

Noise tolerance in a Neocognitron-like network

Ângelo Cardoso*, Andreas Wichert*

*INESC-ID Lisboa and Instituto Superior Técnico, Technical University of Lisbon
Av. Prof. Dr. Aníbal Cavaco Silva, 2744-016 Porto Salvo, Portugal*

Abstract

The Neocognitron and its related hierarchical models have been shown to be competitive in recognizing handwritten digits and objects. However, the tolerance of these models to several types of noise can be low. We will start by briefly overviewing some previous results regarding the tolerance of these models. Afterwards, we report the higher noise tolerance of the winner-take-all response in a hierarchical model over related models. We provide an analysis and interpretation of this tolerance under Bayesian decision theory. Finally, we report on how to further improve recognition for extremely noisy patterns.

Keywords: Neocognitron, hierarchical model, winner-take-all, noise tolerance, lateral competition, selectivity.

1. Introduction

A version of a Neocognitron that greatly increases the robustness of the model against several types of background noise has recently been proposed in (Fukushima, 2011). A related model with winner-take-all responses has been proposed in (Cardoso & Wichert, 2010) with the purpose of achieving the working principles of the Neocognitron in a simplified way. The main difference between this and related hierarchical models is how simple cells respond. Among all simple cells with the same receptive field, only the cell whose preferred stimulus is most similar to the presented stimulus is active

*Corresponding author. Tel.: +351 214233231; Fax: +351 213144843

Email addresses: angelo.cardoso@ist.utl.pt (Ângelo Cardoso),
andreas.wichert@ist.utl.pt (Andreas Wichert)

while the others are silent. The model has been shown to be competitive in handwritten digit recognition (Cardoso & Wichert, 2013).

In this manuscript, we show that the winner-take-all (WTA) response is noise tolerant. We show that this response is the optimal decision in a discrimination task for additive white Gaussian noise using Bayesian decision theory. Simulations show that the WTA response has significantly more noise tolerance than related models (Fukushima, 2003, 2011; Riesenhuber & Poggio, 1999; Serre et al., 2007; Mutch & Lowe, 2008).

2. A hierarchical model with a WTA response

In this section, we briefly describe a hierarchical model that was previously proposed in (Cardoso & Wichert, 2010) and that is closely related to the Neocognitron (Fukushima, 1980, 2003, 2010). The model is composed of two types of cells arranged hierarchically. Simple cells provide selectivity by reacting to a particular stimulus (e.g., oriented lines). Complex cells add invariance to the position of the stimulus, pooling from nearby simple cells reacting to a particular stimulus. The two types of cells are arranged in layers of the same cell type.

All simple and complex layers are unsupervised, and their purpose is to represent the input patterns in a feature space, where patterns that represent similar things are similarly represented. In the context of a classification problem, a supervised classifier then learns a mapping between this unsupervised feature space and the labels.

2.1. Simple cell layer

Simple cells react to a particular preferred stimulus in a particular location. For a particular preferred stimulus, there is a set of cells with the same replicated preference across different locations. Among all cells with the same receptive field, only the cell whose preferred stimulus is most similar (according to the Euclidean distance) to the current one is active (i.e., its response is 1), while other cells that have different preferences in the same location are silent (i.e., their response is 0). This sparse, simple cell response relates to lateral inhibition in biological vision.

To learn the preferred stimulus, all input patterns are tiled with a squared mask. Each position of the mask, representing a receptive field in a given input pattern, results in a sub-pattern. All of the sub-patterns are the inputs

to a clustering algorithm (in this case, K-means) that yields a set of preferred stimuli (or classes), often referred to as a dictionary.

2.2. Complex cell layer

Complex cells perform a pooling operation over simple cells with the same stimulus preference across a region. These cells are active if any of their afferent simple cells are active. All of the afferents to a complex cell react to the same preferred stimulus across nearby positions. If any of the afferents are active, the complex cell is also active. This is equivalent to a maximum operation. Therefore, the complex layer operation is predetermined and is not the result of learning.

2.3. Additional Layers

The training of the network is performed sequentially, i.e., the training of each layer is finished before the next layer is trained, starting from the layer closest to the input. If the network has more than one simple layer, then the output of the first complex layer is used to train the second simple layer analogously. Each cell in the second and following simple layers has afferent connections from complex cells with a different preferred stimulus (e.g., different orientations).

The response of the last layer can be viewed as a feature space to represent an image; images represented in this space can then be used as inputs to a classifier for recognition tasks.

3. Experimental

In this section, we evaluate the noise tolerance of a hierarchical model with a WTA response (Cardoso & Wichert, 2010) in the task of recognizing handwritten digits. We start by evaluating the noise tolerance when the model is not exposed to noise during the learning phase. Afterwards, we evaluate how the performance degrades for extremely high levels of noise and empirically analyze how the noise tolerance relates to the Euclidean distance between patterns and to the correlation between clean and noisy patterns. Finally, we empirically show that exposing the classifier to the responses of the last layer (C2) for noisy patterns can improve the recognition for high levels of noise, as one might expect. A similar result has been reported for a related deep network (Tang & Eliasmith, 2010).

The model has the following parameters: receptive field size – the number of afferents a cell has; shift – the distance between adjacent receptive fields; frame – the number of afferents with a fixed response; and classes – the number of different preferred stimuli in the simple layers. The frame determines the extra area without activity that is added to the input patterns. The frame parameter can be interpreted as a type of preprocessing step that enlarges the original input image by adding background. In the first simple layer, the frame would be equivalent to preprocessing the image to include a border with a background around it. For the following layers, it means that there are additional retinotopic locations where all cells are silent. The rationale for the frame parameter is to allow some receptive fields near the borders of the input to have less actual connections than in the Neocognitron. The frame is not an important property of the model; in fact, the parameterization used for the MNIST dataset not does use it (see Table 3). The parameters are illustrated in Figure 1.

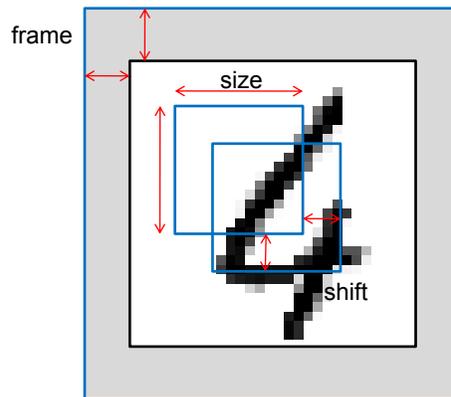


Figure 1: Model parameter illustration. The smaller squares represent two different positions of the mask (or two different receptive fields). The size refers to the size of the mask, and the shift refers to the distance between different positions of the mask. The frame represented by the gray area determines the extra area (or set of retinotopic locations) without activity that is added to the pattern. Additionally, for simple layers, the number of classes k (or preferred stimuli) must also be chosen.

3.1. Noise tolerance

3.1.1. ETL1

We use the ETL1 dataset ¹, which contains 14450 handwritten digits. The images are gray scale, and their size is 64×63 . An additional column is added to the bottom of the patterns to make them square (64×64) by repeating the last one. We linearly rescale all images to $[0,1]$. In all experiments, we add noise only to the digits used to measure the recognition performance and not to the ones used for learning. The preferred stimulus for all layers is learned with K-means (Euclidean distance). Because the ETL1 dataset is not contrast normalized, learning the first simple layer preferences is challenging. To illustrate this process, we show the results of learning preferences for the first simple layer from both gray and binary patches on ETL1 in Figure 2. We can see that, for the gray patches, several preferences are simply different shades of gray. This observation is due to the existence of patterns with significantly different levels of contrast. Making the patterns binary is a simple way to normalize them for contrast. Alternatively, we could use a more complex image processing method to perform contrast normalization, use predetermined preferences for oriented lines as in (Fukushima, 2011) or filter the DC component using on-off cells in a more biologically motivated form of contrast normalization.



Figure 2: A set of 16 preferences learned using K-means on ETL1 for 100000 patches of size 8×8 . On the left, the preferences using binary versions of the patterns. On the right, the preferences using grayscale versions. We can see that the grayscale versions of the patterns produce low-contrast preferences. Several of the preferences are simply different shades of gray.

¹ETL1 database, <http://www.etl.go.jp/etlcdb/index.htm>.

The network parameters (see Figure 1) were set as follows. The number of preferred stimuli in S1 (the first simple layer) is 16, and the number in S2 (the second simple layer) is 100. We fix the shift in simple layers (S1,S2) to 1 and in complex layers (C1,C2) to 2, as in (Fukushima, 2003), representing a 2:1 ratio (i.e., thinning out) of simple cells to complex cells. The receptive field size and frame parameters were obtained using a greedy search and are shown in Table 3. For the greedy search, we used 100 samples (50 for training and 50 for validation) from the entire dataset. We selected the parameters that maximized the recognition rate on the validation sample using a nearest neighbor classifier.

ETL1	no noise	faint digit	lines	white Gaussian
this work	$1.33 \pm 0.18\%$	$1.71 \pm 0.24\%$	$2.62 \pm 0.18\%$	$6.07 \pm 0.47\%$
Orig. Neoc. ^a	$1.40 \pm 0.20\%$	$37\%^b$	$26\%^b$	$32\%^b$
Imp. Neoc. ^a	$1.43 \pm 0.21\%$	$14\%^b$	$7\%^b$	$11\%^b$

^a(Fukushima, 2011)

^b(Fukushima, 2011) – Figure 16

Table 1: ETL1 results: average recognition error and standard deviation with 3000 training and 3000 test patterns over five runs for different types of noise. For the faint digit and line noise, the intensity is noise (p-p) / signal (p-p) = 0.4, and for the white noise, the intensity is the noise (r.m.s) / signal (p-p) = 0.4. The intensity for the several types of noise is clearly illustrated. For the white noise, the gray intensity is linearly rescaled for display. The lowest and highest values are represented by black and white.

The recognition error without noise using 3000 training patterns and 3000 test patterns over 5 runs is $1.33 \pm 0.18\%$. We start by evaluating the tolerance of the model for the same types of noise (see Table 1) as in (Fukushima, 2011) for this dataset. Two different types of noise are added to the background: the faint image of a different digit and random line segments ². The noise intensity for both types of noise is defined as noise (peak-to-peak) divided by signal (peak-to-peak). We also consider superimposed white noise. The white noise intensity is defined as the root-mean-square of the noise divided

²For each pattern, four line segments are added, each starting and ending in random positions in each of the patterns. The thickness of all lines is two pixels.

by the signal (peak-to-peak). The white noise is sampled from a zero-mean Gaussian with a standard deviation equivalent to the root-mean-square of the noise signal.

The recognition error for the different types of noise using a linear classifier is shown in Table 1. The model shows a high tolerance to all of the types of noise considered. The recognition performance is almost unaltered for the faint digit and line noises, while it slightly increases for the white noise. As observed from Table 1, the error produced using this method is lower than for related models.

3.1.2. MNIST

We use the MNIST dataset ³. The images are linearly rescaled to [0,1]. As in the previous section, we add noise only to the test set. The network parameters (see Figure 1) were set as follows. The number of preferred stimuli in S1 (the first simple layer) is 16, and the number in S2 (the second simple layer) is 100. The receptive field size, shift and frame parameters were obtained using a greedy search and are shown in Table 3. The white noise is sampled from a zero-mean Gaussian with a standard deviation of 0.1. For the salt & pepper noise, we randomly set 10% of the pixels to either black or white.

The recognition error for the different types of noise using a linear classifier is shown in Table 2. The recognition error for HMAX (Serre et al., 2007) and modified version of HMAX (Mutch & Lowe, 2008) is also shown in Table 2 according to results presented elsewhere for both models (Hamidi & Borji, 2010). The model shows a high tolerance to both types of noise considered. The error is lower than for related models, namely HMAX (Serre et al., 2007) and a modified version of HMAX (Mutch & Lowe, 2008), which, among other differences, adds inhibition to suppress weaker responses. As in the case of the WTA response, inhibition in this model might also be related to an improved noise tolerance. We should note that the HMAX model was not parameterized specifically for handwritten digits; it is also worth noting that whereas HMAX performs similar to the WTA response for noiseless conditions, it performs considerably worse for noisy inputs. As we argue in detail later in this manuscript, this performance difference may be due to the computation performed in simple cells. The responses are

³<http://yann.lecun.com/exdb/mnist/>



MNIST	no noise	white	salt & pepper
this work	0.71% ^a	1.17%	1.98%
HMAX ^b	2.9% ^c	50% ^c	55% ^c
Modified C2 feat. ^b	1.27%	38% ^c	37% ^c

^a(Cardoso & Wichert, 2013)

^b(Hamidi & Borji, 2010)

^c(Hamidi & Borji, 2010) – Figures 5, 9 and 11

Table 2: MNIST results: average recognition error on the entire dataset with 60000 training and 10000 test patterns for different types of noise. For white noise, we add noise sampled from a zero-mean Gaussian with a standard deviation of 0.1. For the salt & pepper noise, we randomly set 10% of the pixels to either white or black. The intensity for the several types of noise is clearly illustrated. For white noise, the gray intensity is linearly rescaled for display, in which the lowest and highest values are represented by black and white, respectively. The results for HMAX (Serre et al., 2007) and Modified C2. Feat. (Mutch & Lowe, 2008) were both obtained from (Hamidi & Borji, 2010). The noiseless results for this work were obtained from (Cardoso & Wichert, 2013)

independent for the standard HMAX model (Serre et al., 2007), but in the modified version (Mutch & Lowe, 2008), some are suppressed according to a parameter that controls the inhibition level. The results of the modified version of HMAX were obtained using the same parameter for inhibition as for a noiseless dataset. While one could, in principle, tune this parameter to improve the performance for higher levels of noise, this would result in degraded performance for noiseless conditions. This would also contradict the experimental setup used for the Neocognitron (Fukushima, 2011) in which the parameters are tuned in noiseless conditions.

3.2. Extreme noise

We have observed that the noise tolerance is relatively high for moderate levels of noise and have shown how it compares to related models. We will now focus on how this tolerance degrades for extreme levels of noise and how this might offer some insight into the noise tolerance of the C2 responses.

We now consider higher levels of Gaussian and salt & pepper noise, as illustrated in Figure 3. The ETL1 dataset, unlike most handwritten digit

datasets, contains a significant amount of noise originating from the scanning process. We opt for making the patterns binary so that we can have noiseless patterns as a starting point. This enables a clearer analysis of the distortion on the C2 responses caused by adding noise to the input. We convert the patterns to binary using the thrho software by Taiichi Saito, which implements the discriminant criterion (Otsu, 1979). We focus on these two types of noise because they are particularly relevant in engineering and nature. We note that for the higher levels of noise considered, it becomes difficult for humans to recognize the digits. As before, the Gaussian noise intensity is defined by noise (r.m.s) / signal (p-p), which is equivalent the standard deviation in this case. The salt & pepper noise intensity is defined by the percentage of pixels that are randomly set to either black or white.

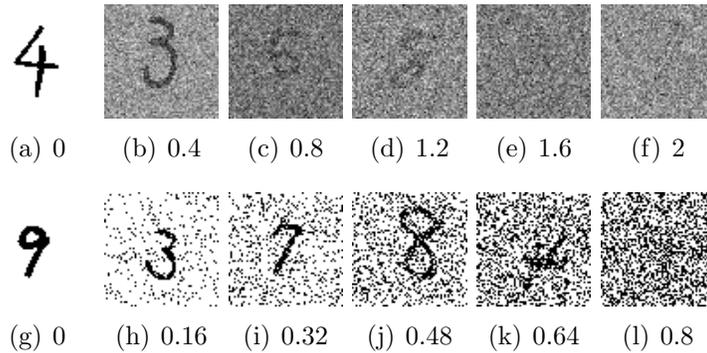


Figure 3: Digits corrupted with noise. Gaussian noise (a-f) and salt & pepper noise (g-i) of varying intensities defined by noise (r.m.s) / signal (p-p) and noise (percentage), respectively. For the white noise, the gray intensity is linearly rescaled for display. The lowest and highest values are represented by black and white, respectively.

We argue that the noise tolerance is closely related to the ability of the model to have similar C2 responses to patterns with and without noise contamination. We show the correlation of the C2 responses under different levels of noise to the noiseless C2 responses in Figure 4. We can see that the correlation in the responses decreases with the noise intensity and that this agrees with the increase in the recognition error (see Figure 6). However, the level of distortion in the responses due to noise is not the only factor; another key aspect, which is also important in the noiseless condition, is that the responses for patterns of different classes should initially be far apart in the Euclidean space.

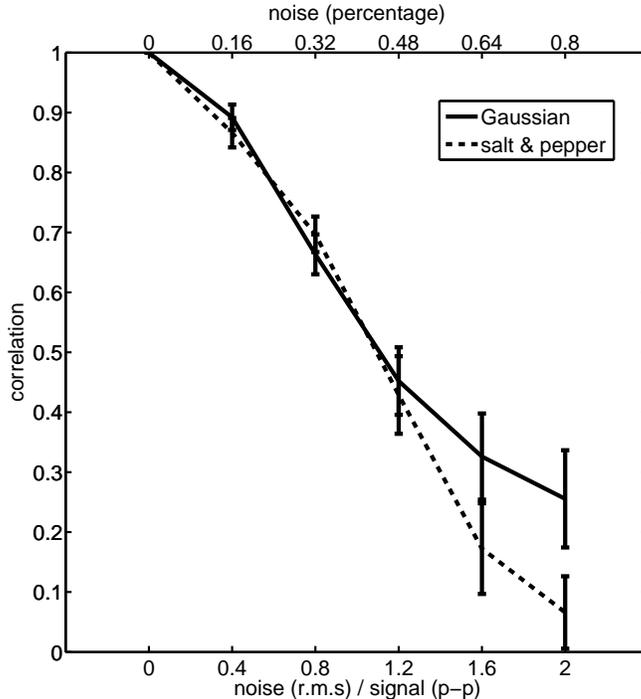


Figure 4: Correlation of C2 responses to original patterns (no noise) with responses to contaminated patterns, averaged over all patterns. Bars represent the average and standard deviation over 5 runs for Gaussian and salt & pepper noise.

We use a classical statistical method to normalize the Euclidean distances between the C2 responses of each pattern: the Z-score (also known as the standard score). The Z-score measures the distance of a given sample from the average in number of standard deviations. The Z-score is useful because it provides a relative comparison between each value and all others. The actual distance values would be much more difficult to interpret because we have no a priori notion of what is a small or large distance between the responses. The Z-score is positive for values above the average and negative for values below the average. The formula that calculates the Z-score from a raw score x is as follows:

$$z(x) = \frac{x - \mu}{\sigma} \quad (1)$$

where μ is the population mean and σ is the population standard deviation.

Henceforth, we will use the Z-score to quantify the distance between the

C2 responses of a pair of patterns and as a measure of their similarity. Ideally, we expect to find that the Z-score of the distance between C2 responses is low (negative) for patterns from the same class and high (positive) for patterns of a different class, meaning that the C2 responses for same class patterns are close in Euclidean space (i.e., similar) and that the responses for patterns of different classes are far apart (i.e., dissimilar). Therefore, we calculate the Z-score of the Euclidean distance of the C2 responses between each all pairs of training and test patterns, i.e. the population is defined as the Euclidean distances between the C2 responses of all training data and test data pairs. We take μ and σ as the average and standard deviation of this population, to then calculate the Z-score $z(x)$ from the raw score x . Then by definition, $z(x)$ represents the number of standard deviations below (negative) or above (positive) the average, regarding the population.

Considering a nearest neighbor classifier, the ideal conditions would be such that patterns from the same class, now represented by C2 responses, are close in the Euclidean space and patterns from different classes are far apart. Otherwise, it would be easier for a distortion in the responses, due to noise in the input, to make the classifier select a sample from the wrong class as its nearest neighbor. With this in mind, we now consider how the average distance between each of the C2 test responses and its closest C2 train response for noiseless conditions is affected by noise, i.e. for each of the test samples we pick the train sample whose Euclidean distance between the C2 responses is smallest without noise. Then we use the Z-score calculated using Euclidean distances between the C2 responses of all training data and test data pairs as previously explained for each particular level of noise, evaluating how the distance of each of the test patterns to the closest pattern of the same class in training (for noiseless conditions) is affected by noise (see Figure 5).

The Z-score of the distance of the C2 responses of a test pattern to the closest pattern of the same class in training is approximately minus four standard deviations for noiseless conditions (see Figure 5). This result implies that for each test pattern, there is a training pattern of the same class that is very close in Euclidean space relative to the average distance, which is consistent with a low recognition error. The average distance between the C2 response of a test pattern and its closest response in the training set from the same class is closely related to the performance of a classifier. We can see in Figure 5 that the Z-score increases with the noise intensity. This means that the test patterns move further from their closest reference training sample (in noiseless conditions) and, as the noise intensity increases, their

distance moves closer to the average (i.e., the Z-score moves closer to 0). This metric is only slightly affected for a moderate level of noise as in the previous experiment (Gaussian=0.4, salt & pepper =0.16); however, it is progressively more affected for higher levels of noise, which is in close agreement with the recognition error of the linear classifier (see Figure 6). The model has a significant noise tolerance until an already high level of noise (Gaussian 0.8, salt & pepper 0.32), but then the performance decreases sharply. The recognition error increases significantly from 0.8 for Gaussian and from 0.32 for salt & pepper noise. For salt & pepper noise, the error is approximately at the chance level when the intensity is 0.64, whereas for Gaussian noise, when the intensity is 2, it still performs slightly below the chance level (see Figure 6).

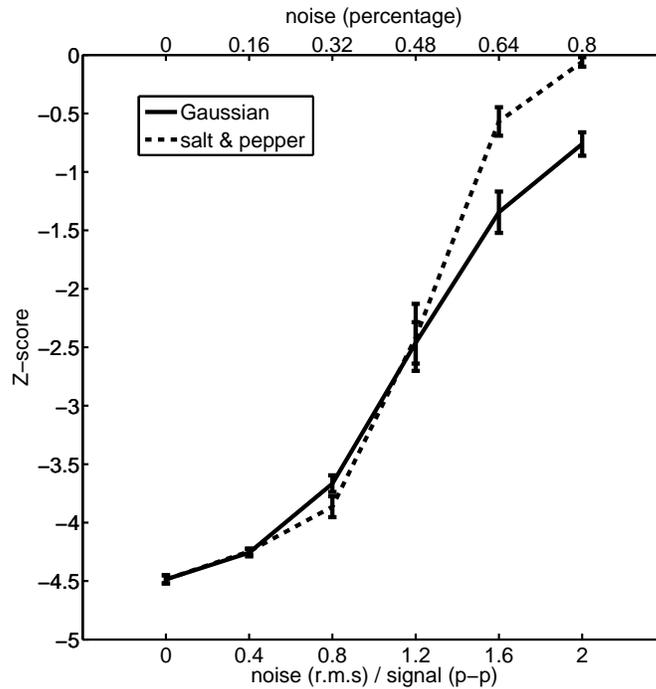


Figure 5: Average Z-score of the Euclidean distance between the C2 responses of each test pattern to the closest pattern of the same class in training (without noise). Z-score calculated relatively to the Euclidean distances between the C2 responses of all train and test data pairs for a given level of noise. The bars represent the average and standard deviation over 5 runs for Gaussian and salt & pepper noise.

Finally, we seek to further improve the noise tolerance by exposing the classifier to the training patterns contaminated with different noise intensities. The rationale is that by including multiple versions of each pattern with different levels of noise, the classifier has prior knowledge of the effects the noise distribution has on the C2 responses. It is not surprising that the exposure to noisy patterns during training improves the noise tolerance. Still, from an application perspective it is relevant how much can the tolerance be improved. The exposure to noise during training has also been reported to improve noise tolerance in a related deep network (Tang & Eliasmith, 2010).

The unsupervised learning in the hierarchical layers is still performed with 3000 training patterns without noise as before. After the unsupervised learning is finished, we now generate multiple versions of the same pattern contaminated with different intensities of the same type of noise. We then use the C2 responses of the original and noisy versions of the train patterns to learn the mapping between the network responses and the class labels as before. The noise is independently sampled for each pixel, pattern, noise level and set. Therefore, for each run, after the training of the hierarchical layers is finished, we expand the C2 responses used to train the classifier by contaminating the original 3000 training patterns with different intensities of either Gaussian noise (0.4, 0.8, 1.2, 1.6 and 2) or salt & pepper noise (0.16, 0.32, 0.48, 0.64 and 0.8), resulting in 18000 C2 responses in each case (five additional for each of the original training patterns) that are then used to train a classifier. We finally evaluate the recognition error as before on a test set contaminated with same type of noise used to expand the training set, but independently sampled.

The recognition error is significantly reduced for high levels of noise by expanding the training set with contaminated versions of the patterns for both Gaussian and salt & pepper noise, as shown in Figure 6. The addition of contaminated versions of the training patterns (indicated by gray lines) significantly reduces the recognition error for higher levels of noise; more notably, it reduces the error from approximately 65% for both Gaussian noise (1.2 intensity) and salt & pepper noise (0.48 intensity) to just 25%. For levels of noise where the performance was either close to or at the chance level (90% error) the recognition is also significantly improved.

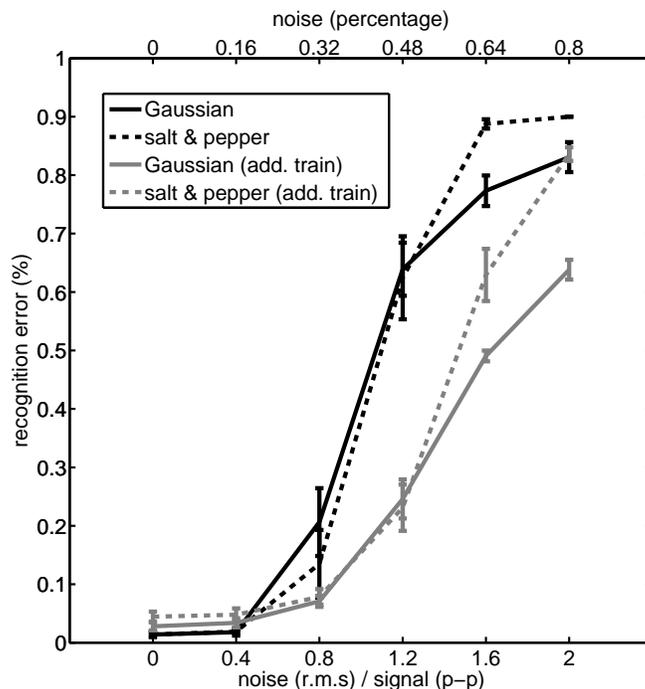


Figure 6: Results from the training set expanded with noisy versions of patterns. The bars represent the average recognition error and standard deviation with 3000 training and 1000 test patterns over five runs for Gaussian and salt & pepper noise. The black lines are the error when training only with clean patterns. The gray lines are the error when extending the training set with a version of each pattern for each level of noise, i.e., adding five additional versions of each pattern to the training set. The noise intensity is clearly illustrated in Figure 3.

4. WTA noise tolerance

The most fundamental difference between the model discussed in this manuscript and related hierarchical models is that the response of simple cells depends on the response of other simple cells. The WTA response described here, in which a single cell responds with a constant value and all others are silent, can be interpreted as an extreme case of divisive normalization (Carandini & Heeger, 2012), i.e., the response of a simple cell is divided by the sum of the responses of other cells after each of the responses has been increased to a sufficiently large power. The WTA response is obviously oversimplified from a neurophysiology perspective, but it appears to capture

the important functional property of noise tolerance. We briefly discuss an example that motivates the difficulty of achieving noise tolerance without depending on the response of other cells with the same receptive field. We will use a simple cell computation that is similar to the one used in the Neocognitron, but the argument for this simplification is more general as detailed below.

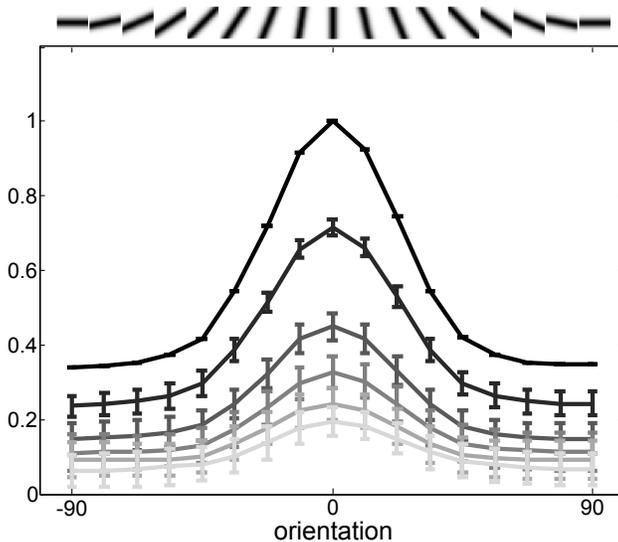


Figure 7: The response of a set of cells whose preferences are oriented lines of different orientations (shown on top) to a vertical line (same as the preference at 0). The horizontal axis measures the orientation distance from the vertical stimulus in degrees. The vertical axis show the average response and standard deviation (over 100 trials) of the cells calculated as the inner product between their preferences and the vertical line stimulus corrupted with additive white Gaussian noise of different intensities defined by noise (r.m.s) / signal (p-p) in the range 0, 0.4, 0.8, . . . , 2. The noiseless condition is represented by black lines, and the strongest noise is represented by the lightest gray lines.

Consider a set of preferred stimuli $\mu_{j \in c}$ where μ_j is the preference of cell w_j and c is a set of cells with the same receptive field. The response of each cell w_j to a stimulus \mathbf{x} is simply given by the inner product of μ_j and \mathbf{x} ; to simplify, we will assume that both \mathbf{x} and μ_j will always have the unit norm. We now consider that each μ_j represents an oriented line split apart by $c/180^\circ$ and pick a stimulus \mathbf{x} as another oriented line with some randomly chosen orientation represented in $w_{j \in c}$. The response will be maximal for the

w_j cell that has the same orientation, and the response will be progressively smaller according to the angle distance between the oriented lines of each of the corresponding μ_j . However, if we now contaminate \mathbf{x} with additive white Gaussian noise, the responses will tend to decrease for the optimally oriented cell, becoming more similar to the responses for non-optimally oriented cells as shown in Figure 7.

In a version of the Neocognitron with increased noise tolerance (Fukushima, 2011), subtractive inhibition suppresses responses below a given threshold. A possible guiding principle for setting this threshold is to choose a value that suppresses spurious responses from non-optimally tuned cells while still allowing optimally tuned cells to continue to respond. Another alternative is to simply find the value that yields optimal performance using some form of optimization. Either way, we are faced with a difficult problem: the dependence between the optimal threshold and the noise intensity. If we set the threshold to low, then we have spurious responses; if we set it too high for some levels of noise, then all cells will become silent. For example, considering the responses shown in Figure 7, if we set the threshold to 0.3, then it would not suppress any responses in the noiseless case, whereas it would unacceptably suppress all responses for higher levels of noise. If we want the model to operate for different levels of noise while suppressing spurious responses, we are faced with a complex problem. Other Neocognitron versions use divisional inhibition and the problem is analogous. Neocognitron divisional inhibition only depends on the cell inputs and not on other simple cell responses. Therefore, it should not be confused with divisive normalization.

We argue that a desirable property to have in a hierarchical model (so that it is tolerant to noise in the input) is that each of the cells responses are individually noise tolerant, i.e., their responses should not be affected by noise. For this property to occur, we argue that the cells response should depend on one another as in divisive normalization (Carandini & Heeger, 2012). The WTA response, by having a maximal and constant response for the optimally tuned cell until the noise perturbation is sufficiently strong to change it, is a reasonable method to achieve this property.

4.1. Bayesian interpretation

We now review a problem under a Bayesian perspective that motivates the noise tolerance of the WTA response, particularly for additive white Gaussian noise. Consider a set of preferred stimuli $w_{j \in c}$; we generate a new stimulus \mathbf{x} by adding white noise to one arbitrarily chosen element w_j with

equal probability. Afterwards, without the knowledge of which w_j we used to generate the stimulus \mathbf{x} , we want to predict w_j while minimizing the error rate. We can relate this problem to a set of simple cells that share a receptive field; however, each of these cells has a different preferred stimulus. If we consider that the first simple layer is in the extracting orientation, this set of cells has to encode the information of which orientation is present in their common receptive field.

The zero-one loss function, which assigns a cost of 0 to correct answers and a cost of 1 to wrong answers, results in a classifier that minimizes the error rate. The Bayesian risk corresponding to this loss function is (Duda et al., 2001):

$$R(\alpha_i|\mathbf{x}) = \sum_{j \neq i} P(w_j|\mathbf{x}) = 1 - P(w_i|\mathbf{x}) \quad (2)$$

where $P(w_i|\mathbf{x})$ is the posterior. Finding the decision α_i that minimizes Equation 2 is equivalent to finding the (Duda et al., 2001)

$$\arg \max_{i \in c} P(w_i|\mathbf{x}), \quad (3)$$

or alternatively, using the posterior definition and the logarithmic version

$$\arg \max_{i \in c} P(\mathbf{x}|w_i)P(w_i) = \arg \max_{i \in c} [\log P(\mathbf{x}|w_i) + \log P(w_i)]. \quad (4)$$

If we now consider the case where $P(\mathbf{x}|w_i)$ is a multivariate Gaussian density function (Duda et al., 2001)

$$P(\mathbf{x}|w_i) = \frac{1}{(2\pi)^{\frac{d}{2}} |\boldsymbol{\Sigma}|^{\frac{1}{2}}} \exp \left[-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_i)^t \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}_i) \right], \quad (5)$$

where $\boldsymbol{\mu}_i$ is the mean of class w_i , $(\mathbf{x} - \boldsymbol{\mu}_i)^t$ is the transpose of $\mathbf{x} - \boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}$ is the covariance matrix common to all classes (in this case representing noise). $|\boldsymbol{\Sigma}|$ is the determinant and $\boldsymbol{\Sigma}^{-1}$ is the inverse. If we consider that the noise is independent of class w_i and is defined by an n -dimensional vector of zero-mean Gaussian random variables with σ^2 variance, then substituting $P(\mathbf{x}|w_i)$ in Equation 4 using Equation 5 and considering that in this case $\boldsymbol{\Sigma}^{-1} = \mathbf{I}/\sigma^2$ (Duda et al., 2001), we obtain

$$\arg \max_{i \in c} [2\sigma^2 \log P(w_i) - (\mathbf{x} - \boldsymbol{\mu}_i)^t (\mathbf{x} - \boldsymbol{\mu}_i)], \quad (6)$$

or alternatively

$$\arg \min_{i \in c} [-2\sigma^2 \log P(w_i) + \|\mathbf{x} - \boldsymbol{\mu}_i\|^2], \quad (7)$$

where $\|\mathbf{x} - \boldsymbol{\mu}_i\|$ is the Euclidean distance between \mathbf{x} and $\boldsymbol{\mu}_i$. Finally, if we now consider that all w_i have equal probability, we simply obtain

$$\arg \min_{i \in c} \|\mathbf{x} - \boldsymbol{\mu}_i\|, \quad (8)$$

which is the WTA operation in simple cells. Therefore, if we would train a classifier on the WTA responses for a single receptive field, we would be using this decision function.

While we related this problem to a set of simple cells with a shared receptive field, we have made a shared set of assumptions in this relation. While additive white Gaussian noise is a general type of noise, other types of distortion can occur. We considered the problem of discriminating which stimulus was present, while we could have considered the problem of estimating the orientation of a stimulus, at least for the first layer. We also considered that all preferences have equal probability, which might not always be the case. Moreover, we are only focusing on which stimulus is present in a receptive field, while the actual problem we would like to solve is what class is present in an entire pattern.

5. Discussion

We have shown the intrinsic high tolerance of a hierarchical network with a WTA response to several types of noise (Cardoso & Wichert, 2010), which is greater than in related models (Fukushima, 2011; Serre et al., 2007; Mutch & Lowe, 2008). We empirically explored the reasons for this tolerance, which can be explained by two factors: the ability of the model to have similar responses for patterns contaminated with different intensities of noise and the resilience to noise of the low similarity of the responses for patterns of different classes. We reported that further noise tolerance can be achieved by training the classifier with additional network responses for the same patterns contaminated with different levels of noise. Finally, we discussed the relation between the WTA response and Bayesian discrimination under the condition of additive white Gaussian noise. We will most likely require a different computation for when the stimuli are not closely related to the

preferences or for other types of noise. We argue that divisive normalization, of which the WTA response is a particular and extreme case, is an important computation for noise tolerance in hierarchical models.

Layer	Param.	ETL1	MNIST
S1	size	8	4
	shift	1	1
	frame	4	0
	classes	16	16
C1	size	4	3
	shift	2	1
	frame	3	0
S2	size	5	5
	shift	1	1
	frame	2	0
	classes	100	100
C2	size	12	6
	shift	2	1
	frame	0	0

Table 3: Parameters used in the experiments for each of the datasets.

Acknowledgements. The authors would like to thank João Sacramento for helpful comments and reviewing this manuscript. This work was supported by national funds through FCT – Fundação para a Ciência e Tecnologia, under project PEst-OE/EEI/LA0021/2013 and through an individual doctoral grant awarded to the first author (contract SFRH/BD/61513/2009).

Carandini, M., & Heeger, D. J. (2012). Normalization as a canonical neural computation. *Nat. Rev. Neurosci.*, *13*, 51 – 62.

Cardoso, Â., & Wichert, A. (2010). Neocognitron and the Map Transformation Cascade. *Neural Networks*, *23*, 74 – 88.

Cardoso, Â., & Wichert, A. (2013). Handwritten digit recognition using biologically inspired features. *Neurocomputing*, *99*, 575 – 580.

Duda, R., Hart, P., & Stork, D. (2001). *Pattern classification*. John Wiley & Sons.

- Fukushima, K. (1980). Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, *36*, 193–202.
- Fukushima, K. (2003). Neocognitron for handwritten digit recognition. *Neurocomputing*, *51*, 161–180.
- Fukushima, K. (2010). Neocognitron trained with winner-kill-loser rule. *Neural Networks*, *23*, 926 – 938.
- Fukushima, K. (2011). Increasing robustness against background noise: Visual pattern recognition by a neocognitron. *Neural Networks*, *24*, 767 – 778.
- Hamidi, M., & Borji, A. (2010). Invariance analysis of modified c2 features: case studyhandwritten digit recognition. *Machine Vision and Applications*, *21*, 969–979.
- Mutch, J., & Lowe, D. G. (2008). Object class recognition and localization using sparse features with limited receptive fields. *International Journal of Computer Vision*, *80*, 45 – 57.
- Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, *9*, 62 – 66.
- Riesenhuber, M., & Poggio, T. (1999). Hierarchical models of object recognition in cortex. *Nature Neuroscience*, *2*, 1019 – 1025.
- Serre, T., Oliva, A., & Poggio, T. (2007). A feedforward architecture accounts for rapid categorization. *Proceedings of the National Academy of Sciences USA*, *104*, 6424.
- Tang, Y., & Eliasmith, C. (2010). Deep networks for robust visual recognition. In *Proceedings of the 27th International Conference on Machine Learning*