IMAGE HALFTONING WITH CELLULAR NEURAL NETWORKS

by

K. R. Crounse, T. Roska, and L. O. Chua

Memorandum No. UCB/ERL M91/106

15 November 1991

IMAGE HALFTONING WITH CELLULAR NEURAL NETWORKS

by

K. R. Crounse, T. Roska, and L. O. Chua

Memorandum No. UCB/ERL M91/106

15 November 1991

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

Image Halftoning with Cellular Neural Networks *

K. R. Crounse† T. Roska‡
L. O. Chua

Electronics Research Laboratory
Dept. of Electrical Engineering and Computer Science
University of California at Berkeley
Berkeley, CA 94720

November 15, 1991

Abstract

Many algorithms exist for the halftoning of digital images. These algorithms all suffer well studied defects, which are especially apparent in the case where the resulting image is to be displayed at the marginally sampled resolution and viewed at the critical pixel merge distance.

Recently, it has been shown that neural network approach may be useful for halftoning [KA91]. Here, the feasibility of using neural networks in a practical application is considered. The Cellular Neural Network (CNN) architecture is chosen for its proven implementibility in VLSI, high speed, and programmability [CC91b]. Since both the CNN and halftoning have a geometrically local character, the CNN provides a natural implementation. The CNN template weights are derived by analogy to the well known error diffusion algorithm for halftoning. Some limitations of the neural network approach are analyzed providing an advance in designing template weights over previous methods. These limitations are shown to be especially critical in the case of the small interconnection neighborhoods needed for efficient implementation.

Our design criteria are validated by direct simulation. The resulting halftones are shown to be more faithful reproductions of the original than those produced by the error diffusion algorithm. It is suggested that a CNN with optical inputs could provide a high-speed scanner/halftoner for applications such as FAX.

0 Introduction

Cellular Neural Networks (CNNs) are locally connected analog nonlinear processing arrays. The simple processing elements are placed on a 3-D grid (in 2-D layers) [CY88b, CY88a]. The first tested working VLSI chip has a very high speed: 0.3×10^{12} operations/second on $1cm^2$. Using nonlinear and delay-type template elements ('synapses') [RC90] a very broad class of dynamic nonlinear spatial convolution operators can be implemented.

*This work is supported in part by the National Science Foundation under Grant INT 90-01336 in cooperation with the Hungarian Academy of Sciences, the Office of Naval Research under Grant N00014-89-J-1402 and by the National Science Foundation under Grant MIP 86-14000.

†Internet: crounse@fred.berkeley.edu

‡visiting scholar from the Hungarian Academy of Sciences, Uri-u. 49, Budapest, H-1014, Hungary

Halftoning is a local operation on grey-scale images resulting in binary images giving a similar impression to the viewer. Due to this locality, CNN is an ideal 'one-to-one' device for halftoning.

In this paper different halftoning CNN templates are developed. Their main geometrical and circuit properties are studied. Rigorous analytical results and experimental tests are presented.

In Section 1, the background information on halftoning as well as preliminary results and the CNN framework are presented. Section 2 contains the main results: the development of the CNN templates, their properties, and some limitations. Template design examples and experimental verification are shown in Section 3. The application potential is discussed in Section 4. In the conclusion it is explained that the method presented here can be used in other types of problems as well.

1 Background and Preliminaries

1.1 Halftoning

Image halftoning is the process of converting a grey-toned image to a binary-valued image. This process is required in many applications where the display medium can only support binary output. For instance, photographic halftoning techniques have long been used in newspaper printing where the resulting binary values represent the presence or absence of black ink. Upon display it is hoped that, by the blurring of the eye, the halftone image will appear similar to the original continuous toned image. Digital image halftones are required in many present day electronic applications such as facsimile (FAX), electronic scanner/copying, laser printing, and low bandwidth remote sensing.

The framework in which halftoning is usually embedded is depicted in Figure 1. A digital image is assumed to be generated by windowing a real-world scene which contains unlimited detail followed by spatial low-pass filtering and sampling. The image is low-pass filtered to half the sampling rate to prevent aliasing. Sampling takes place on a regular square grid. Each sample may be quantized to a finite representation of the light intensity at that location. At this point, we have as much of the original information that can ever be expected to be recovered by ideal reconstruction and viewing. Halftoning such an image will further quantize the intensity representation at each pixel to one of two possible values. In some cases it may be possible to increase the sampling rate before halftoning for a higher resolution display. The display device then marks the halftoned image on the binary medium producing a physical reconstruction of the image. Finally, the halftoned and displayed image is viewed at the proper distance and should give the illusion of a continuous toned image.

In general halftoning with finite resolution results in the loss of some information which it may replace with artificial information. A good halftone will retain as much of the relevant information as possible, and in that sense can be considered an image compression technique. How one defines the relevant information depends on the application, and can be formulated in terms of a cost between the display and view operators ($\mathcal{A}$) applied to the halftone ($\mathbf{y}$) and the ideal reconstruction and view ($\mathcal{B}$) applied to the sampled input image ($\mathbf{u}$).

A key aspect of our framework is the block representing the processing done by the human visual system. In [MS74] a compressive nonlinearity[1] followed by a linear filter is used as

[1]The standard model is that perceived brightness increases only logarithmically with intensity, but experimental evidence in [MS74] suggests that $u^{0.33}$, where u is the input intensity, may be better.

a model for human perception. The form of the equations was assumed *a priori* and the optimal coefficients were chosen by human subjects. The resulting model is demonstrated to agree closely with other experimental evidence.

The linear portion in the frequency domain, called the Modulation Transfer Function (MTF), is shown along one radial frequency direction in Figure 2. It can be expressed by

$$MTF(f_r) = 2.6(0.0192 + 0.114 f_r)e^{-(0.114 f_r)^{(1.1)}} \tag{1}$$

which describes a radial bandpass characteristic which rises quickly to peak at about 8 cycles/degree and then drops of linearly until an attenuation of 1/10 at 35 cycles/degree which continues exponentially to zero[2]. However, it is suggested in [BKT89] that the transfer function near zero for low frequency is actually an artifact of experimental techniques.

In addition, most linear models make compensation for the anisotropy in the visual system. [KA91] suggests the use of a L_1 measure of distance from the frequency origin, i.e. $f_r = |f_1| + |f_2|$ resulting in lines of constant magnitude along square diamonds. [SRM91] proposes complex adjustments which attenuate the diagonal frequencies even further.

Note that the frequency axis in Figure 2 is given in cycles/degree. It can be seen that C cycles/degree corresponds to a frequency of F cycles/mm in an image viewed at R mm if

$$FR\frac{\pi}{180} = C$$

for small subtended angles. As an example, a common FAX sampling rate is 8 samples/mm. If the output is viewed at a half meter, then the sampling rate is about 70 cycles/degree. This image is considered near critically viewed since the highest frequency preserved in the original image, 35 cycles/degree is near the largest frequencies the eye can detect. If it is viewed closer, the eye reconstruction will begin aliasing. If viewed farther away, some of the information in the image will be lost by the filtering of the eye.

To demonstrate these principles, see Figure 3, the Arden Chart. The highest frequency in the image will correspond to 70 cycles/degree if the chart is observed at 2.8 meters. The Arden chart increases in frequency from left to right and decreases in contrast from top to bottom. If the chart is covered and slowly exposed from the bottom, you should be able to determine the first point when each frequency becomes visible. The resulting graph should give a rough idea of your MTF. This method ignores the anisotropy of the human eye.

The display function block represents the technique by which the image is reconstructed for viewing. For instance, on a printer, a zero-order hold can be approximated by marking black squares on a white page. But, this is an over simplification. If you look at a black box made by a laser printer it will appear as more of a splattering of toner particles which may overlap with neighboring boxes. This effect is not symmetrical; black markings may spill over into unmarked white regions, but not the reverse, causing a general darkening of the displayed image. For a video monitor, the output intensity produced by a pixel can be approximately modeled as obeying a spatial Gaussian function. In general, the display function of a device is nonlinear and stochastic. For a detailed discussion of these and other output devices, see [Uli90].

Despite the nonlinearities and other complexities of the display and viewing process, throughout this paper we will be so bold as to model $\mathcal{A}$ in the halftone path and $\mathcal{B}$ in the ideal

[2]Different parameters are used in [SRM91] resulting in a slightly different transfer function.

path by linear filters. In addition, these linear filters will be discrete space approximations to the continuous space processes[3].

Many deficiencies of halftoning algorithms have been studied. Stated in the positive, there are many possible criteria for the evaluation of a given halftoning technique. They can be divided into categories, which are not necessarily mutually exclusive, as follows:

effective quantization The interval of possible input grey levels can be partitioned into equivalence classes according to the output produced by the halftone algorithm on constant input images. The effective quantization is the number of these equivalence classes. If this is too small, artificial contouring may be introduced in slowly varying regions of the image. If there are too many partitions, objectionable microstructure could result. For instance, an algorithm that attempts to emulate a grey level very near black by putting one white dot on a black page is not necessarily doing the correct thing. In addition, there is no need to generate output accuracy beyond that of the input or what is perceptible by the human eye.

representation linearity The lengths of the partitions of the input grey level interval may vary. This may be desirable, especially if the effective quantization is small. For instance, the human eye is logarithmically sensitive to intensity. Therefore, it may be wiser to use more of the equivalence classes to represent darker grey values where small changes have a greater impact on perception.

average density Each equivalence class has an average density associated with it. This is the average component of the output image produced by this class. In the case where the partitions are connected intervals it is considered desirable that the average density be equal to the centroid of the partition. However, it is common in image processing to apply a nonlinearity to adjust the tone scale to provide some 'snap' or 'graphic punch' to account for the reproduction technique [Uli90]. In this case, the average density produced by halftoning could be modified to provide this effect.

edges A halftoning algorithm must not only preserve the presence an edge, but also its location. In the worst case, an edge could be completely obliterated. More likely, the edge will be blurred so that its position is no longer exact. Other algorithms may actually enhance edges along some or all directions. The desirability of this feature would depend on the intended use. Another possibility is the introduction of edges that did not exist in the original. This is known as artificial contouring.

microscopic structure The preservation of small structures can be measured for varying levels of contrast against the object's background. This is an especially useful measure in the case that text is present in the image. More important, some algorithms introduce annoying and misleading microstructure. This can take the form of meandering structure termed 'worms' or 'squiggles'.

blue noise spectrum In [Uli90] it is found desirable that the output of a constant grey level input image have a 'blue noise' frequency spectrum. For each possible input intensity a principal wavelength is defined as the expected distance between output pixels of the same value. A "Well Formed Dither Pattern" for a constant input should have a frequency spectrum which has the proper DC component, a region with no

[3]One could argue that the retina actually performs discrete space filtering, albeit on a much finer scale than we will be modeling.

energy, a peak at the principal frequency, followed by a level region of noise. That is, the output spectrum will have all its noise energy in the high frequencies above the principal frequency. According to Ulichney, "Blue noise patterns enjoy the benefits of aperiodic, uncorrelated structure without low frequency graininess." [Uli90, page 233]

frequency weighted mean square error It is proposed in [Ana89] and [SRM91] that the mean square error (MSE) between the perceived input and output images is a relevant measure of the performance. This has the advantage of reducing the evaluation of a particular halftone algorithm on a given image to a single quantitative value.

isotropy Isotropy is the ability of an algorithm to perform the same invariant of direction. Isotropy in the frequency domain of output images is deemed very important by [Uli90]. Others [SRM91, especially] have found some anisotropy beneficial as the human visual system is less sensitive to high frequencies along the diagonal directions.

From the framework described in Figure 1 it can be understood that one should not expect a single halftoning algorithm to be universal. In particular, its performance will depend on the characteristics of the display device, the distance of viewing, and the interpretation of relevant information used in the application. The most challenging halftoning problems occur when the image has been sampled and displayed at or slightly above the Nyquist rate and is being viewed closer than the critical merge distance. When the image has been oversampled, there is lots of room for halftone noise in the unused high frequencies. When viewed from too great a distance, the high frequency information in the original image would be unseen and can be replaced by halftoning error. In either case, introduced microstructure would not be visibly distracting. In practice, these conditions are not often met and the problem, which is addressed here, becomes the placement of the error introduced by halftoning in the least objectionable way. As Ulichney states, "This is a study of controlled noise" [Uli90, page 2].

If the image has been severely oversampled or can be displayed with greatly increased resolution, simple halftoning techniques such as ordered dither suffice. However, in many applications the image may only be marginally oversampled. In this case, simple techniques may obscure edges and destroy fine detail such as text. The more advanced error diffusion algorithm can handle this situation but have been shown to introduce distracting patterns because of the directional nature of the process.

1.2 Mathematical Preliminaries

To formalize the current discussion, the following definitions are introduced.

Definition 1 The *support* of an $M \times N$ image, $\mathcal{S}_{M\times N} \subset \mathcal{Z}^2$, is a collection of ordered pairs such that $\mathcal{S}_{M\times N} = \{(i,j) : i \in \{0,1,...,M-1\}, j \in \{0,1,...,N-1\}\}$. The elements of $\mathcal{S}_{M\times N}$ are called *pixels*, or sometimes *cells*.

Definition 2 An $M \times N$ *digital image* is a mapping from some finite support $\mathcal{S}_{M\times N}$ to a set of values which represent reflectivity. For our purposes we make the useful definitions:

$$\text{grey-scale image: } S_{M\times N} \rightarrow [-1,1] \subset \Re$$

$$\text{binary image: } S_{M\times N} \rightarrow \{-1,1\} \subset \mathcal{Z}$$

Of course other representation sets may be used. For instance, in reality a digital image will be quantized at each pixel so that the representation set contains some power of two number of elements for representation in a computer memory.

Definition 3 A *halftoning algorithm* is a mapping from the set of all grey-scale images into the set of all binary images. That is:

$$\text{halftoning algorithm: } [-1,1]^{M\times N} \rightarrow \{-1,1\}^{M\times N}$$

Definition 4 The *signum function* is defined:

$$\text{sgn}(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{else} \end{cases}$$

Definition 5 The *two-dimensional discrete spatial convolution* of two functions is given by

$$(\mathbf{w} ** \mathbf{x})_{i,j} \mapsto \sum\sum_{(k,l)} w_{k,l} x_{i-k,j-l}$$

The following notation has been adopted to distinguish among various types of mappings:

- Functions defined on a connected domain will be lower case.
- Functions defined on a discrete domain will be bold lower case.
- Functions described on a discrete domain for which it is convenient to think of as a matrix will be upper case.
- Transformed functions will have a tilde over them and will inherit their boldness and case. It should be clear from context whether the Fourier Transform, DTFT, DFT, or Z-Transform is being used.
- Spatial coordinates will be shown as subscripts.
- Temporal or transform coordinates will be shown as arguments.

1.3 Current Algorithms

For motivational and comparison purposes three of the most popular digital halftoning algorithms are presented. For a good overview of many current techniques see [SM81].

1.3.1 Ordered Dither

The popularity of ordered dither is largely due to the simplicity of the algorithm rather than its halftoning properties. The main advantage is that the ordered dither process is memoryless and can be applied directly to the data stream without buffering or performing complex computation, making it popular in FAX machines and laser printers. Also, the output is highly compressible, an advantage in FAX machines, but an indication that much information is lost by the algorithm.

The method converts a pixel to a binary value in a method that can be considered the electronic analogue of the classical printer's screen. A threshold matrix ("screen") is periodically applied to the input image. The entries of the threshold matrix represent the value to which the input pixel underneath is compared against. If the input falls below the threshold, a decision is made to set the output to black (ink). Otherwise, the output is set to white (no ink). In our terminology:

Definition 6 Let **u** and **y** be $M \times N$ *input* and *output* grey scale images respectively. Also let **T** be an $K \times L$ threshold matrix. Then for $(i,j) \in \mathcal{S}_{M\times N}$,

$$\text{ordered dither: } y_{i,j} = \text{sgn}(u_{i,j} - T_{i \bmod K+1, j \bmod L+1})$$

The algorithm has the effect of replacing a constant grey region of the image by a regular pattern with the approximately correct density. The number of these patterns, hence the effective grey-level quantization, is directly related to their spatial period. Therefore, increasing quantization levels requires a larger threshold matrix. However, the size of the threshold patterns is inversely related to the size of the smallest object that can be guaranteed to be preserved by the algorithm. Given this constraint, the algorithm will only produce acceptable results when the image is well oversampled or displayed at increased resolution.

The entries of the matrix can be chosen in many ways, each giving different dot patterns for a constant input. The best of these is *dispersed dot ordered dither*, of some order k which determines the effective quantization. The method introduces a minimal amount of low frequency structure by maximally separating dots for a given grey level input while supplying the appropriate overall density. To use this method, an accurate display device is needed to mark the isolated dots.

A fifth-order dispersed dot ordered dither, as derived in [Uli90], is used for the examples in this paper. The threshold matrix is given by

$$\mathbf{T} = \begin{pmatrix} 1 & 30 & 8 & 28 & 2 & 29 & 7 & 27 \\ 17 & 9 & 24 & 16 & 18 & 10 & 23 & 15 \\ 5 & 25 & 3 & 32 & 6 & 26 & 4 & 31 \\ 21 & 13 & 19 & 11 & 22 & 14 & 20 & 12 \\ 2 & 29 & 7 & 27 & 1 & 30 & 8 & 28 \\ 18 & 10 & 23 & 15 & 17 & 9 & 24 & 16 \\ 6 & 26 & 4 & 31 & 5 & 25 & 3 & 32 \\ 22 & 14 & 20 & 12 & 21 & 13 & 19 & 11 \end{pmatrix}$$

where the input has been scaled to fit the range $[0,33]$. Figure 5 shows all $2^k + 1 = 33$ possible representation values using this method. Dispersed dot dithering methods can be seen to suffer from artificial contouring because of the very discrete change in grey-level representation in a slowly varying region (see Figure 26). Also apparent are loss of edges and introduction of periodic microstructure. The visibility of periodic microstructure can be reduced by using a lower order dither, but at the expense of some grey level representation.

1.3.2 Error Diffusion

Error diffusion is a very popular halftoning algorithm because it does much to correct the problems of dithering yet remains fairly computationally simple. As the name implies, error diffusion begins at one corner of the image and proceeds to distribute the halftoning error across the image so as to make the average error zero. A more rigorous explanation is given later. The standard error diffusion algorithm can be described by the following state and output difference equations.

Definition 7 Let **u**, **x**, **y** be $M \times N$ *input, state*, and *output* grey scale images respectively. Also, define $u_{i,j}, x_{i,j}, y_{i,j} \equiv 0$ for $(i,j) \notin \mathcal{S}_{M\times N}$. Let **w**, the error diffusion filter, be defined

on a causal mask with non-zero support $\mathcal{S}$. Then, error diffusion is defined to sequentially generate the elements of the state and output images according to:

$$x_{i,j} = u_{i,j} - (\mathbf{w} ** g \circ \mathbf{x})_{i,j} \quad (2)$$

$$y_{i,j} = \text{sgn}(x_{i,j}) \quad (3)$$

where $g(x) \triangleq \text{sgn}(x) - x$. The states and corresponding outputs are generated in any order for which the convolution makes sense.

The most popular masks have finite non-symmetric half-plane support with $\sum\sum_{(i,j)\in\mathcal{S}} w_{i,j} = 1$. This allows the recursion to proceed on a row-by-row basis without violating causality. The unity sum condition maintains the proper representation density. Some popular 12-weight error diffusion filters of this type are shown in Figure 4.

All three filters introduce a good deal of anisotropic microstructure (See Figures 6,7,and 8). It can also be noted that the high frequency information in the image is actually magnified, which may be desirable in certain applications.

1.3.3 Mean Square Error Minimization

In Section 1.1, it was discussed that the mean-square filtered error (MSE) between the input and output images could be used as a measure of halftoning algorithm performance. By this criteria, minimizing the MSE over all possible binary output images will result in an optimal halftone. Therefore, this criteria along with a minimization algorithm would perform as a halftoning technique.

There are many means of achieving at least a "local minimum" which should produce an acceptable output. One of these, simulated annealing, is employed by [SRM91] for this purpose. Of more interest here, [AK88, and others by the same authors] proposes the use of Hopfield type neural networks to perform the minimization. A slightly more general analysis of this technique is given here so that it is consistent with the block diagram framework.

For convenience, we write the $M \times N$ input, output, and state images as vectors of length MN by applying an ordering to their support $\mathcal{S}_{M\times N}$. Let $\mathbf{A}$ and $\mathbf{B}$ be $MN \times MN$ convolution matrices which represent some linear filter model for the systems $\mathcal{A}$ and $\mathcal{B}$ with respect to this ordering. Then the filtered or frequency-weighted MSE between the input image $\mathbf{u}$ and output image $\mathbf{y}$ can be written

$$\text{dist}(\mathbf{y}, \mathbf{u}) = (\mathbf{Ay} - \mathbf{Bu})^T(\mathbf{Ay} - \mathbf{Bu})$$

Now, for a Hopfield neural network

$$\frac{d}{dt}\mathbf{x}(t) = -\mathbf{x}(t) + \mathbf{W}\mathbf{y}(t) + \mathbf{V}\mathbf{u} \quad (4)$$

$$\mathbf{y}(t) = \text{sgn}(\mathbf{x}(t)) \quad (5)$$

where $\mathbf{W}$ and $\mathbf{V}$ represent weighting matrices, the network finds a local minimum for the well known Lyapunov-energy function

$$E = -\frac{1}{2}\mathbf{y}^T\mathbf{W}\mathbf{y} - \mathbf{u}^T\mathbf{V}^T\mathbf{y} \quad (6)$$

Now,

$$\text{dist}(\mathbf{y},\mathbf{u}) = (\mathbf{Ay}-\mathbf{Bu})^T(\mathbf{Ay}-\mathbf{Bu}) = \mathbf{y}^T\mathbf{A}^T\mathbf{Ay} - \mathbf{y}^T\mathbf{A}^T\mathbf{Bu} - \mathbf{u}^T\mathbf{B}^T\mathbf{Ay} + \mathbf{u}^T\mathbf{B}^T\mathbf{Bu}$$

since $\mathbf{u}$ represents a constant input vector, minimizing $\text{dist}(\mathbf{y},\mathbf{u})$ is equivalent to

$$\min_{\mathbf{y}\in\{-1,1\}^{MN}} \frac{1}{2}\mathbf{y}^T\mathbf{A}^T\mathbf{Ay} - \frac{1}{2}\mathbf{y}^T\mathbf{A}^T\mathbf{Bu} - \frac{1}{2}\mathbf{u}^T\mathbf{B}^T\mathbf{Ay} = \frac{1}{2}\mathbf{y}^T\mathbf{A}^T\mathbf{Ay} - \mathbf{u}^T\mathbf{B}^T\mathbf{Ay} \qquad (7)$$

Matching terms with Equation 6 it can be seen that by choosing $\mathbf{W} = -\mathbf{A}^T\mathbf{A}$ and $\mathbf{V} = \mathbf{A}^T\mathbf{B}$ the neural network will find a local minimum of the desired error function.

Several methods for choosing the $\mathbf{A}$ and $\mathbf{B}$ matrices are proposed in [KA91]. Using our framework notation, one method is to derive a linear approximation to $\mathcal{A}$ is by using an ideal reconstruction to model the display and the MTF to model viewing. A discrete space filter with small non-zero support is found by windowing the inverse DFT of a sampled version of this approximation in the frequency domain. Then, $\mathbf{A}$ represents the convolution done with this filter[4]. For choice of $\mathbf{B}$, $\mathbf{B} = \mathbf{A}$ and $\mathbf{B} = (\mathbf{A}^T)^{-1}$ are both suggested for their simplicity. The second case will over-enhance high frequencies since it demands that a low-pass version of the output look like a high-frequency gain version of the input.

A little reflection will reveal that multiplying the transpose of a symmetric convolution matrix by another convolution matrix gives the matrix for the convolution of the two. One immediate disadvantageous consequence of this is that if the convolution kernel of the desired filter has support with L_∞ radius r then the weight matrix will have local interconnections with radius $2r$. Therefore, the choice of $\mathbf{A}$ and $\mathbf{B}$ must be constrained to be very diameter-limited so that the resulting weight matrices do not have too many connections to make implementation feasible. The class of all possible weight matrices is therefore artificially reduced.

Several other methods for deriving weights are suggested [KA91], including training the network on outputs produced by error diffusion. None of the actual weights achieved by any of the methods are reported.

1.4 Cellular Neural Networks

A Cellular Neural Network (CNN), as first proposed in [CY88b], is a continuous time neural network with diameter-limited local interconnections and a unity-gain piecewise linear approximation to the standard sigmoidal output function. The 'neurons' are placed in a regular array and 'synaptic' connections are allowed only locally. Due to this topology, the CNN can be considered a composite system – it incorporates discrete space filtering and continuous time dynamics. The local nature of the interconnections are critical when considering VLSI implementation.

A particular subclass of CNN which constrains the allowed interconnections is considered in this paper:

Definition 8 Let $\mathbf{u},\mathbf{x},\mathbf{y}$ be the $M\times N$ *input, state,* and *output* grey scale images respectively. Let $\mathbf{A}$ and $\mathbf{B}$ be functions such that $\mathbf{A},\mathbf{B} : \mathcal{Z}^2 \rightarrow \Re$ with $A_{i,j}, B_{i,j} = A_{i,j}, B_{-i,-j}$ and $A_{i,j}, B_{i,j} = 0$ for $||(i,j)||_\infty \leq r$, some small neighborhood radius. Let I be a constant. Also,

[4]This matrix will have the form symmetric block Toeplitz with symmetric Toeplitz blocks.

define $u_{i,j}, x_{i,j}, y_{i,j} \equiv 0$ for $(i,j) \notin \mathcal{S}_{M\times N}$. The *symmetric space-invariant Cellular Neural Network* can be described by the following state and output equations operating on $\mathcal{S}_{M\times N}$:

$$\frac{d}{dt}x_{i,j}(t) = -x_{i,j}(t) + (\mathbf{A} ** \mathbf{y}(t))_{i,j} + (\mathbf{B} ** \mathbf{u})_{i,j} + I \quad (8)$$

$$y_{i,j}(t) = f(x_{i,j}(t)) \quad (9)$$

where $f : x \mapsto \frac{1}{2}(|x+1| - |x-1|), x \in \Re$

The functions **A** and **B** are called *cloning templates* and can be considered $2r+1 \times 2r+1$ matrices. The condition $A_{i,j} = A_{-i,-j}$ is called template *symmetry*. Note that template symmetry is not required by the original definition [CY88b] and is what enables us to write the state equations in spatial convolution form without reflecting the template.

An even stronger condition on the templates is *isotropy*. For isotropy, the template must satisfy $A_{i,j} = A_{k,l}$ for (i,j) and (k,l) the same L_2 distance from the origin. This condition implies template symmetry and will be useful in halftoning applications.

A very useful theorem follows:

Theorem 1 [[CY88b]] If $A_{0,0} > 1$ and A is a symmetric template, then the system converges with

$$\lim_{t\to\infty} |x_{i,j}(t)| > 1$$

and

$$\lim_{t\to\infty} y_{i,j}(t) \in \{\pm 1\}$$

for $(i,j) \in \mathcal{S}_{M\times N}$.

It can be shown in a more general setting that that symmetry is not required [NSRC90].

The topology and local nature of the CNN suggest amenability to many types of highly parallel analog computation. In fact, several fundamental image processing tasks have been converted to CNN templates [IEE90, CYK91]. The speed of the CNN is mostly constrained by the input/output bandwidth. The use of optical sensors to supply the input in parallel is being investigated. In such a system, applications which map continuous valued inputs to binary output, like halftoning, would be much faster than a digital system performing the same task.

The CNN model is preferable for implementation in this application over the Hopfield network for several reasons. The local geometric interconnection pattern and the form of the output nonlinearity are especially well-suited for implementation in VLSI circuitry. In fact, CNN VLSI circuits have been built and tested [CC91a]. In addition, sound theoretical results are known for the boundedness of the internal states of the system – a necessity when considering circuit implementation [CY88b]. Also, the unity-gain nonlinearity disallows the output oscillations common in a Hopfield network providing fixed-point behavior in the output. Moreover, for intuitive reasons it is expected that a CNN will find a better local minimum for the halftoning application as explained in Section 2.1. Finally, the possibility of using nonlinear templates has been studied [RC90] and may prove useful in a halftoning context.

2 Halftoning with CNN

2.1 Development of the CNN Halftoning Algorithm

The error diffusion algorithm for halftoning is often considered to produce 'good' halftones. For this reason, we explore the possibility of developing a CNN halftoning algorithm which emulates the best aspects of error diffusion.

First, it is shown that error diffusion attempts to solve a certain type of minimization problem. Then, possible sources of defects in the algorithm are identified and corrections are suggested. Finally, it is shown that a CNN can achieve the same minimization and correct some defects simultaneously.

The following lemma will be used several times.

Lemma 1 The solution to

$$\min_{y\in\{-1,1\}} |by + c| \; b \neq 0, c \in \Re$$

is

$$y = -\operatorname{sgn}(bc)$$

Proof:

$$|b\operatorname{sgn}(bc) + c| - |-b\operatorname{sgn}(bc) + c| =$$
$$|b\operatorname{sgn}(b) + c\operatorname{sgn}(c)| - |b\operatorname{sgn}(b) - c\operatorname{sgn}(c)| =$$
$$|b| + |c| - ||b| - |c|| \geq |b| + |c| - |b| - |c| \geq 0$$

■

Proposition 1 The error diffusion algorithm described in Definition 7 is equivalent to a determination of the output by the following decision criterion:

$$\min_{y_{i,j}\in\{-1,1\}} |((\mathbf{h} ** (\mathbf{y} - \mathbf{u}))_{i,j})|$$

with

$$\mathbf{h} = \mathcal{Z}^{-1}[1 + \frac{\tilde{\mathbf{w}}}{1 - \tilde{\mathbf{w}}}]$$

where the $y_{i,j} \in \mathcal{S}_{M\times N}$ are chosen in the causal order such that the convolution makes sense.

Proof:

$$\begin{aligned} x_{i,j} &= u_{i,j} - (\mathbf{w} ** \mathbf{g} \circ \mathbf{x})_{i,j} \\ &= u_{i,j} - (\mathbf{w} ** (\mathbf{y} - \mathbf{x}))_{i,j} \\ &= u_{i,j} - (\mathbf{w} ** \mathbf{y})_{i,j} + (\mathbf{w} ** \mathbf{x})_{i,j} \end{aligned}$$

then, taking the two-dimensional Z-transform:

$$\begin{aligned} \tilde{\mathbf{x}} &= \tilde{\mathbf{u}} - \tilde{\mathbf{w}}\tilde{\mathbf{y}} + \tilde{\mathbf{w}}\tilde{\mathbf{x}} \\ (1 - \tilde{\mathbf{w}})\tilde{\mathbf{x}} &= \tilde{\mathbf{u}} - \tilde{\mathbf{w}}\tilde{\mathbf{y}} \\ \tilde{\mathbf{x}} &= \frac{\tilde{\mathbf{u}} - \tilde{\mathbf{w}}\tilde{\mathbf{y}}}{1 - \tilde{\mathbf{w}}} \\ &= \tilde{\mathbf{u}} - \frac{\tilde{\mathbf{w}}}{1 - \tilde{\mathbf{w}}}(\tilde{\mathbf{y}} - \tilde{\mathbf{u}}) \\ &= \tilde{\mathbf{h}}(\tilde{\mathbf{u}} - \tilde{\mathbf{y}}) - \tilde{\mathbf{y}} \end{aligned}$$

then, by inverse transformation,

$$x_{i,j} = (\mathbf{h} ** (\mathbf{u} - \mathbf{y}))_{i,j} - y_{i,j}$$

substitution into Equation 3 gives

$$y_{i,j} = \text{sgn}((\mathbf{h} ** (\mathbf{u} - \mathbf{y}))_{i,j} - y_{i,j})$$

now $h_{0,0} = 1$ by the initial value property and the fact that $w_{0,0} = 0$. Namely,

$$\lim_{z_1, z_2 \to \infty} \tilde{\mathbf{h}}(z_1, z_2) = 1$$

So, by Lemma 1, this satisfies the given minimization. ▪

In words, the goal of error diffusion is at each successive pixel, to minimize the absolute value of the filtered error of all previous decisions. Now, it is clear that the filter **h** is not BIBO stable. This follows since $\tilde{\mathbf{w}}(1,1) = \sum\sum_{(i,j)\in\mathcal{S}} w_{i,j} = 1$ means that there is a pole of $\tilde{\mathbf{h}}$ on the unit surface. However, on the finite domain given, the output will always be bounded. In fact $\mathbf{h}_{i,j}$ can be considered to be space-varying FIR filter if it is restricted to the domain where the convolution makes sense.

This interpretation of the action performed by the error filter gives strong insight into the meaning of the filter coefficients. The shape of the filter in the frequency domain will determine the characteristics of the halftoning noise. For instance, if the resulting IIR filter is not circularly symmetric, the algorithm will favor some directions over others. Consequently, it will be more likely that halftoning noise will be placed in less-weighted directions. This is one source of the many squiggles that appear in error diffusion output for popular diffusion weights [5]. Another thing that would be nice to correct is the directionality of the algorithm which is imposed by iterative computation. The non-symmetric half plane mask could be converted to a filter with whole-plane support. This will mean that all the outputs will need to be determined simultaneously. Finally, comparing the block diagram in Figure 1 to the above interpretation, it can be seen that error diffusion conforms to our model. If the relevant error is taken to be the absolute value of the difference between the outputs and the filter **h** represents the visual system and other output blocks. This implies a loss of generality, since according to our model, separate filters could be used. Reintroducing this generality switches the emphasis from "filtering the error between the input and output" to "error between filtered input and filtered output."

The CNN can be made to perform a halftoning operation by supplying the input image to **u** and interpreting **y** as the halftone output, as follows:

Definition 9 Let **a** and **b** be the impulse responses of chosen linear filter approximations to given systems $\mathcal{A}$ and $\mathcal{B}$. Then the associated *CNN Halftone Template* is given by

$$A_{i,j} = \begin{cases} 1 + \varepsilon,\ \varepsilon > 0, \text{small} & \text{if } (i,j) = (0,0) \\ -a_{i,j} & \text{else} \end{cases}$$

$$B_{i,j} = b_{i,j}$$

$$I = 0$$

[5] This suggests that the interpretation here could provide a way to derive error filters to control the types of artifacts produced and improving on those currently in use.

The initial states of the CNN carry no exact meaning in this context and are assumed to be small and random. To retain the local nature of the connections, the approximation filters **a** and **b** must have non-zero impulse response on a small support. A method for choosing these filters is explained in the next section. The halftone CNN will always converge to binary outputs given by Theorem 1.

If the output image at a specific cell at a specific time is examined, it can be seen that the dynamics are "heading" the output in the proper direction to perform the same minimization as the error diffusion algorithm. That is, the sign of the derivative is the same as the direction of the optimal output and the magnitude is related to the cost of making a wrong decision! Now the advantage of not making a solid $+1$ or -1 decision at this point is clear. The cells with the greatest error will be corrected the fastest. The correspondence to error diffusion is clarified in the following claim:

Proposition 2 Assume the halftoning CNN has reached steady state. Then, for each cell (i,j) define

$$K_{i,j} = -(\mathbf{a} ** \mathbf{y})_{i,j} + (\mathbf{b} ** \mathbf{u})_{i,j} + a_{0,0}y_{i,j}$$

Now, by Lemma 1, the solution to

$$\min_{y_{i,j}\in\{-1,1\}} |(\mathbf{a} ** \mathbf{y})_{i,j} - (\mathbf{b} ** \mathbf{u})_{i,j}|$$

is $\hat{y}_{i,j} = \operatorname{sgn}(K_{i,j})$. Let $\hat{\mathbf{y}}$ be the output $\mathbf{y}$ with $y_{i,j}$ replaced by $\hat{y}_{i,j}$.
Then, the halftoning CNN has converged to an output $y_{i,j} \in \{-1,1\}$ which satisfies

$$|(\mathbf{a} ** \mathbf{y})_{i,j} - (\mathbf{b} ** \mathbf{u})_{i,j}| - |(\mathbf{a} ** \hat{\mathbf{y}})_{i,j} - (\mathbf{b} ** \mathbf{u})_{i,j}| < 2\min\{\varepsilon, |a_{0,0}|\}$$

Proof: The convergence to a binary output is guaranteed directly by $A_{0,0} > 1$ and Theorem 1. Therefore we have $|x_{i,j}| > 1$. Now, assume the output has converged to the steady state. Then,

$$0 = -x_{i,j}(\infty) + (\mathbf{A} ** \mathbf{y}(\infty))_{i,j} + (\mathbf{B} ** \mathbf{u})_{i,j}$$

$$0 = -x_{i,j}(\infty) + (1+\varepsilon)y_{i,j}(\infty) + K_{i,j}$$

now, since $|x_{i,j}| > 1$, $y_{i,j} = \operatorname{sgn}(x_{i,j})$ so

$$x_{i,j}(\infty) = (1+\varepsilon)\operatorname{sgn}(x_{i,j}) + K_{i,j}$$

now, for $|K_{i,j}| > \varepsilon$,

$$y_{i,j} = \operatorname{sgn}(K_{i,j})$$

is the solution. This is the optimal one, so the inequality certainly holds. Now, for $|K_{i,j}| < \varepsilon$, both possibilities, $y_{i,j} = \pm\operatorname{sgn}(K_{i,j})$, satisfy the condition. In this case,

$$|-\operatorname{sgn}(K_{i,j})a_{0,0} - K_{i,j}| - |\operatorname{sgn}(K_{i,j})a_{0,0} - K_{i,j}|$$

$$= 2\min\{K_{i,j}, |a_{0,0}|\}$$

$$\leq 2\min\{\varepsilon, |a_{0,0}|\}$$

■

The Proposition claims that, when considering a specific cell in steady state, that cell has converged to minimize the difference at that location between a filtered version of the input and a filtered version of the output, unless making the wrong decision introduced nearly the same error as the correct decision would have. In practice, the center element $a_{0,0}$ will be bigger than ε so that the usefulness of this inequality is concerned with ε. Note that as $\varepsilon \to 0$ the CNN will be made to make decisions which are arbitrarily close to the form of those made by error diffusion.

2.2 Choosing Template Coefficients and Effects on Performance

Given the above discussion of interpretation and the block diagram description of the halftone evaluation process, it would seem that choosing template coefficients would be straightforward. Namely, **a** and **b** should be chosen to be linear filter approximations to the filtering done in the halftone and ideal paths respectively of Figure 1. However, in a real system which is to be implemented, there are many complications. These complications are introduced by the physical constraints of the VLSI process which then exacerbate the mathematical constraints due to the nature of the algorithm. The primary physical constraint is on network connectivity due to spatial constraints for VLSI routing. To build a reasonably sized array in current technology, the A and B template size should be 5×5 or smaller, meaning $r \leq 2$. A secondary problem in construction is the relative magnitude of the template coefficients. From [CC91b] it can be seen that for considerations of silicon area usage, the ratio between the maximum and minimum template entries should not be too large. However, this is not a great concern since the nature of the algorithm also does not encourage big ratios of this type, as explained below.

The linear filter model for the eye plus display blocks must be realized in a filter of finite size. This and template isotropy drastically reduce the degrees of freedom for filter design. Methods of designing this filter used in [KA91] include sampled and windowed spatial impulse responses or finding the filter corresponding to the least squares approximation in the frequency domain. This approach proves to be deficient for this application. There are three reasons for this, which are enumerated here as additional design constraints on template design.

1. The relative magnitudes of the entries of the templates have a strong effect on the DC (that is, constant input) transfer characteristics. Ideally, the relation between a constant input and the average halftone output density would be a linear function in some middle range of intensity with some threshold characteristic at the extremities. Now, the template size determines the effective quantization of the output since any given cell can only modify itself according to the density it sees in its template neighborhood. For instance, in a 5×5 neighborhood there are only 26 representable densities. If some of the template entries are small, or worse, negative, it effectively reduces the size of the neighborhood. It can be shown, by manipulation of the dynamical equations in steady state, that the range of input values $u_{i,j} = u$ such that only one cell in any neighborhood will dissent from the others is given by

$$\frac{\varepsilon - 2\min_{i,j}\{a_{i,j}\} + \sum_{i,j\neq(0,0)} a_{i,j}}{\sum_{i,j} b_{i,j}} \leq |u| \leq \frac{-\varepsilon + \sum_{i,j\neq(0,0)} a_{i,j}}{\sum_{i,j} b_{i,j}} \tag{10}$$

   That is, there will be only one cell in any given neighborhood with output $-\operatorname{sgn}(u)$. This corresponds to the intensity levels of input which map to the first reliably representable shade of grey above complete black or below complete white. Ignoring the

effects of ε, it can be seen that reducing the minimum template element diminishes this region, when ideally we would like to have about 1/26th of the input space mapped here. And, in fact, if $\min_{i,j}\{a_{i,j}\}$ is not as big as ε, these representations will never be used. The resulting DC transfer characteristic will have a sharp drop near the ends as seen in Figure 10.

2. The lack of circular symmetry of the DTFT of the template filters is an important source of the microstructure which can appear in the output. By circular symmetry it is meant that the lines of constant magnitude in the frequency domain are circles. This can be explained by the minimization interpretation of the halftoning algorithm. If the frequency representation is not circular, more of the error will be tolerated in some directions over others. Although, the eye may be less sensitive to error in these directions by the linear model, in reality correlated error in the form of squiggles produce false texture and misleading structure.

3. Nonmonotinicity along radii of the DTFT of the template filters is an important source of clustering or clumping in the output. Again, by the water-filling interpretation, more error gets put into the frequencies corresponding to the valleys of the function. If these valleys cover a narrow range of frequencies, the output will show periodic noise. This provides an important understanding into the pattern formation properties of the CNN.

A relation can be made showing Items 1 and 2 to conflict with our primary design goal, and that Item 3 conflicts with Item 1.

Item 1 conflicts with the goal of halftoning to observe the image as closely as possible and still retain most the original information. That means the filter representation of the eye must extend into the high frequencies. However, it is well known that the widths of the impulse response and frequency representation are inversely proportional. Therefore halftoning for close observation will always be at the expense of the linearity and quantization of the DC transfer characteristic.

Item 2 directly contradicts the anisotropic nature of the eye, so we must disregard the idea that their is an advantage to putting noise along some directions than others. Even worse, it is not even possible to get circular symmetry, in general, with a FIR filter even if the template is chosen to be isotropic.

Item 3 conflicts with Item 2 by another well known property of the DTFT. Namely, sharp cutoffs in the spatial domain produces ringing in the frequency domain. So, if we desire no ringing in frequency, the template coefficients must become near-zero along the edge of the template, a violation of Item 1.

3 Results

3.1 Evaluation Techniques

Several test images were developed to expose many of the potential deficiencies of the algorithm. They include a grey scale ramp, an Arden chart, and the standard 'Lena' image.

The grey scale ramp is a 64×512 image which starts with minimal reflectivity (as provided by the toner) on one side and proceeds to full reflectivity (as provided by the paper). The ramp reveals many things about a halftoning algorithm including artificial contouring, number of representation levels, and microstructure associated with certain levels of constant input.

The Arden chart used in this test is 256×256 grey scale image with discrete frequencies to prevent aliasing. Although the visual system is a nonlinear processor, the Arden chart has proved important in certain clinical testing of acuity and spatial frequency contrast sensitivity [Ard78]. In this sense, halftoning an Arden chart can give an idea of the preservation of horizontal frequencies by the algorithm. Ideally, we would like to see exactly the same experimental MTF when observing the halftoned chart as the original. Of course, this idea could be generalized to frequencies at any angle.

Finally, the 512×512 standard 'Lena' image provides a real-world image for comparison purposes. It is suggested in [SM81] that a real image is the most reasonable criteria.

3.2 Template Design

Three halftoning templates were developed for the CNN. A reconstruction function of an ideal zero-order hold was assumed. That is, perfectly square, non-overlapping, black or white boxes are to be marked on the display medium. The frequency representation of the zero-order hold is nearly unity within the unaliased frequency region and tapers off slowly outside. Therefore, a discrete space filter approximation to the visual MTF must decrease to near zero to avoid aliasing so that it accurately represents the reconstruction plus eye functions.

The desired filter was chosen to be the composition of the linear zero-order hold and a linear model for the eye. Call this filter $\mathbf{q}$ for the halftone path and $\mathbf{r}$ in the ideal path. The linear model for the eye was chosen to be a modified MTF – the low frequency characteristic was changed to be unity gain[6]. Then, we would like $\mathbf{a}$ to be as close as possible to $\mathbf{q}$ but obey the constraints discussed in the previous section.

Template 1 has 5×5 **A** and **B** templates and is optimized for presentation of 8 dots/mm at 0.5 meters. Template 2 also has 5×5 templates and is optimized for viewing at twice the distance, or equivalently, the same distance with twice the resolution. Template 3 is a hybrid 5×5 in the **A** template and 3×3 in the **B** template. What it is optimized to do will be discussed below.

The template design was accomplished in the following manner. A spatially isotropic 5x5 filter

$$\mathbf{a} = \begin{pmatrix} a_5 & a_4 & a_3 & a_4 & a_5 \\ a_4 & a_2 & a_1 & a_2 & a_4 \\ a_3 & a_1 & a_0 & a_1 & a_3 \\ a_4 & a_2 & a_1 & a_2 & a_4 \\ a_5 & a_4 & a_3 & a_4 & a_5 \end{pmatrix}$$

has six unique elements. Using the shorthand $\mathbf{a} = \{a_0, a_1, a_2, a_3, a_4, a_5\}^T$, and the DTFT kernel

$$\mathbf{k} = \begin{Bmatrix} 1 \\ 2\cos\omega_1 + 2\cos\omega_2 \\ 2\cos(\omega_1 + \omega_2) + 2\cos(\omega_1 - \omega_2) \\ 2\cos 2\omega_1 + 2\cos 2\omega_2 \\ 2\cos(\omega_1 + 2\omega_2) + 2\cos(2\omega_1 + \omega_2) + 2\cos(\omega_1 - 2\omega_2) + 2\cos(2\omega_1 - \omega_2) \\ 2\cos(2\omega_1 + 2\omega_2) + 2\cos(2\omega_1 - 2\omega_2) \end{Bmatrix}$$

the DFT of the template filter can be written $\tilde{\mathbf{a}}(\omega_1, \omega_2) = \mathbf{a}^T\mathbf{k}$. One of these is not independent of the others, since all scalar multiples of a filter are equivalent for this purpose. The

[6]Since the output intensity range is fixed, the average grey level cannot be allowed to drift.

sum of the coefficients was demanded to be one for unity gain. Then the coefficients were forced to be bounded below. The desired frequency response was sampled and a least-square minimization was performed to match these points. If the resulting filter strongly violated any of the design principles, the desired function was sampled in different locations chosen intuitively, and the procedure was repeated. A disadvantage of this design procedure is that it does not terminate in an obvious way.

The conditions can be stated more formally as

$$\min_{a_i} \sum_{\vec{s}} (\tilde{\mathbf{q}}(\vec{s}) - \tilde{\mathbf{a}}(\vec{s}))^2$$

subject to

$$\sum_i a_i = 1$$

$$\frac{|\max_i a_i|}{\min a_i} < \sigma_1$$

$$\max_{r,\theta} \frac{d}{dr} \tilde{\mathbf{a}}(r\cos\theta, r\sin\theta) \leq +\sigma_2$$

$$\|\tilde{\mathbf{a}}(r\cos\theta, r\sin\theta) - \frac{1}{2\pi r}\int_0^{2\pi} \tilde{\mathbf{a}}(r\cos\theta, r\sin\theta)d\theta\|_\infty < +\sigma_3$$

where σ_1, σ_2, and σ_3 were chosen to be as small as possible. The first two constraints were attained computationally, while the last two were measured visually and adjusted by altering the sample points, $\vec{s}$. Following this procedure gave these two templates:

Template 1:
$$\begin{aligned} A^1 &= \{1.05, -0.2342, -0.1767, -0.0666, -0.0155, -0.0155\} \quad \times 1.1317 \\ B^1 &= \{1.00, \ 0.2342, \ 0.1767, \ 0.0666, \ 0.0155, \ 0.0155\} \end{aligned}$$

Template 2:
$$\begin{aligned} A^2 &= \{1.05, -0.6041, -0.3592, -0.1298, -0.0860, -0.0304\} \\ B^2 &= \{1.00, \ 0.6041, \ 0.3592, \ 0.1298, \ 0.0860, \ 0.0304\} \quad \times 1.1068 \end{aligned}$$

The design of the third template was motivated by the observation that the wider template spatially gives better DC characteristics while narrow templates have a better frequency response. Therefore, it makes sense to develop a spatial wide **A** template so that the halftoning patterns look good and a spatial narrow **B** template to enhance the frequency components of the image where they exist. In effect, we are demanding that an image viewed from a distance appear as it were close up. This will boost the high frequencies. The same **A** template was used as in Template 2, and the **B** template was designed by the above procedure, but with the viewing distance that of Template 1 and the corner coefficients constrained to zero. Note that this template will have the same DC characteristics as Template 2.

Template 3:
$$\begin{aligned} A^3 &= \{1.05, -0.6041, -0.3592, -0.1298, -0.0860, -0.0304\} \\ B^3 &= \{1.00, \ 0.3565, \ 0.1672, \ 0.0322, \ 0.0000, \ 0.0000\} \quad \times 2.1223 \end{aligned}$$

The scaling terms in the above templates will now be explained. The templates must be scaled so that the average density of the output will have a threshold characteristic as a function of the input giving the proper graphic punch. Using inequality of Equation 10 it can be seen that the critical input is given by

$$|u_{crit}| = \frac{-\varepsilon + r_A(\sum_{(i,j)} a_{i,j} - 1)}{r_B \sum_{(i,j)} b_{i,j}}$$

Either-or r_A or r_B , the scaling factors, can be solved for to give the desired cutoff points. Note that the critical input for the black and white cutoffs are symmetrical. For the examples given here, the cutoffs are designed for $|u_{crit}| = .75$ meaning that the middle 75% of the range is linear and the outer 12.5% are thresholded. For they ramp input the template was changed to $|u_{crit}| = .96$ so that the interesting things are displayed.

Two other templates were designed for the Hopfield network using the Anastassiou technique. The squared MTF was sampled at rates corresponding to the two viewing distances above. The closest viewing distance used an L_1 metric for measuring frequency distance from the origin. For the further an L_2 metric was used. The resulting templates are derived by hard windowing the inverse DFT, and are

$$\text{Template 4:} \quad \begin{aligned} A^4 &= \{0.00, -0.7112, -0.4766, -0.2825, -0.1413, \ 0.0400\} \\ B^4 &= \{1.00, \ 0.7112, \ 0.4766, \ 0.2825, \ 0.1413, -0.0400\} \ \times 1.6353 \end{aligned}$$

$$\text{Template 5:} \quad \begin{aligned} A^5 &= \{0.00, -0.8757, -0.7665, -0.5861, -0.5116, -0.3363\} \\ B^5 &= \{1.00, \ 0.8757, \ 0.7665, \ 0.5861, \ 0.5116, \ 0.3363\} \ \times 1.2464 \end{aligned}$$

where the scaling serves the same purpose as the CNN examples. The DTFT of all the templates are shown in Figures 17 through 20. Notice that the low-pass attenuation characteristic of the MTF is completely lost in Figures 19 and 20.

In addition, the standard algorithms, 5th order dispersed dot ordered dither and the Jarvis error diffusion were run on the test images. A threshold nonlinearity was applied to the Lena Image with cutoff of 0.8 before processing to supply the proper contrast.

3.3 Experiments and Evaluation

Simulations for the dynamical systems were run using numerical integration with small enough step size to ensure proper convergence. The initial conditions on the state were chosen to be uniform pseudo-random on $[-0.1, 0.1]$ for the CNN simulations. The error diffusion and ordered dither output are simply calculated by a one pass calculation.

The outputs for a ramp input are shown in Figures 5 through 15. All three error diffusion templates exhibit much meandering structure near $|u| = \frac{1}{3}$. The squiggles in the stucki filter are mostly in the diagonal directions whereas in the Jarvis and Floyd filters the noise is correlated in either the vertical or diagonal directions depending on the input. Notice all three examples exhibit artificial contouring due to the transient. All three have excellent effective quantization capabilities for low level inputs. However it is questionable whether a few random white dots on a constant black background will convince the observer that this is a slightly lighter shade or in fact just some stray spots. The Floyd and Stucki filters produce the checkerboard pattern near the middle grey, which is desirable. Unfortunately this pattern is not entirely stable and is frequently interrupted with patches of courser structure.

The ordered dither output shows the many defects of this technique. There are clear artificial contours as the input crosses a boundary between two of the 33 representation levels. In addition the noise is strongly coordinated in the diagonal directions (which is better than the orthogonals). Also distracting are the patterns produced over areas of constant input.

The two CNN ramp outputs are displayed in Figure 9 and 11. Template 1 exhibits some course two and three pixels connected grainyness. The output of Template 2 and 3 shows less correlated noise. The noise that does exist is more diagonal which is less important to the human visual system. The histograms representing the DC transfer characteristics also show

that the second template has less correlated structure. Also, it can be seen that Template 1 has a highly nonlinear representation for extreme inputs and does not represent them well. This is caused by the small values in the template. It is possible that using nonlinear template elements could correct for this as well as the nonlinear perception of intensity by the human eye.

The first template of the inverse DFT design shows lots of wiggles in the diagonal directions. In addition, the range of representations ends abruptly with a sharp drop-off meaning that many grey-levels cannot be approximated. The second shows heavy correlation in the orthogonal directions, as would be expected from the frequency domain anisotropy.

The CSF plots, Figures 21 through 24, demonstrate the design principle of frequency preservation. Template 1 has a very broad reproduction capability whereas Template 2 which was designed for more distant viewing attenuates structure associated with higher frequency. The CSF of Template 3 meets our expectation that this can be recovered by enhancing the high frequencies by modifying the B template. The error diffusion CSF demonstrates that the error diffusion algorithm actual enhances high frequencies even more.

Finally, we evaluate the Lena images in Figures 26 through 30. The ordered dither Lena reveals how annoying this technique can be. Almost all the fine detail has been destroyed, in Lena's hat and hair. In addition, the false contouring due to quantization is very apparent on Lena's cheek.

The error diffusion Lena shows good frequency rendition but lots of misleading microstructure. From context you will know that the squiggles which run from her forehead to her hat are algorithmic artifacts. But, without looking at the original would you guess that the down-right sloping lines in Lena's hat are part of the image or an artifact?

The CNN Lena images all appear fine grained. Template 1 exhibits good detail rendition and is optimized to appear like the original when viewed from 1.4 meters, the proper distance for this resolution. A fair amount of blocky noise is present as seen in the middle of the shoulder. Template 2 has a much improved dot correlation at the expense of small structure preservation. Notice the loss of texture in the upper part of the hat. Presumably, this would not have been a noticeable feature when viewed from 2.8 meters anyway. Template 3 recovers this frequency capability while keeping the better DC characteristics. For this reason, we consider Template 3 to provide the best rendition.

Another advantage of the CNN technique is now made clear. For the error diffusion, the image was thresholded before processing to increase the sharpness of the output. This represents a preprocessing loss of information. In the CNN, this information is lost during the halftoning process, but only if the input is spatially constant. If there is frequency information above the threshold it may still appear in the output. This accounts for the fact that the DC threshold must be lower for the CNN than the error diffusion to obtain the same level of graphic punch.

Upon first glance, the error diffusion output for the Lena image may be more appealing than the CNN output. The reason for this is primarily the sharpening properties of the error diffusion algorithm. However, as Ulichney states, "... the virtues of a halftoning scheme should be decoupled from its ability to sharpen. The improved output perceived from a method that intrinsically sharpens can misleadingly outweigh other shortcomings in its ability to render grey levels accurately and without algorithmic artifacts."[Uli90, page 334] This is true to the degree that you are able to prefilter to adjust for the sharpening properties. This is the case here, as we can adjust the **B** template to get the desired sharpening characteristics. To demonstrate, Figure 31 shows the output when using the A^2 template with an identity **B** template.

Another concern with this statement would be that if a certain frequency is completely filtered out of an image during halftoning, no amount of prefiltering will be able to save it. As seen from the CNN CSF this does not occur. Also note, with the CNN the sharpening and graphic punch can be incorporated as part of the algorithm with no additional processing cost.

3.4 Stability

Earlier it was mentioned that the class of halftoning CNN templates was stable due to template symmetry. However, in actual realization, the **A** template will not be perfectly symmetric. Stability results for nonsymmetric CNN templates [CR90, CW91] are available. In this case, however, the dominance of the **A** template can be used to show that the equivalent large feedback matrix is dominant [Ros88].

4 Applications

A halftoning CNN with optical inputs would be a practical manifestation, for a useful system, of the oft-discussed highly-parallel analog computation paradigm. The parallel optical inputs and serial binary output would provide input/output rates suitable to the processing speed. Since the halftoning operation is very robust to site errors, it is possible that integration on a wafer scale would have reasonable yields. Possible applications include FAX machines, scanner copiers, remote video, and other systems constrained by bandwidth or binary reproduction techniques.

For instance, in a FAX machine, system constraints include the communication channel (when a normal phone line is used), unpredictable reproduction capabilities, scanning time, scanning density and resolution, and processing time. A CNN scanning chip could solve some of these problems. The CNN halftoning simulations were shown to converge in the order of a microsecond. If a chip could be built that scanned a 64×64 region in one step, the data rate generated would be commensurate with that of current systems, with the advantage of an excellent halftoning algorithm. Current FAX machine use a dither type algorithm because of its low computational overhead and storage requirements. However, with the low sampling rate of current Group 3 FAX, this is not sufficient for quality renditions of images.

Several issues need still be addressed when considering a VLSI implementation. In the examples in this paper, the initial conditions were assumed to be small and random. The initial conditions in a real circuit, corresponding to capacitor voltages, would be heavily correlated to the previous image. Also, the effects of component variations are unknown. The effect may actually be beneficial for further reducing microstructure in the output as random variations in the error diffusion filter has been shown to do [Uli90]. Finally, in a real halftoning application compensation may be necessary to join without incongruity adjacent edges of separately halftoned portions of the input image.

5 Conclusion

The block diagram framework of Figure 1 provides a general approach to solving halftoning (and other image processing) problems with CNNs. The use of nonlinear template elements could provide better internal models of the ideal and halftone paths.

It has been shown that it is possible to develop excellent image halftones using the CNN paradigm. In addition, by careful choice of template coefficients, it is possible to chose interconnection diameters small enough that the system could have a reasonable VLSI implementation. The resulting halftoning system could perform real-time sampling and halftoning on a single microchip with binary output.

References

[AK88] Dimitris Anastassiou and Stefanos Kollias. Digital image halftoning using neural networks. In *Visual Communications and Image Processing '88*, pages 1062–1069. SPIE, 1988.

[Ana88a] Dimitris Anastassiou. Neural net based digital halftoning of images. In *Proceedings IEEE International Symposium on Circuits and Systems*, pages 507–510, Helsinki, Finland, June 1988. IEEE.

[Ana88b] Dimitris Anastassiou. Neural network based nonstandard A/D conversion. *Electronic Letters*, 24(10):619–620, May 1988.

[Ana89] Dimitris Anastassiou. Error diffusion coding for A/D conversion. *IEEE Transactions on Circuits and Systems*, 36(9):1175–1186, September 1989.

[Ard78] G. B. Arden. The importance of measuring contrast sensitivity in cases of visual disturbance. *British Journal of Ophthalmology*, 62:198–209, 1978.

[BKT89] P. Bijl, J. J. Koenderink, and A. Toet. Visibility of blobs with a gaussian luminance profile. *Vision Research*, 29(4):447–456, 1989.

[CC91a] J.M. Cruz and L.O. Chua. A CNN chip for connected component detection. *IEEE Transactions on Circuits and Systems*, 38(7), July 1991.

[CC91b] J.M. Cruz and L.O. Chua. High speed high density CMOS CNNs. To appear in *International Journal of Circuit Theory and Applications*,, 1991.

[CR90] L. O. Chua and Tamás Roska. Stability of a class of nonreciprocal cellular neural networks. *IEEE Transactions on Circuit and Systems*, 37:1520–1527, 1990.

[CW91] L. O. Chua and C. W. Wu. On the universe of stable cellular neural networks. ERL Memorandum UCB/ERL M91/31, University of California, Berkeley, 1991. to appear in *International Journal of Circuit Theory and Applications.*

[CY88a] Leon O. Chua and Lin Yang. Cellular Neural Networks: Applications. *IEEE Transactions on Circuits and Systems*, 32, October 1988.

[CY88b] Leon O. Chua and Lin Yang. Cellular Neural Networks: Theory. *IEEE Transactions on Circuits and Systems*, 32, October 1988.

[CYK91] L. O. Chua, L. Yang, and K. R. Krieg. Signal processing using cellular Neural Networks. *Journal of VLSI Signal Processing*, 3:25–52, 1991.

[IEE90] IEEE. *International Workshop on CNNs and Their Applications*, Budapest, Hungary, 1990. IEEE.

[KA91] Stefanos Kollias and Dimitris Anastassiou. A unified neural network approach to digital image halftoning. *IEEE Transactions on Signal Processing*, 39(4):980–984, April 1991.

[KTA89] Stefanos Kollias, Tu-Chih Tsai, and Dimitris Anastassiou. Image halftoning and reconstruction using a neural network. In *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 1787–1790, Glasgow, Scotland, May 1989. IEEE.

[MS74] James L. Mannos and David J. Sakrison. The effects of a visual fidelity criterion on the encoding of images. *IEEE Transactions on Information Theory*, 20(4):525–536, July 1974.

[NSRC90] J. Nossek, G. Seiler, T. Roska, and L. O. Chua. Cellular neural networks: Theory and circuit design. Report TUM-LNS-TR-90-7, Technische Universität München, December 1990.

[RC90] Tamás Roska and L. O. Chua. Cellular neural networks with nonlinear and delay-type template elements. In *IEEE International Workshop on Cellular Neural Networks and Their Applications, Proceedings*, pages 12–25, 1990.

[Ros88] T. Roska. Some qualitative aspects of neural computing circuits. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, pages 751–759. IEEE, 1988.

[SM81] J. C. Stoffel and J. F. Moreland. A survey of electronic techniques for pictorial reproduction. In J. C. Stoffel, editor, *Graphical and Binary Image Processing and Applications*, chapter 6.1, pages 289–316. Artech House, Dedham, MA, 1981. From IEEE Transaction on Communication, 29(12), December 1981.

[SRM91] J. Sullivan, L. Ray, and R. Miller. Design of minimum visual modulation halftone patterns. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(1):33–38, January/February 1991.

[Uli90] Robert Ulichney. *Digital Halftoning.* The MIT Press, Cambridge, MA, third edition, 1990.

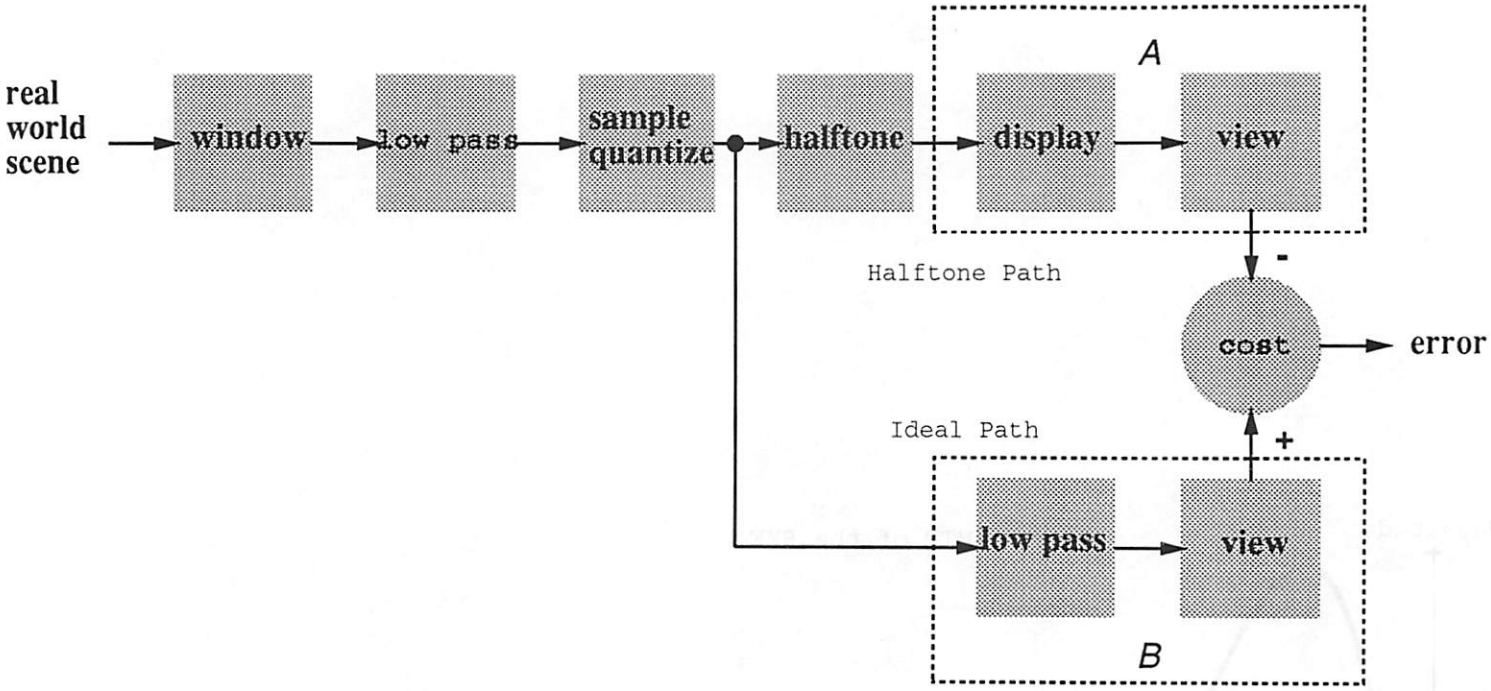

Figure 1: Flow Chart of the Whole Process

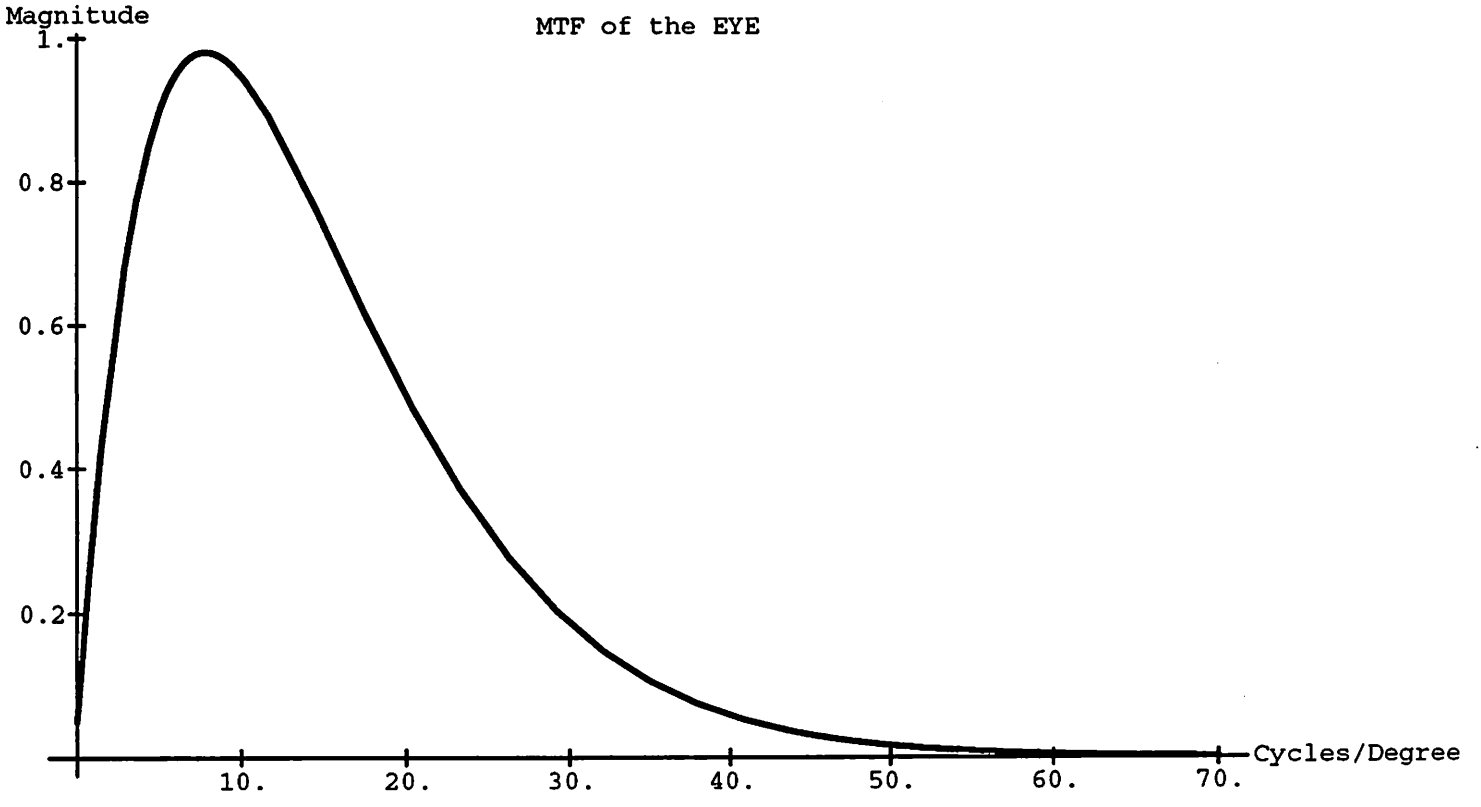

Figure 2: MTF of the Human Visual System

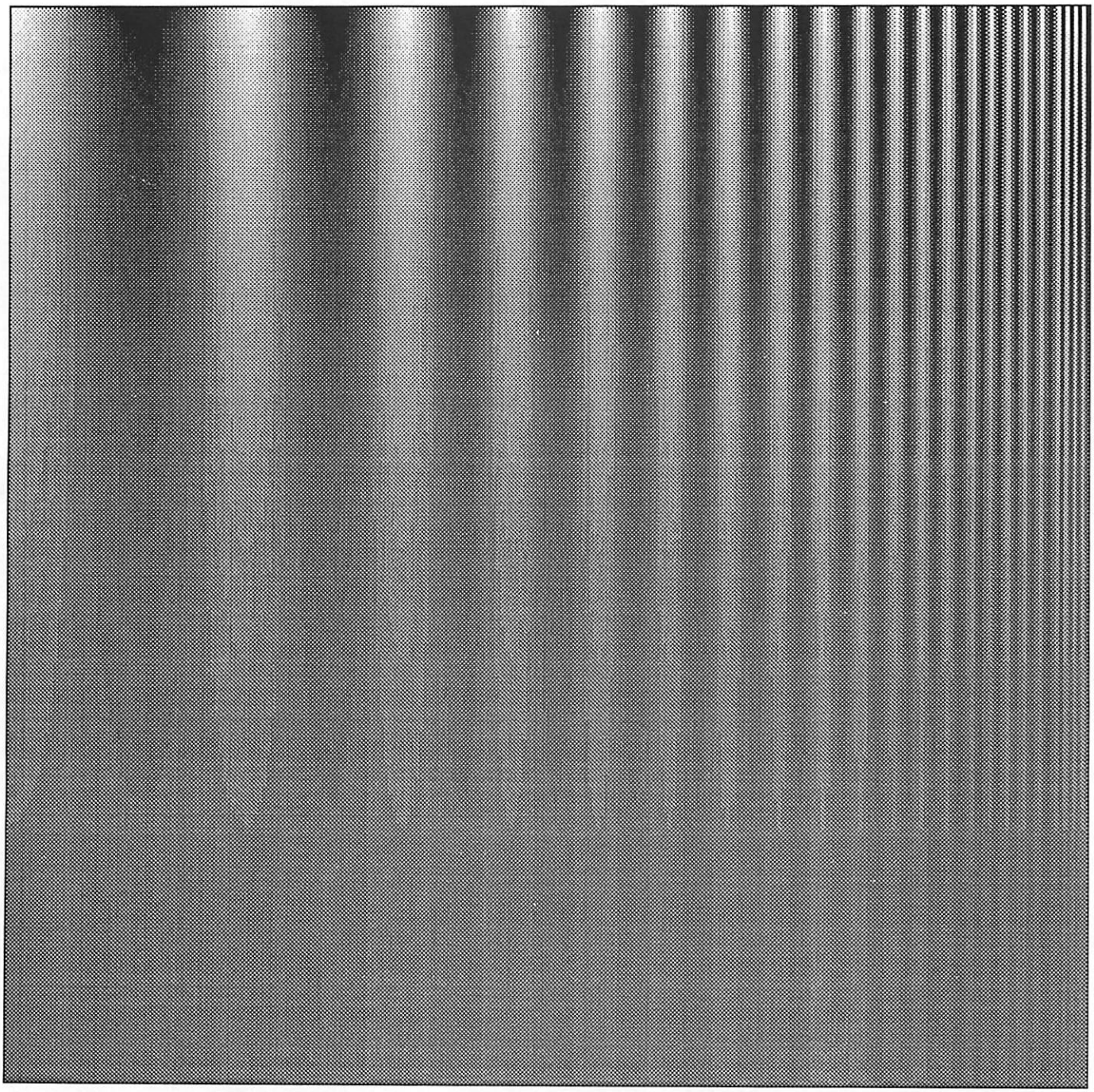

Figure 3: The Arden Chart

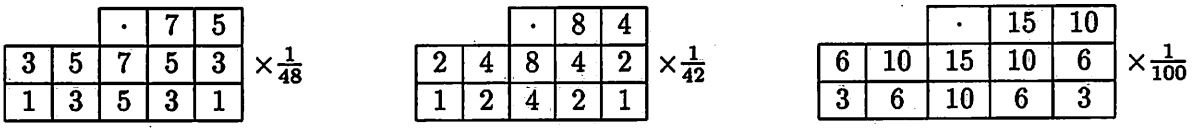

Figure 4: Error Filters of Jarvis, Stucki, and Floyd

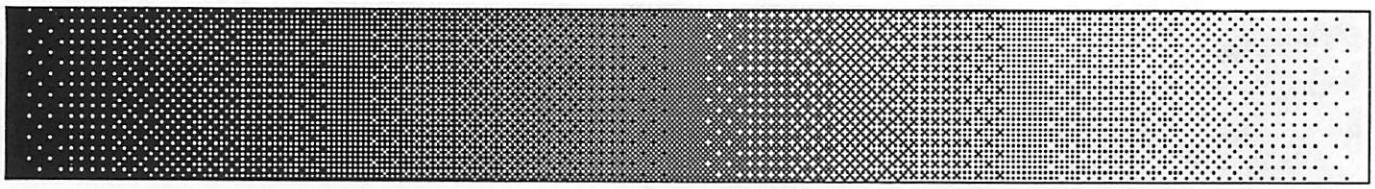

Figure 5: Ordered Dither Ramp Output

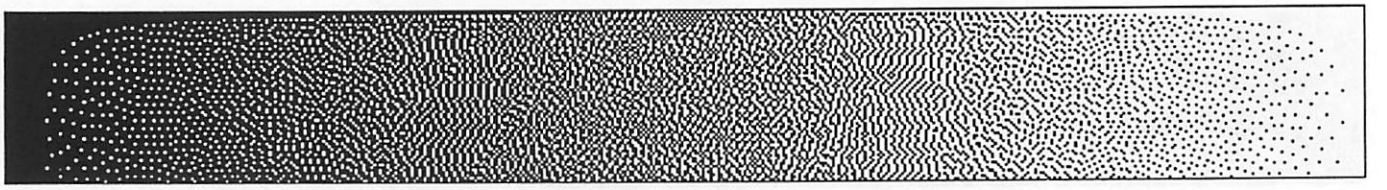

Figure 6: Jarvis Error Filter Ramp Output

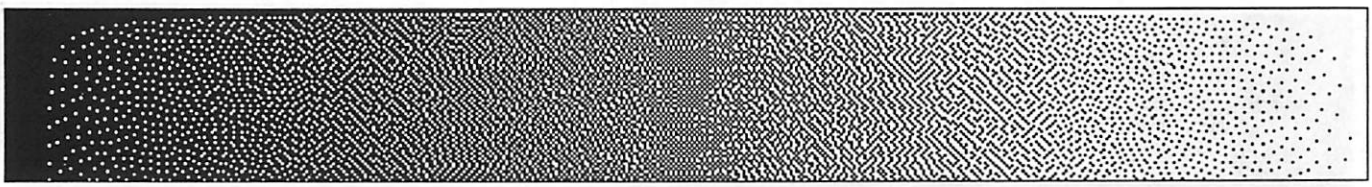

Figure 7: Stucki Error Filter Ramp Output

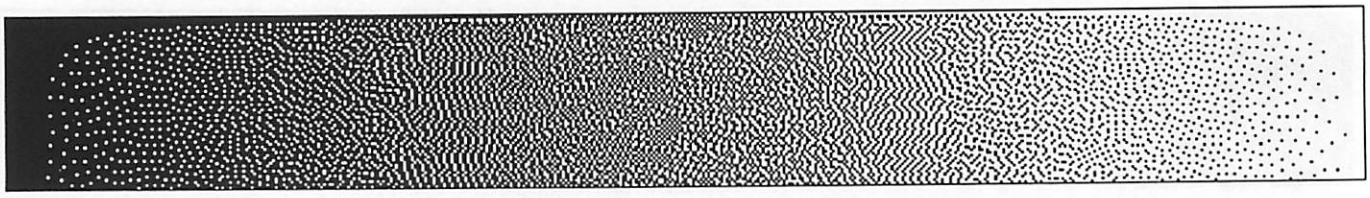

Figure 8: Floyd Error Filters Ramp Output

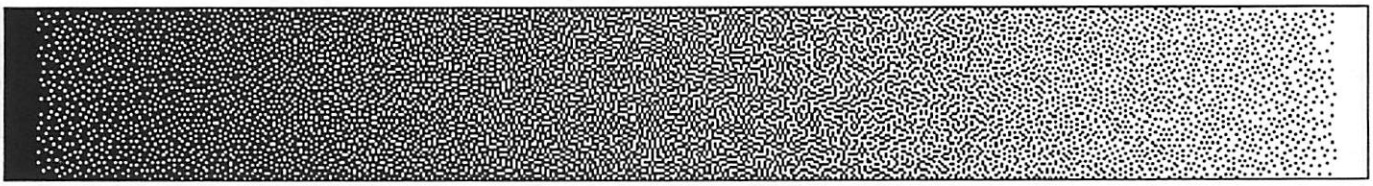

Figure 9: Template 1 Ramp Output

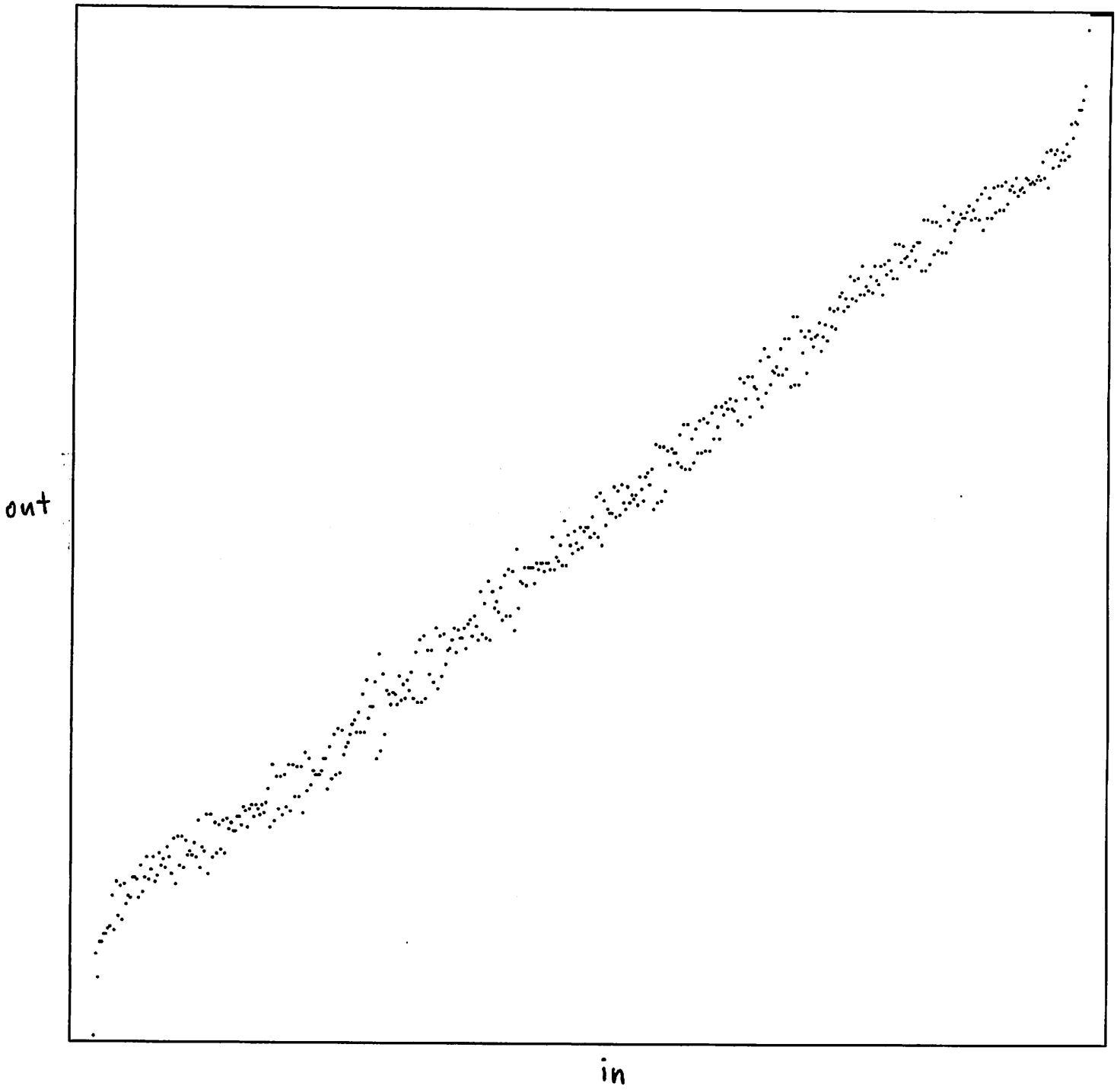

Figure 10: Template 1 DC Characteristics

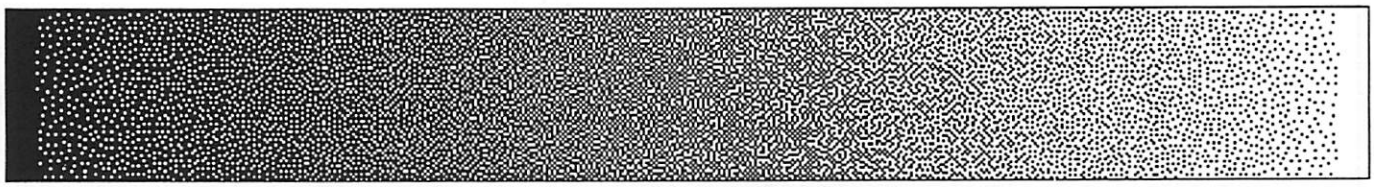

Figure 11: Template 2 and 3 Ramp Output

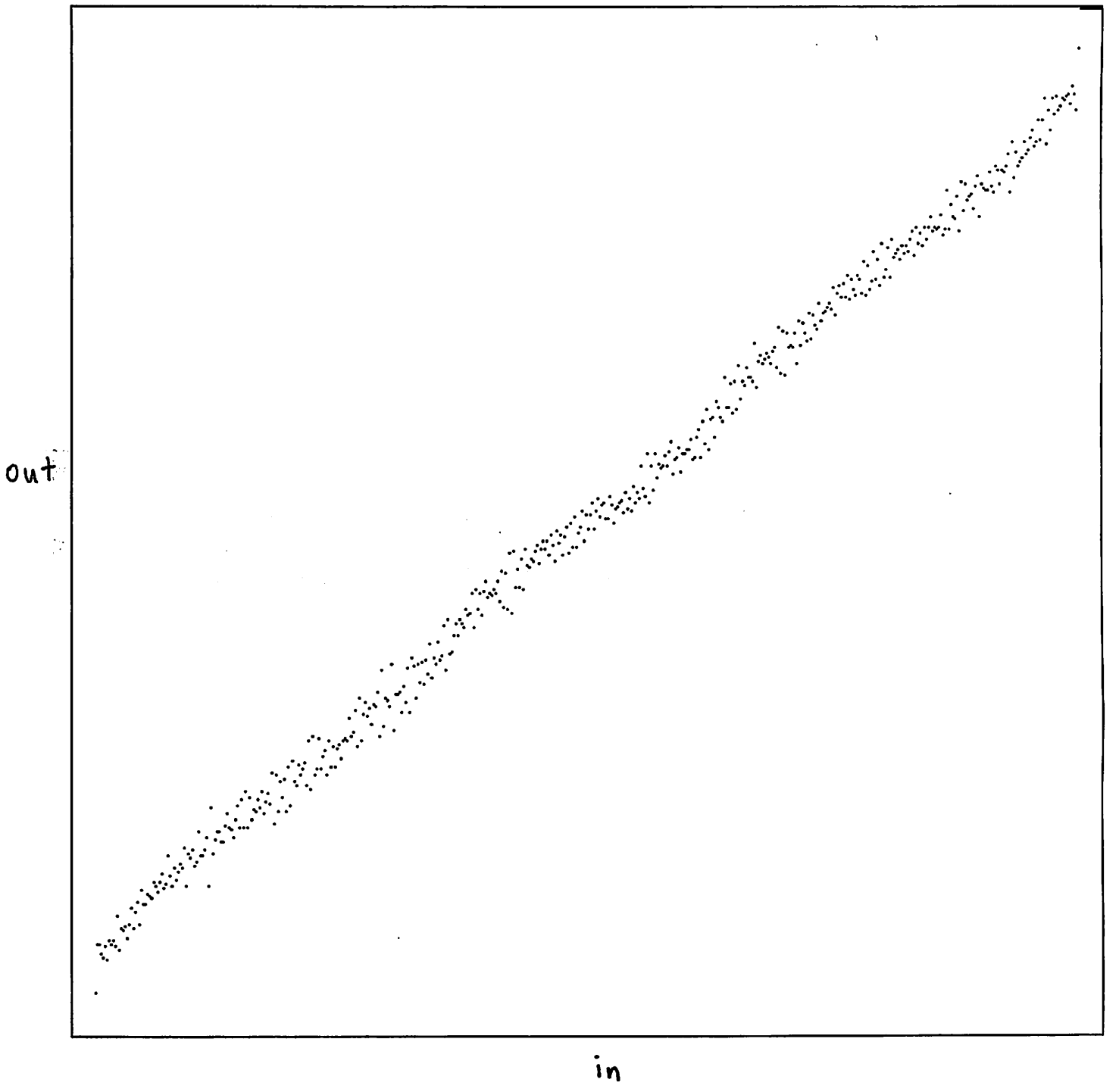

Figure 12: Template 2 and 3 DC Characteristics

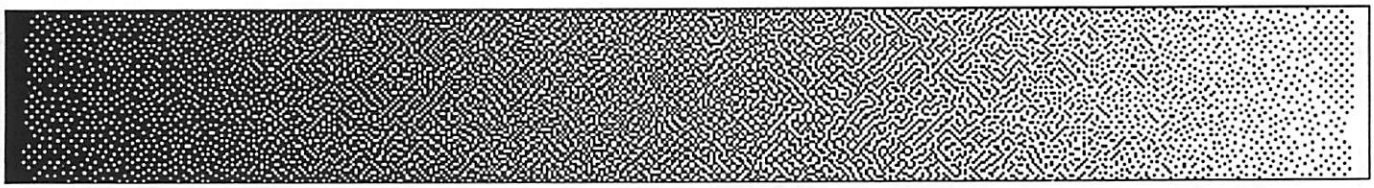

Figure 13: Template 4 Ramp Output

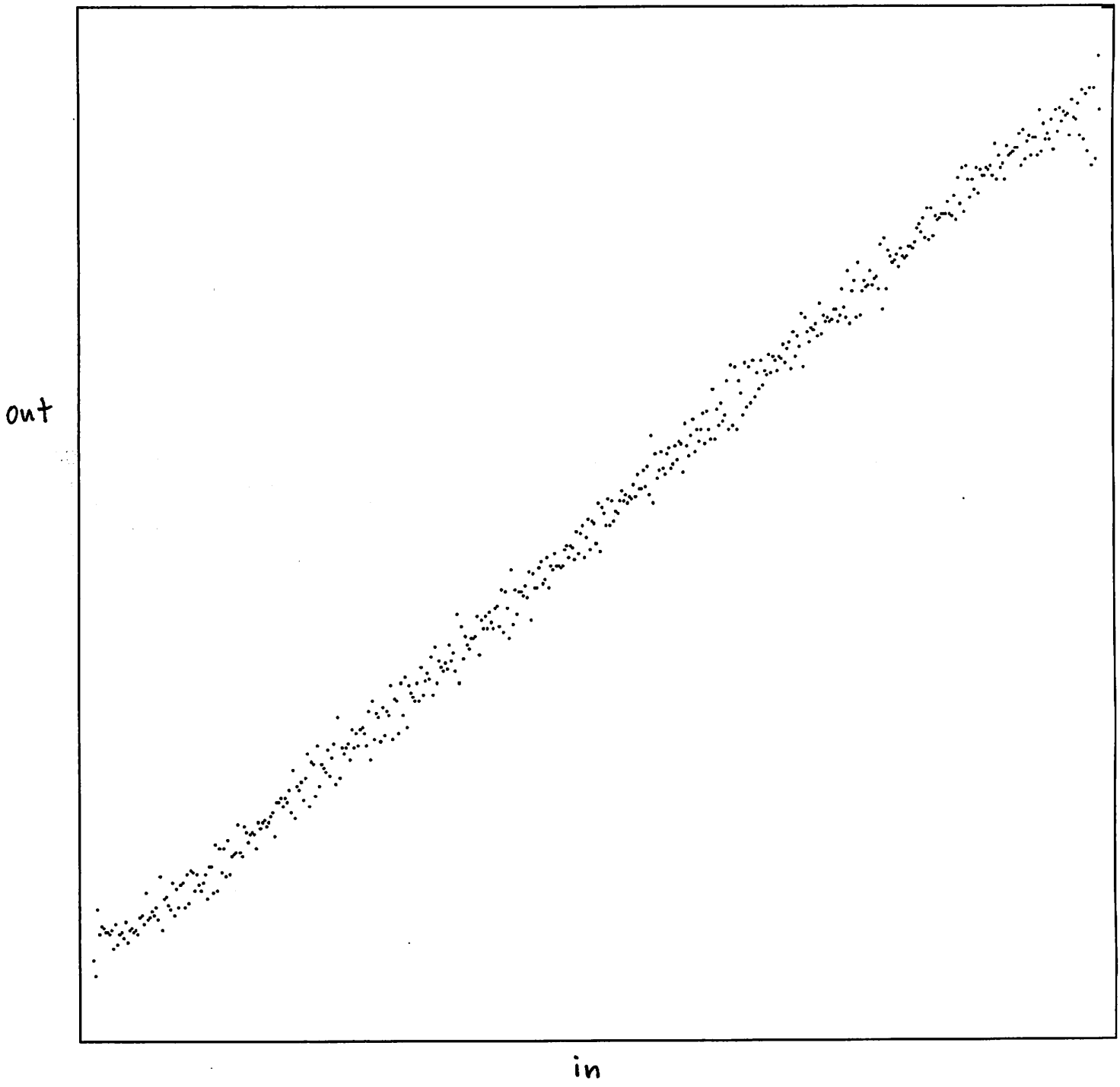

Figure 14: Template 4 DC Characteristics

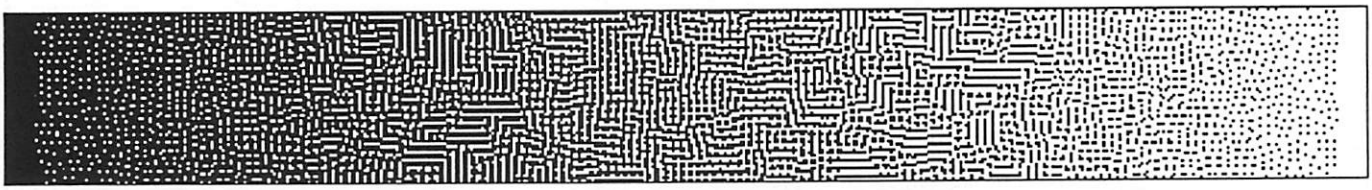

Figure 15: Template 5 Ramp Output

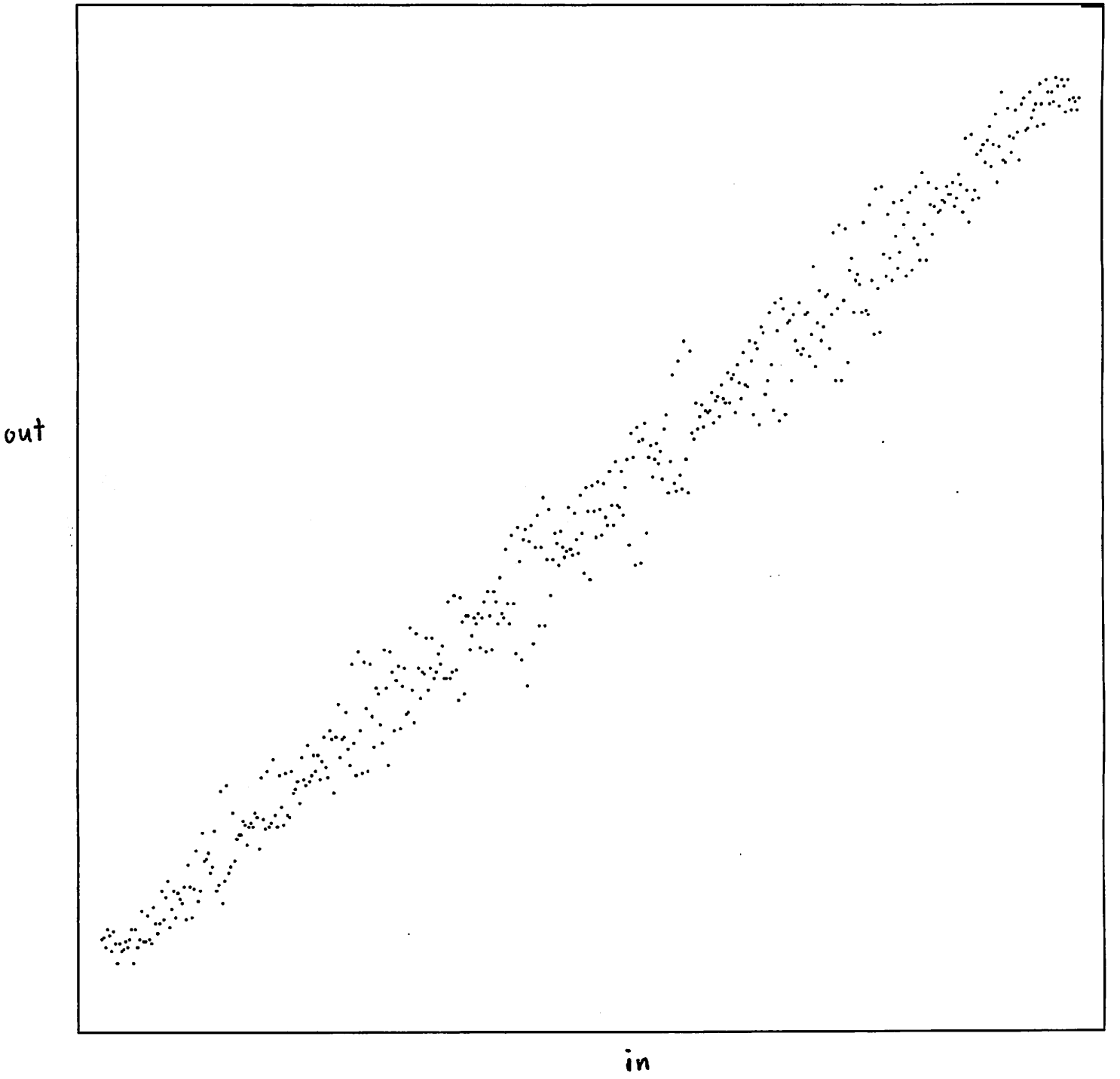

Figure 16: Template 5 DC Characteristics

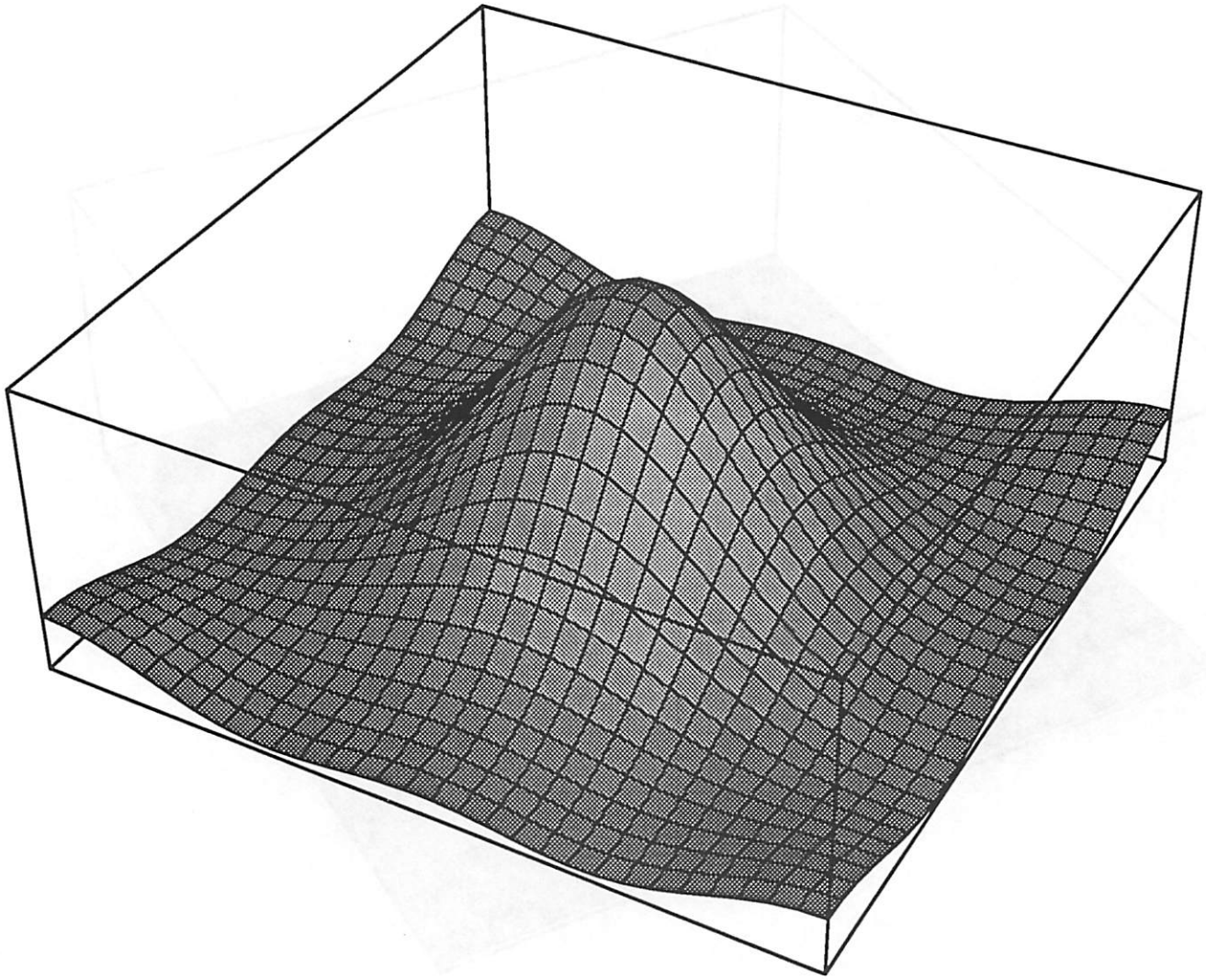

Figure 17: Template 1 Frequency Characteristics

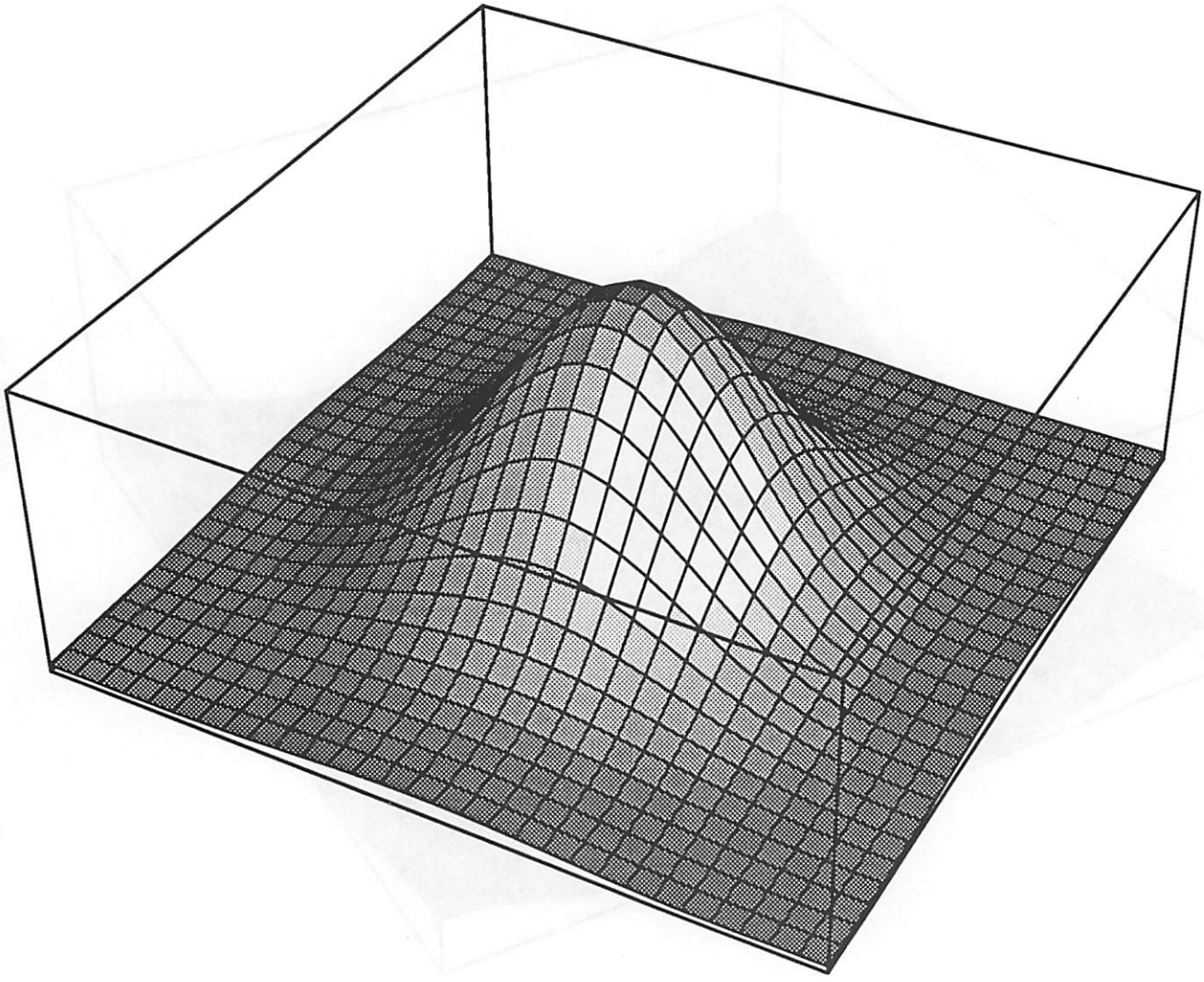

Figure 18: Template 2 Frequency Characteristics

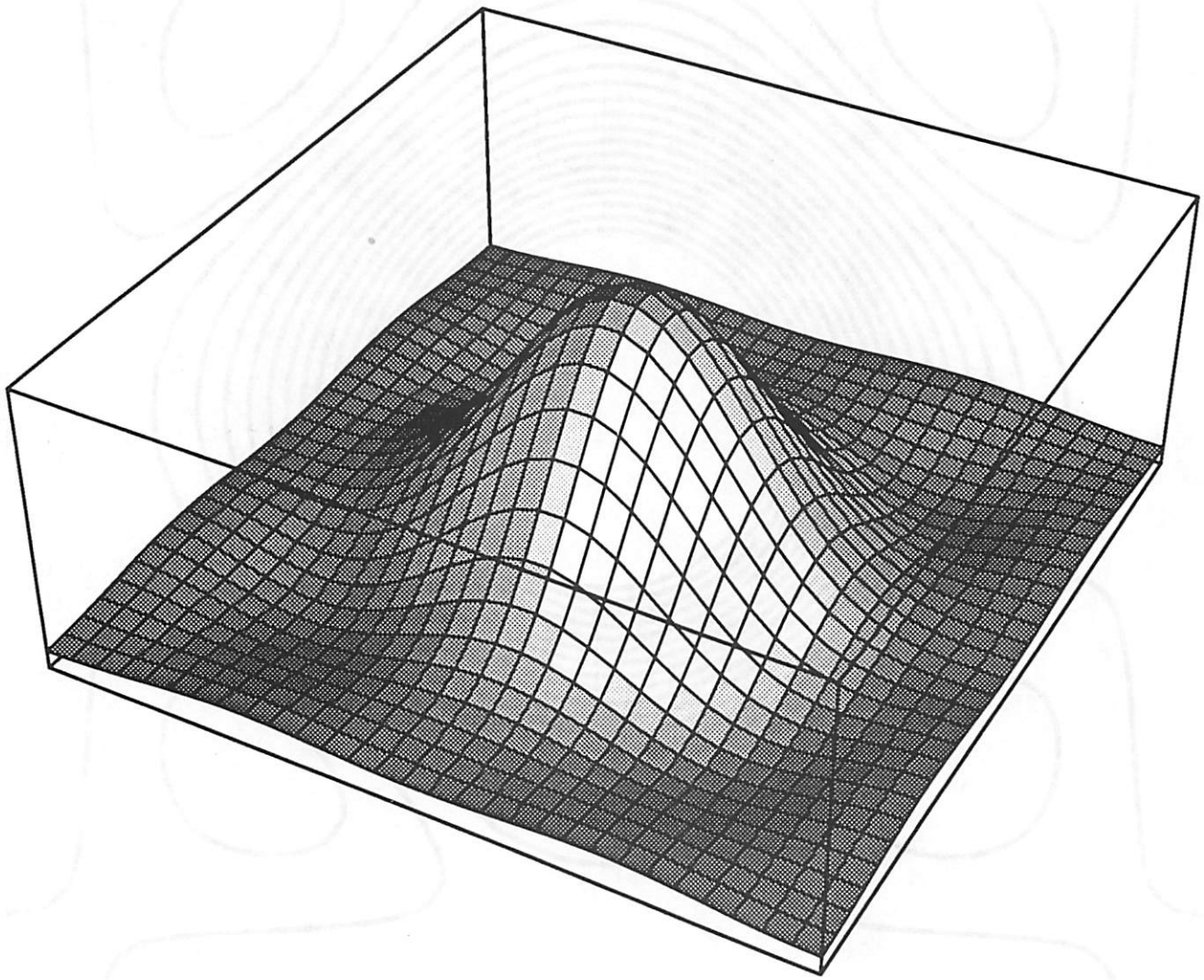

Figure 19: Template 4 Frequency Characteristics

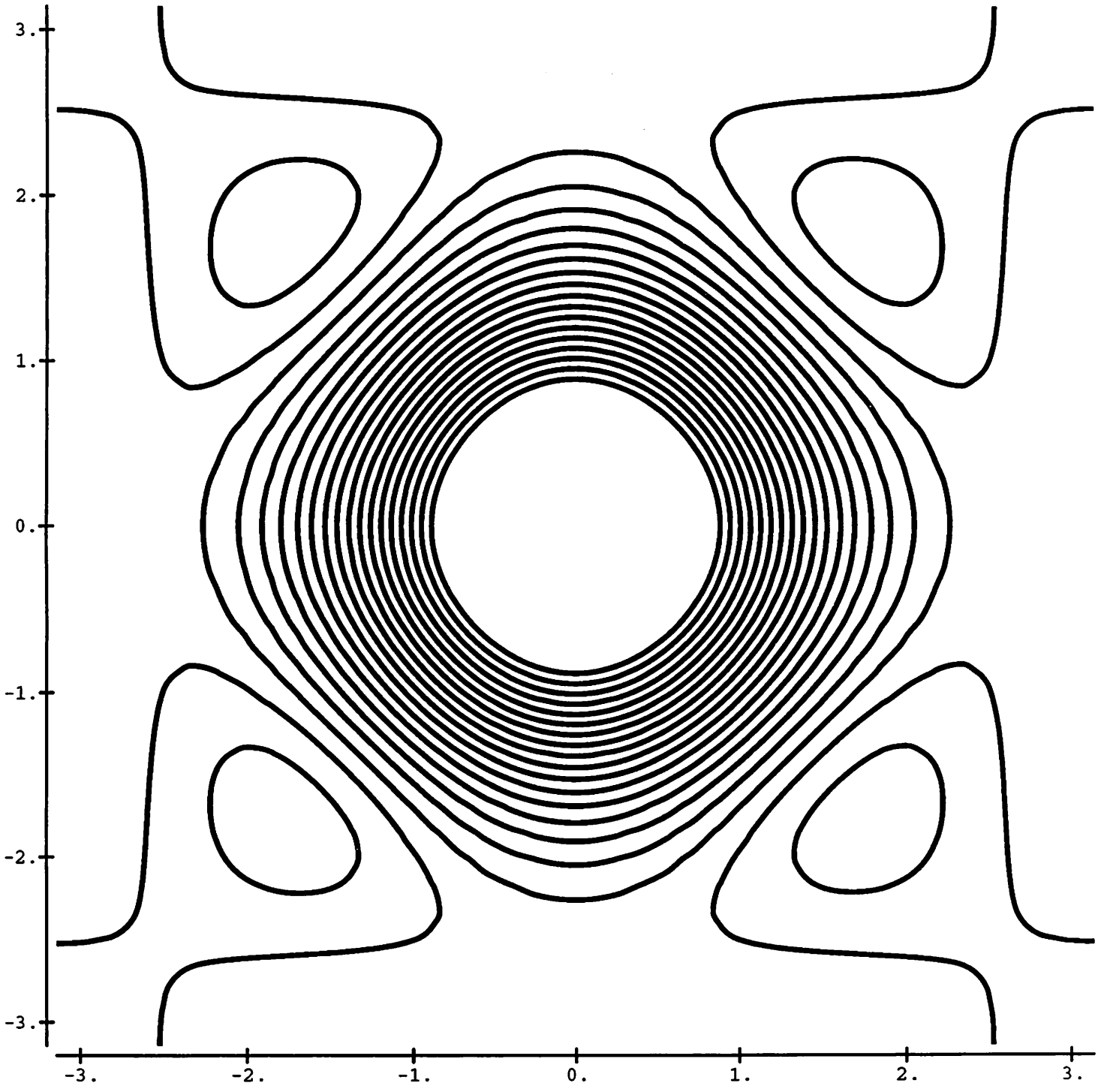

Figure 19 b.

Template 4 Frequency Characteristic Contour Plot

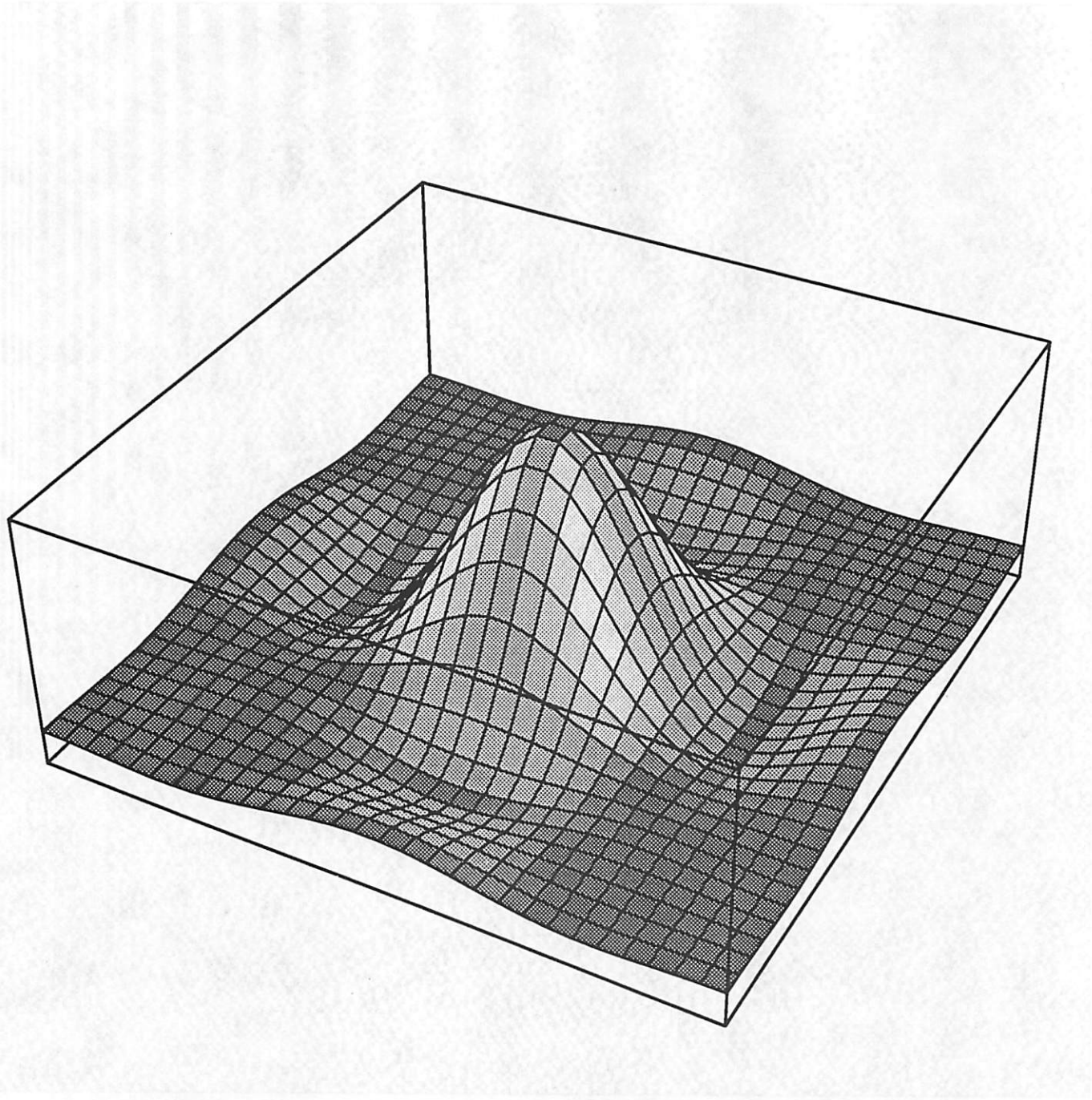

Figure 20: Template 5 Frequency Characteristics

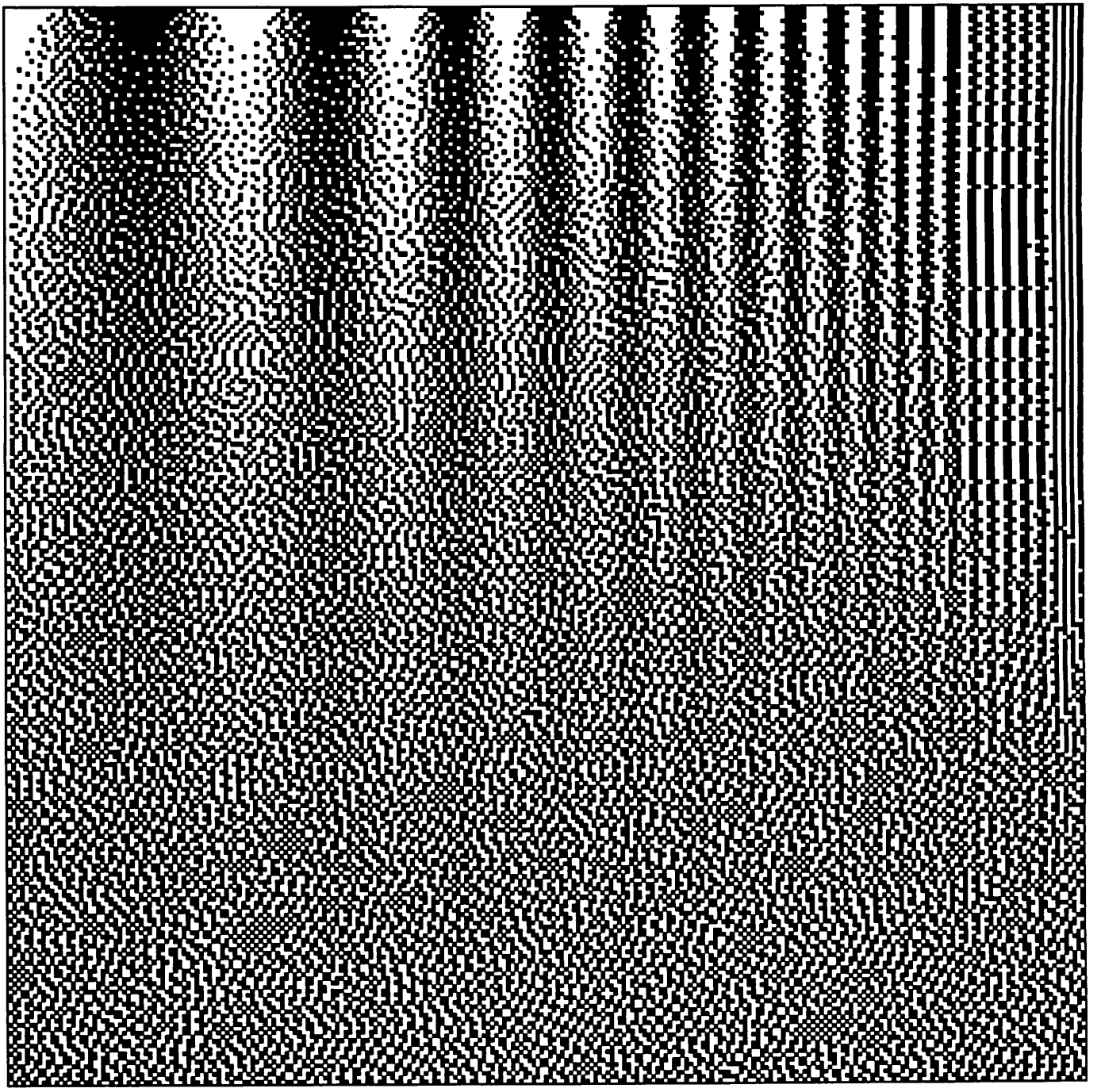

Figure 21: Jarvis Error Filter Arden Chart Output

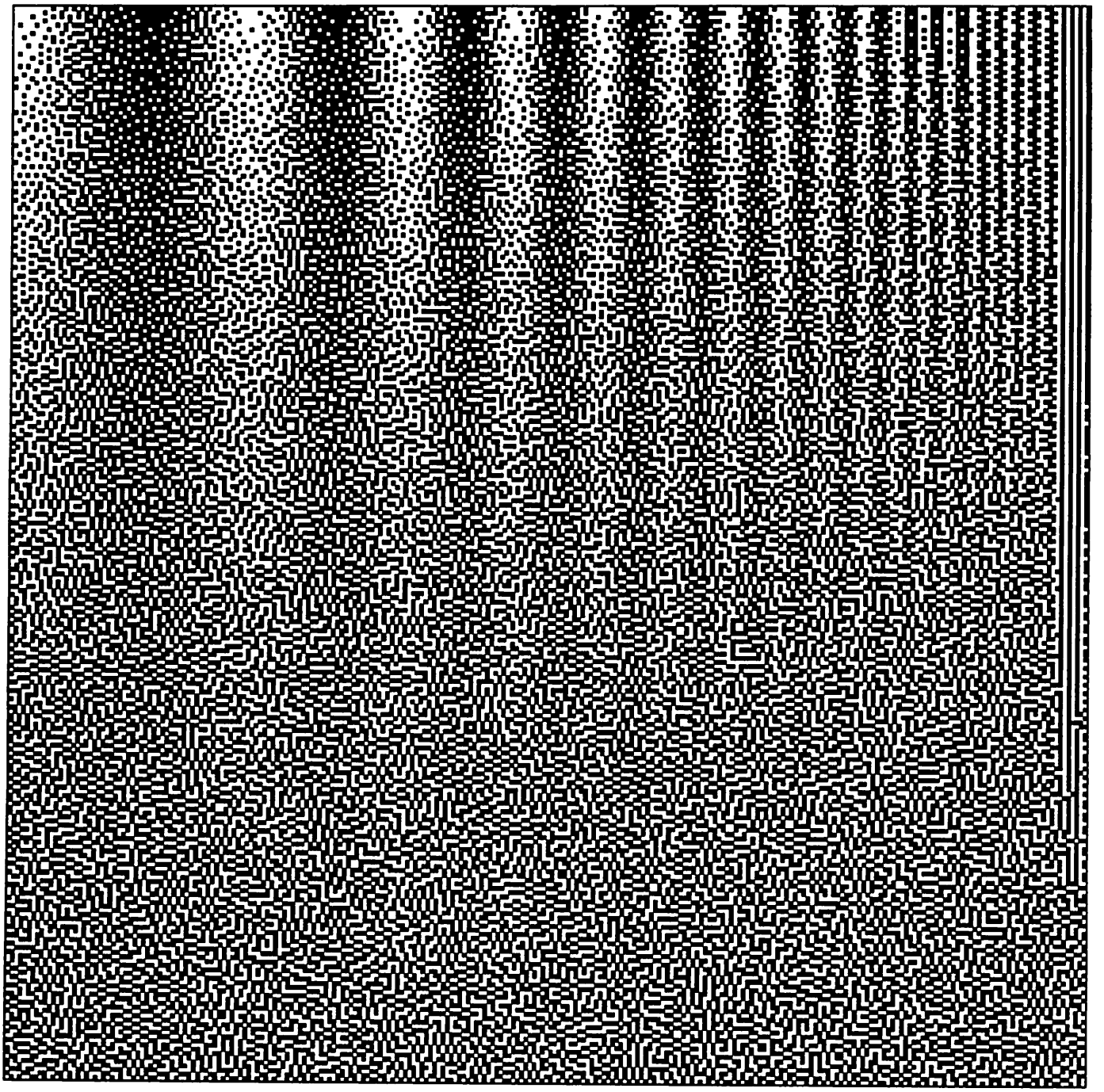

Figure 22: Template 1 Arden Chart Output

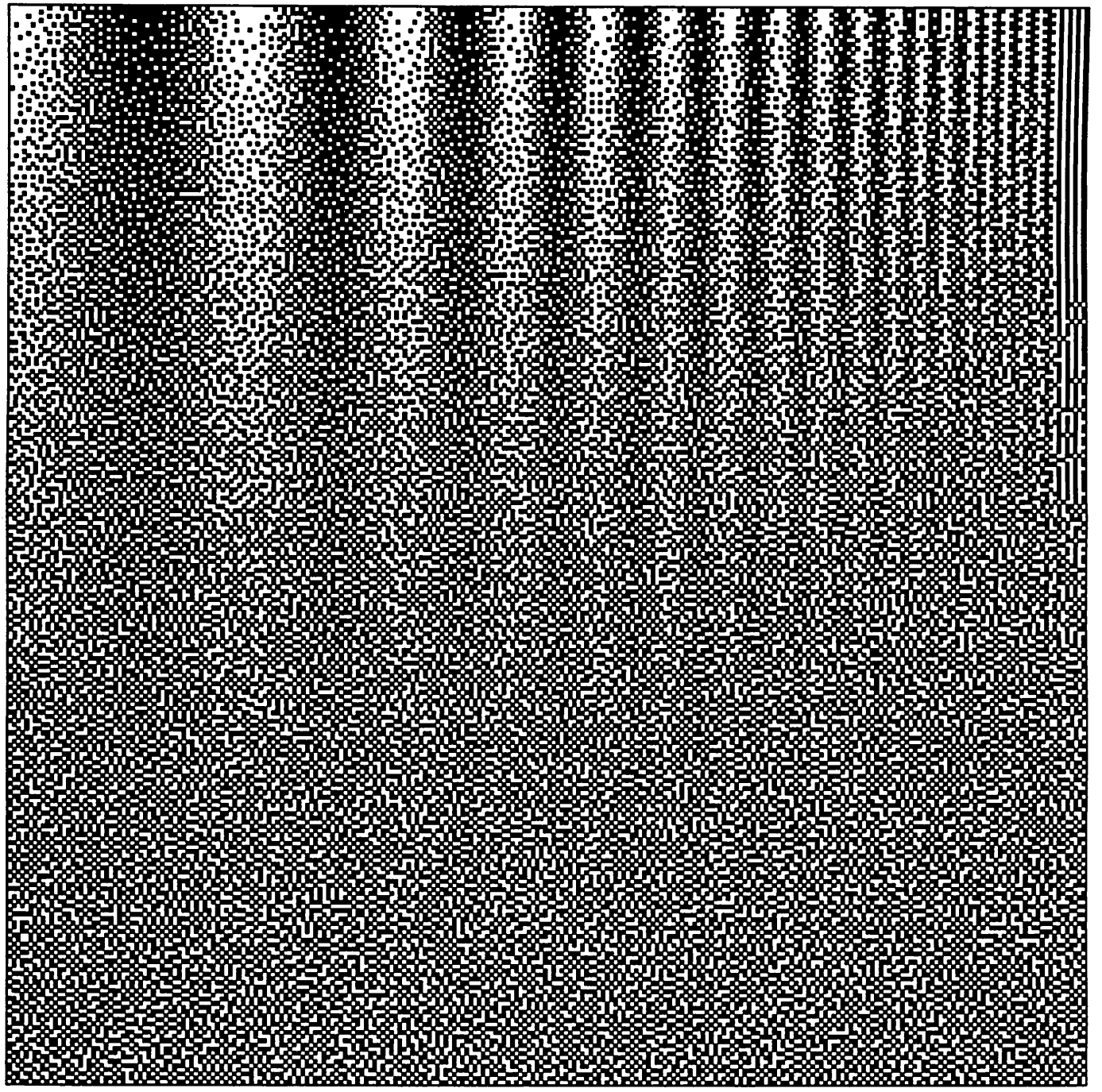

Figure 23: Template 2 Arden Chart Output

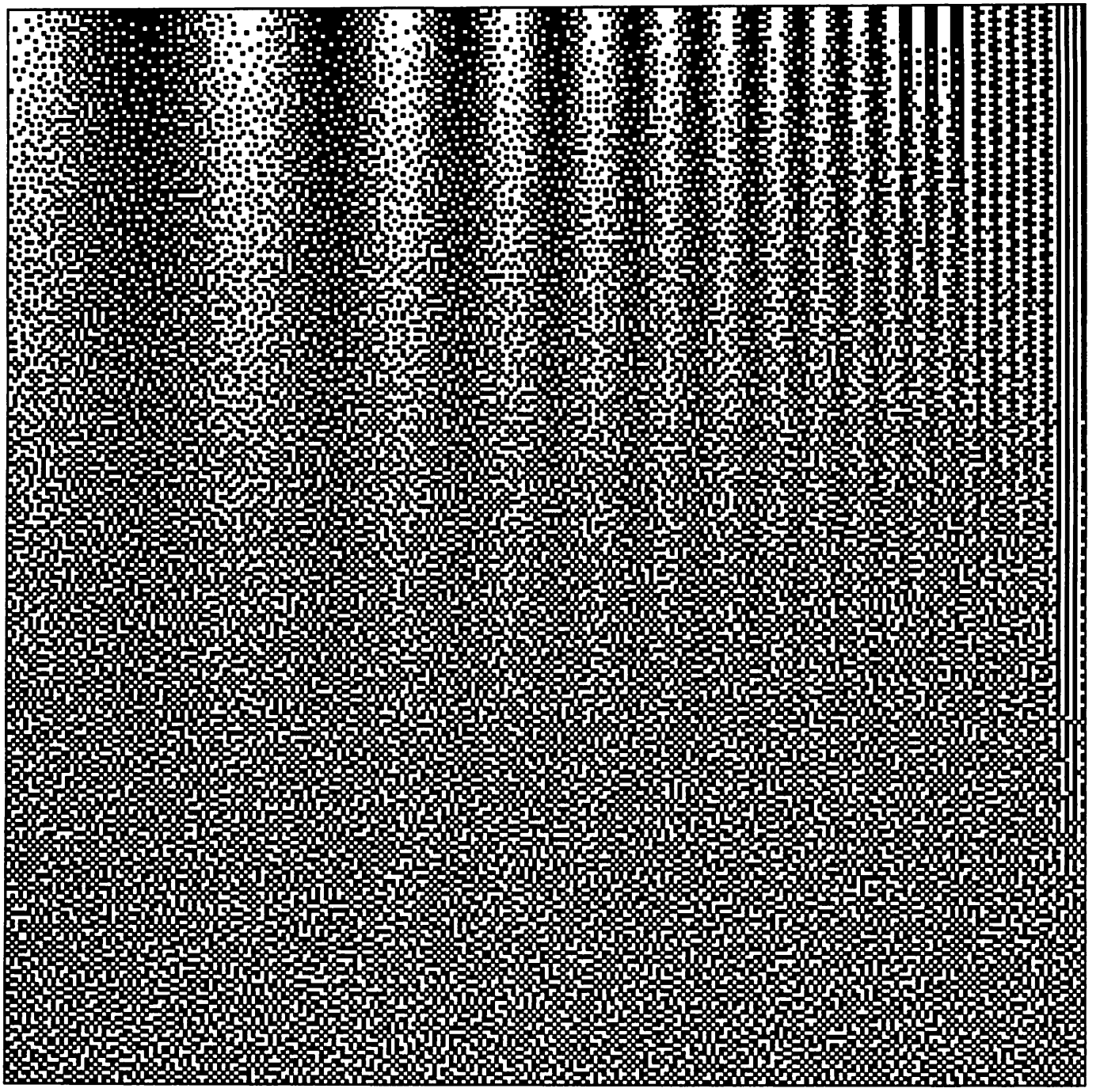

Figure 24: Template 3 Arden Chart Output

Figure 25: Original Lena Image

Figure 26: Ordered Dither Lena Output

Figure 27: Jarvis Error Filter Lena Output

Figure 28: Template 1 Lena Output

Figure 29: Template 2 Lena Output

Figure 30: Template 3 Lena Output

Figure 31: Oversharpened CNN Lena Output