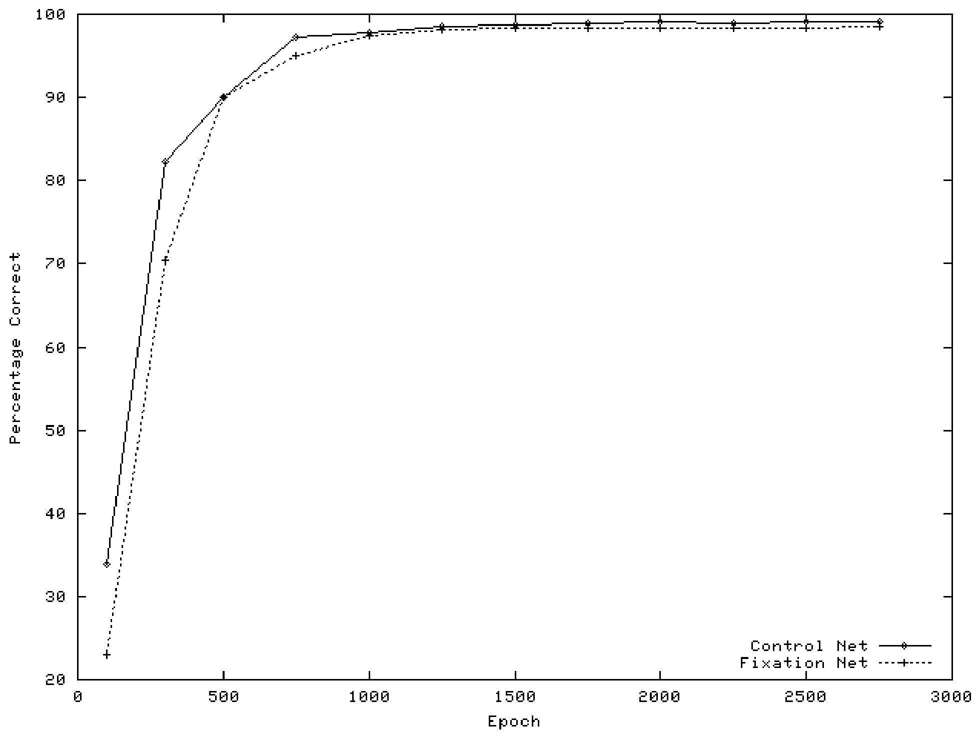

Figure 2.3: Overall results for the two networks. The first measurement was taken at epoch 100 and the last at epoch 2750.

is therfor important for the pronunciation of the word. Since the fixation net sees the word most often as ’pin-t’ it might develop a preference to treat ’in’ as the bigram it has previously learned which would make it more difficult to eventually learn the correct pronunciation. Figure 2.3 shows the evolution of both networks during training. As can be seen from this graph, both networks train fairly quick due to the small size of their training corpus. Also, the control network learns slightly faster than the fixation net. This might be due to the errors introduced in the fixation net by position 1. Position 1 has such a low frequency compared to the other position that it will inevitably lead to more errors at that position than in the control network.

Mean Sum-Square Error The evolution of the mean sum-square error over time at all fixation positions can be seen in tables 2.4, 2.6 and 2.8 for the control net and in figures 2.5, 2.7 and 2.9. Tables 2.4 and 2.5 show the MSE for the respective nets. Both curves tend to zero quite fast, which is expected from the general results of the nets (table 2.3). It can be seen that the MSE curves for the fixation positions are all very similar for the control network. This comes from the fact that there is no preferred fixation position for this net and all the positions have the same frequency. However, for the fixation net, there is a noticeable difference in the MSE curves of the different positions.

To make it easier to analyse the graphs, figures 2.6 and 2.9 show a part of the whole curve for the control network, from epoch 300 to 500 and 2000 to 2750 respectively. The same portion of the curve is shown by figures 2.7 and 2.9 for the fixation network. On these magnified curves, it can be clearly seen that the differences in the curves are larger for the fixation net. In figure 2.7 it can also be seen that the shape of the curves matches their respective frequencies. For example position 4 has the lowest MSE and the highest frequency and position 1 has the highest MSE and the lowest frequency. In the corresponding figure for the control net, there is a slight difference in MSE at epoch 300 but the curves move much closer together at epoch 500. As previously

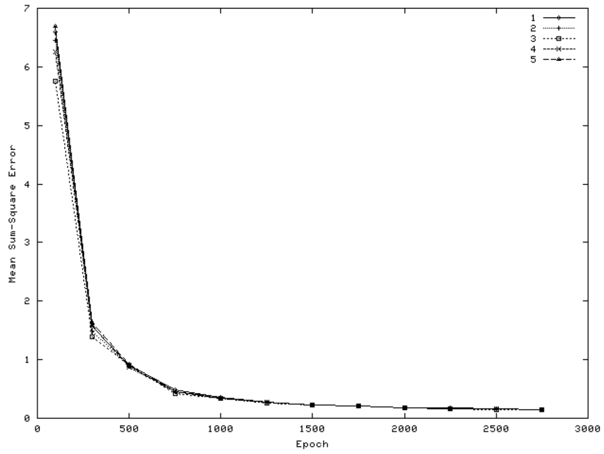

Figure 2.4: Control net: MSE at every fixation position

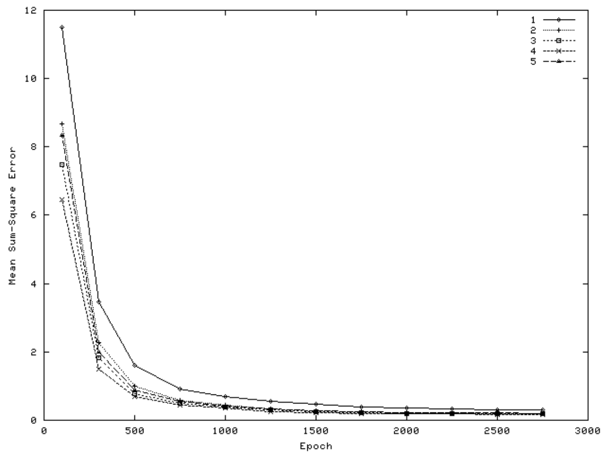

Figure 2.5: Fixation net: MSE at every fixation position

noted, fixations slightly left of centre seem to be easier for the control to learn. This can again be seen at epoch 300 where positions 3 and 2 have the lowest MSEs (in that order). A similar picture can again be seen at the end of training in figures 2.8 and 2.9. The fixation net has the lowest MSEs at positions 4 and 3 which corresponds to its fixation position frequencies and the control net has the lowest MSE's for positions 3 and 2 confirming the finding of the previous model with the complete training set. Furthermore, the curves are further apart for the fixation net, with position 1 being much worse than the rest because it has the lowest frequency.

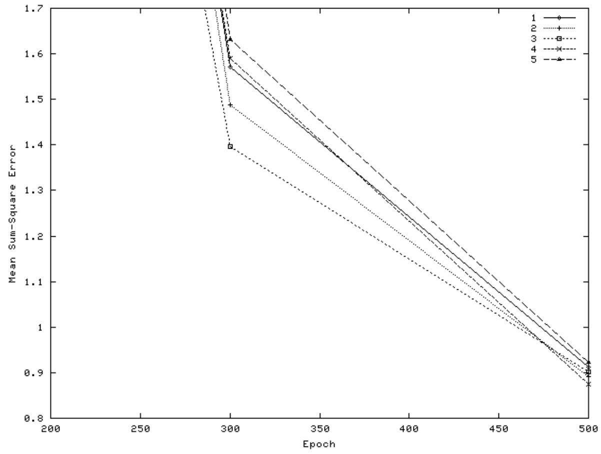

Figure 2.6: Control net: MSE at every fixation position at epochs 300-500

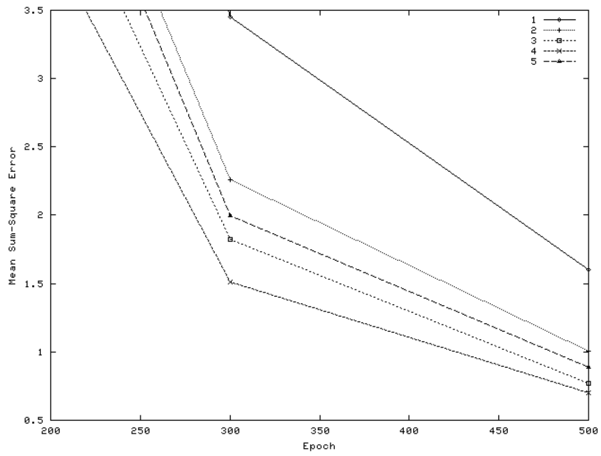

Figure 2.7: Fixation net: MSE at every fixation position at epochs 300-500

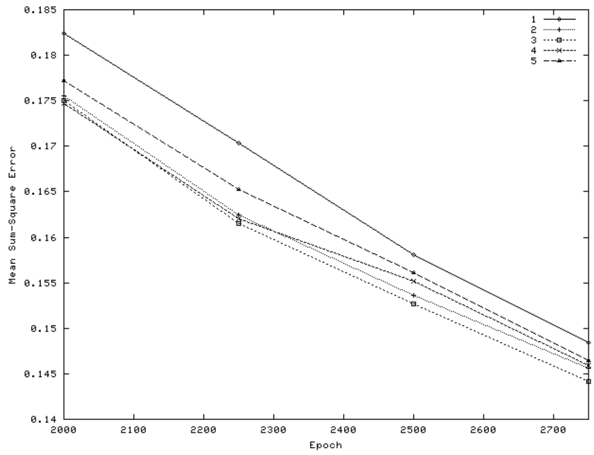

Figure 2.8: Control net: MSE at every fixation position at epochs 2000-2750

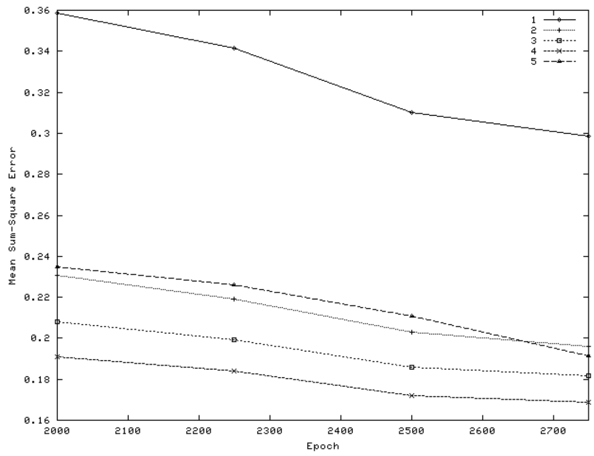

Figure 2.9: Fixation net: MSE at every fixation position at epochs 2000-2750

Note that similar graphs can be drawn for the number of errors made by the networks at the different fixation positions. These graphs would show a similar pictures as the MSE graphs, with the positions having a low MSE also making a low number of errors and vice-versa. To conclude this section it can be said that these networks confirm the findings of the previous networks. There is no noticeable difference in the overall performance of the networks with both getting similar percentages of words correct. However, there is a difference in the more detailed analysis of the networks, namely the performance at the different fixation positions.

Chapter 3

Discussion and Conclusion

This chapter gives an overview and a brief discussion of the results obtained in the previous chapter and it will also present possibilities for future research in this area.

3.1 Discussion

The first part of the discussion involves general results of both networks which do not depend on the different fixation frequencies. The second part of it will look in more detail at the differences and similarities between the control and the fixation net.

General From the two simulations (i.e. the one with a full training corpus and the one with a smaller training corpus), it can be concluded that both the control and the fixation network have accomplished the general task of learning the grapheme-to-phoneme mappings of most of the words in the training corpus. Both networks get around 98.6% of all the words in the training corpus correct. The overwhelming majority of errors were made on low frequency words, which is as expected since the networks have seen these words only a very limited number of times. The tests with different word lists (i.e. regular, irregular and ambiguous words) showed that both networks have successfully learned both regular and exception words. The learning of the exception words is an important test for any connectionist model. Since adherents of the Dual-Route theories claim that it is impossible for a single system to learn both exception and nonwords (e.g. Coltheart et al. (1993)), the fact that the model was able to learn the exception words is a first step to validate the model. The networks were also able to learn most homographs correctly, although the control network was slightly better at this. The difficulty in pronouncing these words comes from the fact that they have two different pronunciations associated with them. This makes them difficult to learn since both pronunciations will interfere with each other during training. Nevertheless the networks were able to learn one of the pronunciation for almost all of the homographs in the training corpus.

The second important test with respect to the claims of the Dual-Route theory is the test on nonwords. The networks were tested on two sets of nonwords, consistent and inconsistent. Both networks scored between 81% and 85% of the nonwords correctly for both sets, with the control net being slightly better than the fixation net. This score is good enough to conclude that both networks have successfully managed to pronounce nonwords. It has to be noted that most connectionist networks are trained on more word presentations. Thus for example the Shillcock and Monaghan (2003) model was trained on 10 million words, whereas the current models were only trained on 4 million words to reduce training times. It is expected that a longer training time would improve the nonword performance of the network since a lot more of the low frequency words especially would be presented more often than in the current simulations. Additionally this would also improve the overall performance of the network on the words in the training corpus, since the majority of errors were made on low frequency words to which the network was not often exposed.

To conclude, it can be said that both networks were able to learn both exception words and nonwords. A single system is responsible for learning both of these types of words.

This validates the networks as models of visual word recognition. Having established this, the next step is to look at the differences between the two networks that were trained.

Differences in the networks. The main part of this paper was to investigate whether there was a significant difference between the control network and the fixation network. Several tests were made to detect any such differences. In both simulations, there was no significant difference found in the overall performance of both networks. The main finding of these tests were that the performance (as measured by the number of errors as well as the mean sum-square error over all the words in the corpus) of the control network was best when the words presented were fixated slightly left of centre. For the fixation net, this tendency was turned around and performance was better for words presented slightly right of centre.

The behaviour of the control network corresponds to the optimal viewing location during single word recognition, where a word at a time is presented to human subjects and the fixation positions are measured (e.g. Nazir (2000)). This is quite interesting since there was no initial bias for the network to evolve in this way. It seems to arise simply out of the structure of English words, where the beginning of a word is usually more informative than the end of the word. A fixation position slightly left of centre will thus ensure that an equal amount of information will be sent into both of the hidden layers, agreeing with Shillcock et al. (2000)'s claim that the optimal viewing location is beneficial to word reading. Note that the same result was obtained in the second simulation with less words in the training corpus, which helps to establish that the previous result was not just a random occurrence.

The preferred fixation position for the fixation net on the other hand is slightly right of centre. This is not surprising since in this network the fixation position with the highest frequency was position 4, just right of the word centre. Thus, during training, the network was presented comparatively more words at position 4 than at any other position. This corresponds to the reading behaviour of humans for 4-letter words when they read words in a text. Note that the differences in the fixation position frequencies were very small. The difference between the highest frequency position (position 4 with a frequency of 0.221) and the lowest frequency position (position 1 with a frequency of 0.177) was only 0.044. Another interesting fact is that the difference between the preferred position of the control network (position 2) and position 4 was only 0.024. Even though these differences are very small, they nevertheless managed to induce a change in the networks behaviour and counter the natural preference of the network for words fixated slightly left of centre. Thus even these very small differences in the initial conditions of the networks manage to change their behaviour. For the second simulation the differences between the frequencies of the fixation positions was enhanced. This simulation confirmed the previous findings. Whereas before, the fixation position right of centre was only slightly preferred, there now was a much bigger preference for a fixation right of centre reflecting the initial conditions of this network.

Another interesting aspect of this simulation was the fact that for the homographs, the control network pronounced the word ’read’ as /rEd/ whereas the simulation net pronounced the word as /riid/ (in all fixation positions). This hints at a difference in the two networks in how information is stored in the hidden layers. Both of these possibilities appeared roughly the same number of times during training. Nevertheless both networks pronounced the word differently. This might come from the fact that the fixation net sees the word more often as ’rea-d’ and thus would use what it has learned about ’ea’, namely that ’ea’ appears most often as /ii/ in the corpus. The control net on the other hand will have a preference to encode the ’ead’ in read as /Ed/ since that is most frequently the case for words ending in ’ead’ (e.g. ’dead’). Note that there would be no problem for either of the nets to learn either of the pronunciations if ’read’ was not a homograph. Since the network here has two possibilities, what it has learned before about other similar words will influence the final result on ’read’. This was the only homograph in the corpus for which this comparison can be made. For all the other homographs, either one of the possibilities had a much higher frequency (and was thus learned) or both had a very low frequency (e.g. both 0.05) and so wouldn't have appeared often enough during training. A similar argument can be applied as to why the fixation network made an error on the word 'pint' in the second simulation with less words.

To conclude it can be said that there is indeed a difference in how both networks evolve during training. The difference was small for the first simulation. However, the second simulation confirmed the differences in the networks by having bigger differences in the fixation positions. The effect is also expected to be bigger for larger words, since the fixation position frequencies for larger words are naturally larger.

3.2 Future Work

The present work can be considered a feasibility study to see if it is worth pursuing this research of combining real fixation data with the split-fovea model of reading. The conclusion drawn before suggests that there are indeed possibilities for future research. An interesting test, which has not been done in the current simulation due to time constraints, would be to lesion the network. More precisely, by cutting the connections between the two hidden layers (i.e. removing the Corpus Callosum), it would be possible to further examine in how far the two networks have encoded the mappings between orthography and phonology differently.

Another direction in which future research could go is to build networks that admit longer words as inputs. Longer words have larger differences in the frequencies of the different fixation positions. Also, the preferred viewing postions for longer words during reading is left of centre as opposed to right of centre for 4-letter words. It would be interesting to see what would come out of a full-scale model including words of different lengths, rather than having a network that is only capable to have 4 letter words as its input.

Appendix A

Regular, Irregular and Ambiguous Word Lists

Word List from Plaut et al. (1996)

Regular	Irregular	Ambiguous	Regular	Irregular	Ambiguous
beam	both	cone	came	bowl	dead
cask	bush	dive	coal	chic	four
cook	comb	gear	cool	come	glow
cord	cost	gone	cove	dead	good
crag	deaf	head	cuff	does	hood
dank	does	know	dare	doll	lone
dark	done	love	days	door	near
dear	foot	pour	deed	four	show
desk	full	year	dole	give	your
dolt	gone	zone	dots	gong	
fact	good		fade	have	
five	head		flat	hind	
flew	hook		form	limb	
fret	lose		glum	lost	
goes	love		goon	most	
grow	move		gull	mule	
hark	none		harm	once	
here	pear		home	pint	
hump	poor		lash	pull	
leaf	said		lisp	says	
lobe	shoe		loom	show	
loss	some		main	swap	
meat	tomb		mend	wand	
moan	want		mole	warm	
moth	warp		mush	wart	
page	wash		paid	wasp	
peel	were		pest	what	

Continued on next page

Word List from Plaut et al. (1996)

Regular	Irregular	Ambiguous	Regular	Irregular	Ambiguous
pine	whom		plod	wolf	
pork	womb		pose	wool	
pump	word		rave	work	
ripe	worm		roam	your	
roll			root		
rune			sand		
sank			slam		
slip			sole		
soon			stop		
swig			tell		
tint			toad		
tote			wade		
wake			wane		
wean			week		
weep			weld		
when			wick		
will			wilt		
wing			wink		
wipe			with		

Word List (split into two to save space)

Appendix B

Errors of the control network

Errors of the control network.

Word	Frequency	Fixation Position	Computed Output	Target Output
adze	0.050	5	Odz	&dz
airs	0.050	3	EOz	E@z
aped	0.050	4	EVpt	EIpt
arse	0.054	5	Vrz	Vrs
arse	0.054	4	Vrz	Vrs
arse	0.054	2	Vrz	Vrs
auks	0.050	5	Orz	Orks
bead	0.050	4	biId	biid
buys	0.095	4	b&Id	b&Iz
chef	0.054	5	JEf	SEf
chef	0.054	4	JEf	SEf
chef	0.054	3	JEf	SEf
chic	0.080	5	Jiik	Siik
chid	0.050	2	SId	JId
chis	0.050	4	k&Id	k&Iz
chis	0.050	3	J&Iz	k&Iz
chis	0.050	1	J&Iz	k&Iz
chit	0.050	2	SIt	JIt
chit	0.050	1	SIt	JIt
chug	0.050	4	JVk	JVg
cite	0.051	5	t&It	s&It
coax	0.050	1	kiUks	k@Uks
coup	0.110	5	kUUk	kUU
coup	0.110	4	kUUk	kUU
coup	0.110	3	kUUk	kUU
coup	0.110	2	kUUk	kUU
coup	0.110	1	kUUk	kUU
czar	0.050	5	sVrr	zVrr
czar	0.050	4	sVrr	zVrr

Continued on next page

Errors of the control network.

Word	Frequency	Fixation Position	Computed Output	Target Output
czar	0.050	3	sVrr	zVrr
czar	0.050	2	sVrr	zVrr
czar	0.050	1	sVrr	zVrr
delf	0.050	4	zElf	dElf
dose	0.109	4	d@Uz	d@Us
dose	0.109	3	d@Uz	d@Us
dose	0.109	2	dEUz	d@Us
dose	0.109	1	d@Uz	d@Us
dost	0.050	5	dOst	dVst
dost	0.050	4	dOst	dVst
dost	0.050	3	dOst	dVst
dost	0.050	2	dOst	dVst
dost	0.050	1	dOst	dVst
doth	0.055	5	dVs	dVT
ewes	0.050	5	NUUz	jUUz
eyot	0.050	3	VIt	EIt
eyot	0.050	2	VIt	EIt
fete	0.050	5	fiIt	fEIt
fete	0.050	4	fiIt	fEIt
fete	0.050	3	fiIt	fEIt
gels	0.050	5	gElz	_Elz
gels	0.050	4	gElz	_Elz
gels	0.050	3	gElz	_Elz
gels	0.050	2	gElz	_Elz
gels	0.050	1	gElz	_Elz
gems	0.050	5	gEmz	_Emz
gems	0.050	4	gEmz	_Emz
gems	0.050	1	gEmz	_Emz
germ	0.055	5	g@rm	_@rm
germ	0.055	4	g@rm	_@rm
germ	0.055	3	g@rm	_@rm
germ	0.055	2	g@rm	_@rm
germ	0.055	1	g@rm	_@rm
gibe	0.050	1	g&Ib	_&Ib
gins	0.050	5	gInz	_Inz
gins	0.050	2	gInz	_Inz
gins	0.050	1	gInz	_Inz
gist	0.050	5	gIst	_Ist
gist	0.050	4	gIst	_Ist
gnus	0.050	1	nVz	nUUz
gush	0.050	3	gVJ	gVS
gyms	0.050	1	gImz	_Imz
hush	0.088	3	hVJ	hVS
iced	0.061	4	&Izt	&Ist

Continued on next page

Errors of the control network.

Word	Frequency	Fixation Position	Computed Output	Target Output
john	0.050	5	gOn	_On
john	0.050	4	gOn	_On
khan	0.107	1	JVrn	kVrn
kris	0.050	3	kriiz	kriis
kris	0.050	2	kriiz	kriis
kris	0.050	1	Jriiz	kriis
lewd	0.050	2	lUUd	ljUUd
lien	0.050	5	lIin	lI@n
lien	0.050	4	lIin	lI@n
lien	0.050	3	lIin	lI@n
lien	0.050	2	lIin	lI@n
lien	0.050	1	lEin	lI@n
lieu	0.050	1	ljEI	ljUU
luge	0.050	5	lUUD	lUUZ
luge	0.050	3	lUUD	lUUZ
luge	0.050	2	lUU_	lUUZ
luge	0.050	1	lUU_	lUUZ
oast	0.050	4	@Uts	@Ust
ooze	0.050	5	UUs	UUz
pals	0.050	1	p&ls	p&lz
pooh	0.050	5	pUU	phUU
pooh	0.050	4	pUU	phUU
pooh	0.050	3	pUU	phUU
quay	0.063	5	kEi	kii
quay	0.063	4	kEi	kii
quay	0.063	3	kEi	kii
quay	0.063	2	kEi	kii
quay	0.063	1	kEi	kii
rhea	0.050	5	rE@	rI@
rhea	0.050	4	rE@	rI@
rhea	0.050	3	rE@	rI@
rhea	0.050	2	rE@	rI@
roux	0.050	5	rUUS	rUU
roux	0.050	4	rUUS	rUU
roux	0.050	3	rUUk	rUU
sear	0.050	5	sE@r	sI@r
sohs	0.050	1	s@Us	s@Uz
spiv	0.050	1	spIf	spIv
suet	0.050	5	sUUt	sUIt
suet	0.050	4	sUUt	sUIt
suet	0.050	3	sUUt	sUIt
suet	0.050	2	sUUt	sUIt
suet	0.050	1	sUUt	sUIt
swab	0.050	5	sw&b	swOb

Continued on next page

Errors of the control network.

Word	Frequency	Fixation Position	Computed Output	Target Output
swiz	0.050	5	swId	swIz
talc	0.050	5	t&Vlk	t&lk
thaw	0.052	2	DOr	TOr
thru	0.050	1	srUU	TrUU
thud	0.054	5	DVd	TVd
thud	0.054	4	DVd	TVd
thud	0.054	3	DVd	TVd
tows	0.050	1	s@Uz	t@Uz
tsar	0.050	5	sVrr	zVrr
tsar	0.050	4	sVrr	zVrr
tsar	0.050	3	sVrr	zVrr
tsar	0.050	2	sVrr	zVrr
tsar	0.050	1	sVrr	zVrr
veld	0.050	5	vEld	vElt
veld	0.050	4	vEld	vElt
veld	0.050	3	vEld	vElt
veld	0.050	2	vEld	vElt
veld	0.050	1	vEld	vElt
waft	0.050	2	wOrft	wVrft
wend	0.050	1	wVnd	wEnd
wops	0.050	2	wVps	wOps
zoom	0.050	4	dUUm	zUUm

Errors of the control network.

Appendix C

Errors of the fixation network

Errors of the fixation network.

Word	Frequency	Fixation Position	Computed Output	Target Output
ache	0.086	4	EIJ	EIk
ache	0.086	3	EIJ	EIk
ache	0.086	2	EVk	EIk
ache	0.086	1	EIJ	EIk
aide	0.096	1	OId	EId
alms	0.050	2	Omz	Vrmz
arse	0.054	5	Vrz	Vrs
arse	0.054	4	Vrz	Vrs
arse	0.054	3	Vrz	Vrs
arse	0.054	2	Vrz	Vrs
arse	0.054	1	Vrz	Vrs
bade	0.050	5	bEd	b&d
beau	0.050	5	biU	b@U
beau	0.050	3	biU	b@U
beau	0.050	2	biU	b@U
beau	0.050	1	biU	b@U
boor	0.050	5	bOOr	bU@r
boor	0.050	1	bOrr	bU@r
char	0.050	1	SVrr	JVrr
chef	0.054	5	JEf	SEf
chef	0.054	4	JEf	SEf
chef	0.054	3	JEf	SEf
chef	0.054	2	JEf	SEf
chew	0.093	3	SUU	JUU
chew	0.093	2	SUU	JUU
chic	0.080	5	Jiik	Siik
chic	0.080	1	Tiik	Siik
chid	0.050	2	SId	JId
chid	0.050	1	SId	JId

Continued on next page

Errors of the fixation network.

Word	Frequency	Fixation Position	Computed Output	Target Output
chis	0.050	5	J&Iz	k&Iz
chis	0.050	4	J&Iz	k&Iz
chis	0.050	3	J&Iz	k&Iz
choc	0.050	4	SOk	JOk
choc	0.050	3	SOk	JOk
choc	0.050	2	SOk	JOk
chum	0.063	1	SVm	JVm
cons	0.050	1	kVnz	kOnz
coup	0.110	5	kUUJ	kUU
coup	0.110	4	kUUJ	kUU
coup	0.110	3	kUUJ	kUU
coup	0.110	2	kUUJ	kUU
coup	0.110	1	kUUJ	kUU
cyst	0.050	3	tIst	sIst
czar	0.050	5	sVrr	zVrr
czar	0.050	3	sVrr	zVrr
doge	0.050	4	d@Ug	d@U_
dons	0.074	4	dVnz	dOnz
dose	0.109	3	d@Uz	d@Us
dose	0.109	2	d@Uz	d@Us
dose	0.109	1	d@Uz	d@Us
dots	0.074	5	dVts	dOts
duct	0.050	5	dVJt	dVkt
duds	0.050	5	dVds	dVdz
fete	0.050	5	fIIt	fEIt
flex	0.063	1	plEks	flEks
gash	0.050	2	k&S	g&S
gist	0.050	3	gIst	_Ist
gist	0.050	2	gIst	_Ist
gist	0.050	1	gIst	_Ist
heir	0.110	5	I@r	E@r
heir	0.110	4	I@r	E@r
heir	0.110	3	I@r	E@r
heir	0.110	2	I@r	E@r
heir	0.110	1	I@r	E@r
hind	0.113	4	h&Vnd	h&Ind
hymn	0.106	5	gIm	hIm
iced	0.061	5	&Isd	&Ist
iced	0.061	4	&Isd	&Ist
iced	0.061	3	&Isd	&Ist
iced	0.061	2	&Isd	&Ist
josh	0.050	3	dOS	_OS
joss	0.050	5	JOs	_Os
khan	0.107	3	gVrn	kVrn

Continued on next page

Errors of the fixation network.

Word	Frequency	Fixation Position	Computed Output	Target Output
knob	0.069	4	nOp	nOb
kris	0.050	5	kriiz	kriis
kris	0.050	4	kriiz	kriis
kris	0.050	3	kriiz	kriis
kris	0.050	2	kriiz	kriis
kris	0.050	1	kriiz	kriis
leys	0.050	2	liIz	lEIz
lien	0.050	5	lIin	lI@n
lien	0.050	4	lIin	lI@n
lien	0.050	3	lIin	lI@n
lien	0.050	2	lIin	lI@n
lien	0.050	1	lEin	lI@n
luge	0.050	5	lUU_	lUUZ
luge	0.050	4	lUU_	lUUZ
luge	0.050	2	lUU_	lUUZ
lute	0.050	5	liUUt	lUUt
mule	0.083	3	mjUIl	mjUUl
niff	0.050	5	mIf	nIf
nude	0.104	4	njUUz	njUUd
oast	0.050	4	@Utt	@Ust
oath	0.092	4	@Us	@UT
ooze	0.050	5	UUs	UUz
phew	0.050	1	sjUU	fjUU
phiz	0.050	2	fEz	fIz
pooh	0.050	4	pUU	phUU
quay	0.063	5	kEi	kii
rasp	0.050	3	rVrtp	rVrsp
rein	0.051	4	rVIn	rEIn
rend	0.050	3	rVnd	rEnd
sass	0.050	1	s&z	s&s
shed	0.223	5	SVd	SEd
sigh	0.161	5	T&I	s&I
sigh	0.161	4	T&I	s&I
smog	0.050	2	tmOg	smOg
sohs	0.050	3	s@Us	s@Uz
spiv	0.050	1	spIf	spIv
thew	0.050	5	DjUU	TjUU
thru	0.050	4	DrUU	TrUU
thru	0.050	3	DrUU	TrUU
thru	0.050	1	DrUU	TrUU
thug	0.050	5	DVg	TVg
thug	0.050	3	sVg	TVg
toad	0.062	2	s@Ud	t@Ud
tsar	0.050	5	sVrr	zVrr

Continued on next page

Errors of the fixation network.

Word	Frequency	Fixation Position	Computed Output	Target Output
tsar	0.050	4	tVrr	zVrr
tsar	0.050	3	sVrr	zVrr
tsar	0.050	2	sVrr	zVrr
tsar	0.050	1	sVrr	zVrr
urns	0.050	5	Ornz	@rnz
vacs	0.050	5	f&ks	v&ks
vase	0.074	5	vEUz	vVrz
vase	0.074	4	vErz	vVrz
vase	0.074	3	vErz	vVrz
vase	0.074	2	vErz	vVrz
veld	0.050	5	vEld	vElt
veld	0.050	4	vEld	vElt
veld	0.050	3	vEld	vElt
veld	0.050	2	vEld	vElt
veld	0.050	1	vEld	vElt
void	0.096	2	fOId	vOId
yeas	0.050	2	jEiz	jEIz
zoom	0.050	2	sUUm	zUUm
zoom	0.050	1	sUUm	zUUm

Errors of the fixation network.

Appendix D

Nonword List

Glushko (1979)'s nonwords with their pronunciation.

Inconsistent Nonwords		Consistent Nonwords	
BILD	/b&Ild/; /bIld/	BEED	/biid/
BINT	/b&Int/; /bInt/	BELD	/bEld/
BLEAD	/bliid/; /blEd/	BINK	/bINk/
BOOD	/bUUd/; /bVd/; /bUd/	BLEAM	/bliim/
BOST	/b&Ust/; /bVst/; /bost/	BORT	/b&Urt/
BROVE	/br&Uv/; /brUUv/; /brVv/	BROBE	/br&Ub/
COSE	/k&Us/; /k&Uz/; /kUUz/	CATH	/k&T/; /kOT/
COTH	/k&UT/; /koT/	COBE	/k&Ub/
DERE	/dEIr/; /diir/; /dUr/	DOLD	/d&Uld/; /dOld/
DOMB	/d&Um/; /dUUm/; /dOm/; /dOmb/	DOON	/dUUn/
DOOT	/dUUt/; /dUt/	DORE	/d&Ur/
DROOD	/drUUd/; /drVd/; /drUd/	DREED	/driid/
FEAD	/fiid/; /fEd/	FEAL	/fiil/
GOME	/g&Um/; /gVm/	GODE	/g&Ud/
GROOK	/grUUk/; /grUk/	GROOL	/grUUl/; /grUl/
HAID	/h&d/; /hEId/; /hEd/	HEAN	/hiin/
HEAF	/hiif/; /hEf/	HEEF	/hiif/
HEEN	/hiin/; /hIn/	HODE	/h&Ud/
HOVE	/h&Uv/; /hUUv/; /hVv/	HOIL	/hOIl/
LOME	/l&Um/; /lVm/	LAIL	/lEIl/
LOOL	/lUUl/; /lUl/	LOLE	/l&Ul/
MEAR	/mEIr/; /miir/	MEAK	/mEIk/; /miik/
MONE	/m&Un/; /mVnmon/	MOOP	/mUUp/
MOOF	/mUUf/; /mUf/	MUNE	/mUUn/; /mjUUn/
NUSH	/nVS/; /nUS/	NUST	/nVst/
PILD	/p&Ild/; /pIld/	PEET	/piit/
PLOVE	/pl&Uv/; /plUUv/; /plVv/	PILT	/pIlt/
POMB	/p&Um/; /pUUm/; /pOm/; /pOmb/	PLORE	/pl&Ur/
PODE	/p&Ud/	PRAIN	/prEIn/

Continued on next page

Glushko (1979)'s nonwords with their pronunciation.

Inconsistent Nonwords		Consistent Nonwords	
POOT	/pUUt/; /pUt/	SHEED	/Siid/
POLD	/p&Uld/; /pOld/	SOAD	/s&Ud/; /sod/
POVE	/p&Uv/; /pUUv/; /pVv/	SPEET	/spiit/
PRAID	/pr&d/; /prEId/; /prEd/	STEET	/stiit/
SHEAD	/Siid/; /SEd/	SUFF	/sVf/
SOOD	/sUUd/; /sVd/; /sUd/	SUST	/sVst/
SOST	/s&Ust/; /sVst/; /sost/	SWEAL	/swiil/
SPEAT	/spEIt/; /spiit/; /spEt/	TAZE	/tEIz/
STEAT	/stEIt/; /stiit/; /stEt/	WEAT	/wEIt/; /wiit/; /wEt/
SULL	/sVl/; /sUl/	WOSH	/wOS/
SWEAK	/swEIk/; /swiik/	WOTE	/w&Ut/
TAVE	/t&v/; /tEIv/; /tOv/	WUFF	/wVf/
WEAD	/wiid/; /wEd/		
WONE	/w&UnwVn/; /won/		
WULL	/wVl/; /wUl/		
WUSH	/wVS/; /wUS/		

Nonwords with their pronunciation.

Bibliography

R.H. Baayen, R. Pipenbrock, and L. Gulikers. *The CELEX Lexical Database (CD-ROM)*. Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA, 1995.

Anne Castles and Max Coltheart. Varieties of developmental dyslexia. *Cognition*, 47: 149–180, 1993.

Morten H. Christiansen and Nick Chater. Connectionist natural language processing: The state of the art. *Cognitive Science*, 23(4):417–437, 1999.

Max Coltheart, Brent Curtis, Paul Atkins, and Michael Haller. Models of reading aloud: Dual-route and parallel-distributed-processing approaches. *Psychological Review*, 100(4):589–608, 1993.

Max Coltheart, Jacquelin Masterson, and Sally Byng. Surface dyslexia. *Quarterly Journal of Experimental Psychology*, 35A:469–495, 1983.

Chadley K. Dawson, Randall C. O'Reilly, and James L. McClelland. *The PDP++ Software Users Manual*, September 2001. Version 2.1.

Michael S. Gazzaniga. Cerebral specialisation and interhemispheric communication. does the corpus callosum enable the human condition? *Brain*, 123:1293–1326, 2000.

R.J. Glushko. The organization and activation of orthographic knowledge in reading aloud. *Journal of Experimental Psychology: Human Perception and Performance*, 5(4):674–691, 1979.

Michael W. Harm and Mark S. Seidenberg. Phonology, reading acquisition, and dyslexia: Insights from connectionist models. *Psychological Review*, 106(3):491–528, 1999.

David Howard and Wendy Best. Developmental phonological dyslexia: Real word reading can be completely normal. *Cognitive Neuropsychology*, 6(6):887–934, 1996.

Alex Leff. A historical review of the representation of the visual field primary cortex with special reference to the neural mechanisms underlying macular sparing. *Brain and Language*, in press.

Scott A. McDonald and Richard Shillcock. Anatomical constraints on eye movement behaviour: the implications of foveal splitting for saccade planning in reading. submitted.

Marie Montant, Tatjana A. Nazir, and Michel Poncet. Pure alexia and the viewing position effect in printed words. *Cognitive Neuropsychology*, 15(1/2):93–140, 1998.

Tatjana A. Nazir. Traces of print along the visual pathway. In A. Kennedy, R. Radach, D. Heller, and J. Pynte, editors, *Reading as a Perceptual Process*, pages 3–22. Elsevier Science, 2000.

J. K. O'Reagan and A. M. Javobs. Optimal viewing position effect in word recognition: A challenge to current theory. *Journal of Experimental Psychology: Human Perception and Performance*, 10:1129–1157, 1992.

Randall C. O'Reilly and Yuko Munakata. *Computational Explorations in Cognitive Neuroscience: Understanding the Mind by Simulating the Brain*. MIT Press, 2000.

David C. Plaut. A connectionist approach to word reading and acquired dyslexia: Extension to sequential processing. *Cognitive Science*, 23:543–568, 1999.

David C. Plaut, James L. McClelland, Mark S. Seidenberg, and Karalyn Patterson. Understanding normal and impaired word reading: Computational principles in quasi-regular domains. *Psychological Review*, 103(1):56–115, 1996.

D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

Mark S. Seidenberg. Visual word recognition: An overview. In *Speech, Language and Communication*, chapter 5, pages 137–179. Academic Press, 1995.

Mark S. Seidenberg and James L. McClelland. A distributed, developmental model of word recognition and naming. *Psychological Review*, 96(4):423–568, 1989.

Richard Shillcock, T. Mark Ellison, and Padraic Monaghan. Eye-fixation behaviour, lexical storage and visual word recognition in a split processing model. *Psychological Review*, 107:824–851, 2000.

Richard Shillcock and Padraic Monaghan. The split fovea model of reading: Part1. connectionist simulations of normal reading. unpublished, July 2003.

Brendan Weekes and Max Coltheart. Surface dyslexia and surface dysgraphia: Treatment studies and their theoretical implications. *Cognitive Neuropsychology*, 13(2): 277–315, 1996.

Wayne A. Wickelgren. Context-sensitive coding, associative memory and serial order in (speech) behaviour. *Psychological Review*, 76:1–15, 1969.