ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 297-302

On the Robust Design of Uncoupled CNNs

BAHRAM MIRZAI DRAHOSLAV LÍM GEORGE S. MOSCHYTZ
Signal and Information Processing Laboratory, ETH Zurich, Switzerland
mirzai@isi.ee.ethz.ch, drahoslav.lim@isi.ee.ethz.ch

ABSTRACT: *A design procedure for uncoupled CNN templates, based on input/output mappings, is presented. A table of mapping types, parameter inequalities and error tolerances is provided. The minimum relative accuracy of the parameters required in order to guarantee correct operation is given in closed form. We demonstrate the design on several examples.*

1. Introduction

Cellular neural networks [1] (CNNs) constitute a class of locally connected nonlinear dynamic systems which can be realized as analog VLSI circuits. Recent implementation literature includes [2–4]. The basic equation underlying CNN dynamics is

$$\frac{dx_{ij}}{dt} = -x_{ij} + \sum_{kl\in\mathcal{N}_{ij}} a_{ij,kl}\, f(x_{kl}) + \sum_{kl\in\mathcal{N}_{ij}} b_{ij,kl}\, u_{kl} + I\,. \qquad (1)$$

The nonlinearity $f(\cdot)$ is the piecewise-linear function

$$f(x) = \frac{1}{2}\{|x+1| - |x-1|\}\,.$$

$\mathcal{N}_{ij}$ denotes the set of all cells with which the ij-th cell is connected. The network parameters for spatially invariant CNNs are given in a *template set* or simply *template,* consisting of the feedback matrix A and the control matrix B, comprised of $\{a_{ij}\}$ and $\{b_{ij}\}$, respectively, and the overall network bias I. The CNN inputs and outputs are defined to be u_{ij} and $f(x_{ij})$, respectively. Generally, a CNN operates by choosing a template set A, B, I, and appropriately assigning some initial data to u_{ij} and $x_{ij}(0)$. A desired output is then obtained as a stable equilibrium point of (1).

The processing time, i.e., the time required to reach the equilibrium point, is limited only by parasitic capacitances of the analog circuit realization. On the other hand, a compact analog circuit will realize each parameter with only a limited degree of accuracy, on the order of 10% [2, 4].

The nature of some CNN processing tasks requires that each cell interact with the cells in its neighborhood in a static way, meaning that the output of a particular cell depends only on its input and on that of its neighbor cells (feedforward), rather than requiring feedback from neighbor cells during processing. This class of processing tasks can be described by static input/output mappings and implemented by *uncoupled* CNNs. In an uncoupled CNN, each cell is allowed to have at most self-feedback. An extended formulation of this condition in terms of *Boolean function characterization* can be found in [5].

In this tutorial we describe simple design rules for uncoupled CNNs. We associate with each rule a measure that quantifies the robustness degree of that particular rule. Emphasis will be placed on implementing tasks by a single template.

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 297-302

However, this may imply high sensitivity, particularly—as will be seen below—for tasks requiring a large degree of connectivity. Such accuracy may not be convenient or possible to implement in an analog CNN circuit, where any increase in accuracy implies more complex circuitry, more chip area and increased power consumption.

At the cost of a sensitivity–complexity tradeoff, high connectivity processing tasks can be realized in a highly robust way by decomposing the task into simpler CNN subtasks and combining the partial results with suitable logical operations [6]. This requires a programmable CNN, and the CNN circuit must be embedded in a computational device [7] capable of performing combinational logic operations.

Throughout this paper, it is assumed that the inputs and outputs are bipolar (± 1), and that the processing tasks in question do not require feedback among cells. The procedures herein are not applicable to tasks which do require feedback among cells (coupled CNNs).

2. Need for Robust Template Design

The template values, i.e., the entries of A, B and the value of I, can only be implemented or programmed with finite accuracy. In particular, this restriction must be kept in mind if the CNN processing is to be performed by an analog CNN circuit, where accuracy is necessarily limited [2]. Various other error sources are present in an analog CNN circuit. We cannot assume that the input values u_{ij} are highly precise, that the initial states $x_{ij}(0)$ are set very precisely or even that the output function saturation level [1] is exactly ± 1. These non-idealities can be considered to be error sources which occur in addition to template value errors. It is analytically and conceptually convenient to keep the form of the errors as $a_{ij} + \delta a_{ij}$, $b_{ij} + \delta b_{ij}$ and $I + \delta I$, where a_{ij} and b_{ij} are the entries of the A and B templates, and I is the bias of the CNN template, respectively, and to increase the error terms $\delta\cdot$ to account for other errors.

It is straightforward to account for u_{ij} errors: the contribution in (1) due to a B template entry b is $\pm|b|$ if everything were precise. Due to errors in u_{ij} and in b the current magnitude will actually be $I_b = (b+\delta b)(u+\delta u) = b+\delta b(1+\delta u)+b\delta u$. In considering a template's robustness one should use the error term $\delta b(1+\delta u)+b\delta u$ in place of the "template error" of b, in other words increase the relative error from just $\frac{\delta b}{b}$ to $(\delta b(1+\delta u)+b\delta u)/b$.

Accounting for initial state or output saturation errors is not as straightforward. In the case of uncoupled templates and for typical error values these error sources usually do not need to be treated analytically [6].

A significant factor in considering the robustness of a particular template is the *degree of connectivity* of a task, which in the case of uncoupled CNNs means the number of non-zero entries in the B matrix. A higher degree of connectivity necessarily implies less robustness.

2.1. Robust Design of Uncoupled Templates

The approach used here is based on input/output pairs and input configurations to-be-recognized, and expressing these in terms of constraints on the values of the template parameters. We use the convention that a black pixel, shown as $\bullet$, has the value $+1$, and a white pixel, $\circ$, has the value -1. The boundary cells are assumed to be set to white (-1).

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 297-302

Input		**Output**	**Type**	**Condition**	**error term ϵ**	
· · · · • · · · ·	→	•	bistable	$w_{ij} > \ 1-a_c$	$w+a_c-1$	(1a)
· · · · ∘ · · · ·	→	∘	bistable	$w_{ij} < -1+a_c$	$-w-1+a_c$	(1b)
· · ·† · • · · · ·	→	•	bistable	$w_{ij}\|_{b_c=0} = \ 0$	$-\|b_c\|+a_c-1$	(1c)
· · ·† · ∘ · · · ·	→	∘	bistable			
· · · · • · · · ·	→	∘	monostable	$w_{ij} < \ 1-a_c$	$-w+1-a_c$	(2a)
· · · · ∘ · · · ·	→	•	monostable	$w_{ij} > -1+a_c$	$w+1-a_c$	(2b)

Table 1: Uncoupled CNN trajectory types. Input cells denoted by · are given by the prescribed mapping. In each case, w_{ij} has the value given by (2) for the particular input configuration. Except for the center cell, the two configurations †, case (1c), must be identical.

A template is said to *recognize* or *detect* a particular *input configuration* (input pattern) consisting of black, white and "don't care" cells (•, ∘ and $*$, respectively), if the output *for that particular input configuration* is black. The opposite output is produced for all other input configurations.

Consider an uncoupled CNN with neighborhood $\mathcal{N}_{ij}$ described by

$$\dot{x}_{ij} = -x_{ij} + a_c \operatorname{sat}(x_{ij}) + w_{ij}\ , \quad w_{ij} = \sum_{mn \in \mathcal{N}_{ij}} b_{mn} u_{mn} + I\ , \tag{2}$$

and assume that the initial states $x_{ij}(0)$ and the inputs u_{ij} are both set to the input image. From the phase diagram [5] corresponding to (2), the design rules given in Tab. 1 can be obtained. These rules correspond to $a_c > 1$. Two common cases are distinguished, *monostable* and *bistable*. Other less often used cases are listed in [5]. In the monostable case the phase diagram has only one equilibrium, whereas in the bistable case, Fig. 1, there are three equilibrium points, two stable ones separated by an unstable one. A *robust design* is now obtained by choosing

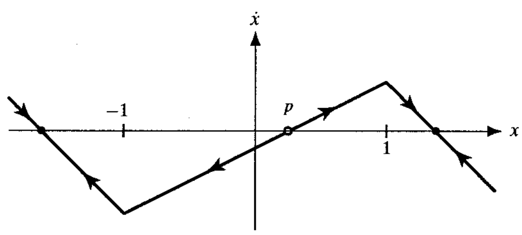

Figure 1: Phase plot, bistable CNN, $|w_{ij}| < a_c - 1$

the nominal values of the A and B templates and of I such that the corresponding inequalities in Tab. 1 hold, even if the templates are subjected to some prescribed

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 297-302

error. In the worst case the error term is

$$\Delta = \sum_{mn} |\delta b_{mn}| + |\delta a_c| + |\delta I| \ , \tag{3}$$

where the indices mn are taken over all non-zero terms in the B template. This error term should be included in the inequalities so as to make the particular condition a stronger upper or lower bound. For example, if $w_{ij} > 1 - a_c$, Tab. 1, row (1a), is to still hold with template errors, then the nominal values have to satisfy the inequality

$$w_{ij} - \Delta > 1 - a_c \ . \tag{4}$$

Relations like (4) can be understood in two ways. First, for a *given* Δ, eq. (4) provides a constraint on the nominal template values, which in this case is $w_{ij} + a_c > 1 + \Delta$. Second, for an uncoupled template whose mapping requires the constraint $w_{ij} > 1 - a_c$, eq. (4) sets a bound on the *maximum possible* Δ, namely $\Delta < \epsilon = w_{ij} + a_c - 1$. The expression for ϵ for each phase plot type is included in Tab. 1.

The case Tab. 1, row (1c) needs to be considered separately. This is the case when both a black and a white cell with *identical neighborhoods* are to be mapped to the same respective output color. If the nature of the task (the other mapping constraints) permits the center B matrix element b_c to be zero, w_{ij} does not depend on the center cell's input. Tab. 1, rows (1a) and (1b) clearly imply that the optimal choice is to set the intermediate equilibrium point p in Fig. 1 so as to obtain $p = 0$, i.e., $w_{ij} = 0$. The other mapping constraints may, however, require that $b_c \neq 0$. In that case the optimum is to "center" the w_{ij} values for the black and white center cell around zero, i.e., require that $w_{ij}\big|_{b_c=0} = 0$, and then obtain b_c from the other mapping constraints.

Assume now that the template entries are subjected to relative errors of α, i.e., $|\delta a_c| < \alpha |a_c|$, $|\delta b_{mn}| < \alpha |b_{mn}|$ and $|\delta I| < \alpha |I|$. It follows from (3) and relations of the type (4) that for each input configuration, i.e., for each w_{ij} or ϵ, we can write

$$\alpha_\epsilon < \frac{\epsilon}{\sum_{mn} |b_{mn}| + |a_c| + |I|} \ . \tag{5}$$

The inequality (5) gives the minimum relative accuracy required for each input configuration, and has an important implication: *The higher a task's connectivity*—which in the case of an uncoupled template means the more non-zero terms there are in the B template—*the more sensitive the task becomes.* The higher sensitivity arises from two related factors. *i*) The number of non-zero B entries, and *ii*) the larger absolute values of the B entries. If the relative template error can be guaranteed to be smaller than $\min \alpha_\epsilon$, where the minimum is taken over all input configurations relevant to the task, then the template is guaranteed to operate correctly.

3. Examples

In this section we introduce several CNN templates as examples of the design procedure for uncoupled CNN templates. Special attention will be paid to the templates' robustness degree.

Horizontally Isolated Point Detection. The templates of this type recognize points isolated with respect to the horizontal direction. This template performs the mapping that the input configuration $\circ \bullet \circ$ leads to a black output ($\bullet$) and all other

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 297-302

input configuration to white (∘). This can be accomplished by assuming the template set

$$A = [0 \ a \ 0] \qquad B = [-c \ 0 \ -c] \qquad I = i\ , \tag{6}$$

and applying the rules of Tab. 1. The initial configurations ∘•∘ and ∘∘∘ are mapped correctly if $w_{ij} = 0$, i.e., $i = -2c$, Tab. 1, row (1c). Furthermore, for the configurations ••∘ and ∘•• we should satisfy $i < 1 - a$, Tab. 1, row (2a). The constraints for the configurations ••• and •∘• are $-4c = 2i < 1 - a$ (Tab. 1, row (2a)) and $-4c = 2i < -1 + a$ (Tab. 1, row (1b)), respectively. The constraints of the configurations •∘∘ and ∘∘• are subsumed in the previous ones. From the constraints, the following values can be obtained.

$$A = [0 \ 2 \ 0] \quad B = [-1 \ 0 \ -1] \quad I = -2\ . \tag{7}$$

By (5) these values have a relative robustness of 16.6%.

It is apparent that the two off-center white cells could be in any position within the CNN neighborhood, so isolation along other "directions" can be found. Furthermore, changing the sign of the off-center B template entries will detect the presence of black neighbors, rather than white neighbors. In general, any input configuration having a black center cell, two off-center black or white cells, and 6 off-center "don't care" cells can be recognized by a template similar to (7).

Line Detection. The task of "line detection" finds groups of two or more pixels which are positioned so as to form a "line" along a direction on the cell grid. Here we consider *horizontal line detection* (HLD); by rotating the templates, lines along 90° or 45° directions can be detected as well.

HLD can be defined as recognizing the input configurations •••, ∘•• and ••∘. Usually this recognition is performed by coupled templates [8, 9]. Here we design an *uncoupled HLD template.*

By the nature of the task we assume the following template:

$$A = [a] \qquad B = [c \quad b \quad c] \qquad I = i\ .$$

For the input configuration ∘•• we get by Tab. 1, row (1a) the relation $b + i > 1 - a$. For the input configuration •∘• we get by Tab. 1, row (1b) the relation $2c - b + i < -1 + a$. Finally, for the input configuration ∘•∘ we get by Tab. 1, row (2a) the relation $-2c + b + i < 1 - a$. It can be shown that the constraints for the remaining configurations •••, ∘∘∘ and •∘∘ are already subsumed in the above inequalities. The constraints simplify to

$$a > 1 \quad c > 0 \quad i < 0 \quad b_{\min} < b < 1 - a - i + 2c\ , \tag{8}$$

with $b_{\min} = \max\{1 - a + c, 1 - a - i, 1 - a + i + 2c\}$. A possible uncoupled HLD template is therefore

$$A = [2], \qquad B = [1 \quad 1 \quad 1], \qquad I = -1\ . \tag{9}$$

This template can, by (5), tolerate relative errors of about 16.6%, as compared to 3.3% for similar templates published elsewhere [8].

Edge Extraction with Diagonals We consider a black cell with exactly eight black neighbors an interior cell; all other black cells are edge cells. Therefore, we seek to recognize black cells with *at least* one white neighbor. The following template set is assumed for the design:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} c & c & c \\ c & b & c \\ c & c & c \end{bmatrix} \qquad I = i\ . \tag{10}$$

ESANN'1998 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 22-23-24 April 1998, D-Facto public., ISBN 2-9600049-8-1, pp. 297-302

The design procedure is similar to the above examples and the defining inequalities turn out to be

$$a > 1, \quad c < 0, \quad i < 0, \quad b_{\min} < b < 1 - a - 8c - i\,, \tag{11}$$

with $b_{\min} = \max\{1 - a - 7c, 1 - a - 6c - i, 1 - a - 8c + i\}$. By (5), the template of (11), with $a = 3$, $b = 6$, $c = -1$ and $i = -1$, admits a relative error of only 5.5%.

Such accuracy is impractical to reliably implement in analog circuits and would require area-consuming circuitry to achieve. Single template edge extraction including diagonal cells is therefore a processing task not well suited for analog hardware. Templates for tasks requiring a high degree of connectivity may not be sufficiently robust to be useable on analog CNN hardware, and a method of decomposing the templates into low-connectivity, low-sensitivity sub-templates and logical operations can be used [6].

4. Conclusions

We have presented exact design rules applicable to uncoupled CNNs. The class of uncoupled CNNs applies only to those tasks which can be described in terms of static input/output mappings. The design rules are based on different characteristics that the phase plot of an uncoupled CNN can have. A measure of robustness for the set of rules implementing a specific task is given. In particular, this measure implies that tasks requiring a high degree of connectivity may turn out to be too sensitive to be useable in an analog CNN circuit. Several examples of the use of the design rules are presented: horizontally isolated point detection, line detection and edge extraction. The template values obtained reflect the connectivity–sensitivity dependence.

References

[1] Leon O. Chua and Tamás Roska, "The CNN Paradigm," *IEEE Transactions on Circuits and Systems–I*, vol. 40, no. 3, pp. 147–156, Mar. 1993.

[2] Peter Kinget and Michiel Steyaert, *Analog VLSI integration of massive parallel signal processing systems*, Kluwer, Dordrecht, The Netherlainds, 1997, ISBN 0-7923-9823-8.

[3] Ari Paasio, Adam Dawidziuk, Kari Halonen, and Veikko Porra, "Minimum size 0.5 micron CMOS programmable 48 by 48 CNN test chip," in *European Conference on Circuit Theory and Design*, Budapest, Sept. 1997, vol. 1, pp. 154–156.

[4] Drahoslav Lím and George S. Moschytz, "A Modular gm-C Programmable CNN Implementation," Sept. 1997, *submitted for publication*.

[5] Leon O. Chua and Tamás Roska, "Cellular Neural Networks: Foundations and Primer," Lecture Notes for the course EE129 at U. C. Berkeley, Mar. 1997.

[6] Drahoslav Lím, Bahram Mirzai, and George S. Moschytz, "Robust CNN Algorithms for High-Connectivity Tasks," Report 9703, ISI, ETH, CH–8092 Zurich, Switzerland, Nov. 1997.

[7] Tamás Roska and Leon O. Chua, "The CNN Universal Machine: An Analogic Array Computer," *IEEE Transactions on Circuits and Systems–II*, vol. 40, no. 3, pp. 163–173, Mar. 1993.

[8] Tamás Roska and L. Kék, "Analogic CNN Program Library, vers. 6.1," Tech. Rep. DNS-5-1994, Dual & Neural Computing Systems Laboratory, Computer and Automation Research Institute, Hungarian Academy of Sciences, Dec. 1994.

[9] Bahram Mirzai, Drahoslav Lím, and George S. Moschytz, "Robust CNN Templates: Theory and Simulations," in *IEEE International Workshop on Cellular Neural Networks and their Applications*, Seville, June 1996, pp. 393–398.