BINARY CORRELATORS FOR OPTICAL COMPUTING AND PATTERN RECOGNITION

Thesis by

Fai Ho Mok

In Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

California Institute of Technology

Pasadena, California

1989

(Submitted May 25, 1989)

I dedicate this thesis to my wife, Suk.

ACKNOWLEDGEMENTS

I would like to thank my parents, whose support in my life has been most valuable. I am also deeply indebted to my wife, Suk, without whom this thesis would not be possible. It is her endless love and patience which help me through my most difficult times at Caltech.

I owe a great deal to my advisor, Professor Psaltis, who has never given up on me even when I was most unproductive. His humorous approach will always be remembered. And of course, his physical effort on the basketball court will not be forgotten either.

I would like to specially thank Dr. Jeffrey Yu who has made valuable contributions to my work. Our discussions on research, though usually resulting in an exchange of punches, have been most fruitful. I would also like to thank Dr. John Hong for his useful suggestions and encouragement.

My gratitude goes to the members of the Optical Information Processing Group . Not only have they been very helpful to my academic study, they have also added numerous happy moments to my life. Among them, I would like to thank Dr. Eung Gi Paek, who taught me valuable experimental techniques, and Alan Yamamura, who introduced me to some very fine restaurants (and also an empty wallet) in Los Angeles. I also thank Dr. Robert Snapp, Cheol Hoon Park, Scott Hudson, Ken Hsu, Mark Niefeld, and David Brady for some very useful discussions. My best wishes go to the group's basketball team. Though they have not made the playoffs of the GSC tournament, they have demonstrated other redeeming qualities: like attempting (not making) Jordan-type shots; or taking (not making) a 35-footer

when the team is down by 1 point and with 45 seconds left on the clock. May they have better luck next season.

My stay at Caltech has been made more pleasurable by the members of the Dai Yung House. They include: Kwok Cheung (House Master), Chi Ho, Peter Tong, Daniel Tang, Alex Ho, Ricky Ng, Ken Leung, and numerous other associates. Together, they gave me some of the most memorable moments in my life (I can almost see the smiling faces of our neighbors when we moved away).

The administrative help that I received from Mrs. Helen Carrier, Mrs. Odessa Myles, Mrs. Linda Dozsa, and Mrs. Su McKinley have made my graduate work considerably easier. To them I express my appreciation.

ABSTRACT

The matrix-vector multiplier is an important building block in optical information processing architectures, examples of which are correlators for pattern recognition, associative memories, and neural networks. Such architectures are most suitable for implementation by optics due to the ease in realizing dense interconnections optically. The success of the implementation partially relies on the quality of the SLM used to record the information for processing. Limited dynamic range for the representation of the data recorded is a common drawback suffered by most commercially available devices. In this thesis, the importance of the dynamic range of the device on the performance of the implementation is investigated. The effect of limited dynamic range on the signal to noise ratio, probability of error, capacity, and training of various forms of matrix-vector multipliers are addressed. Through the use of theoretical analyses, computer simulations, and optical experiments, it will be shown that a large dynamic range is not essential in most applications. Specifically, it is shown that only one bit of dynamic range, i.e. two gray levels, for the representation of each data point, results in acceptable loss in performance.

TABLE OF CONTENTS

I. Introduction

One of the most important building blocks in optical information processing architectures is the correlator. In its analog form, the correlator is composed of an input function $f(x,y)$, a reference kernel $h(x,y)$, and an output function $g(\hat{x},\hat{y})$:

$$g(\hat{x},\hat{y}) = \int\int f(x+\hat{x}, y+\hat{y})h(x,y)dxdy$$

We will assume that f and h are both real functions. The relationship between the input and the output can be described as follows: each output data point is the result of the inner-product between a shifted form of the input pattern and the reference kernel. The correlation center is the output point corresponding to the unshifted input function. (In its discrete form, the reference kernel and the output function become vectors whereas the input function and its various shifted forms can be described by a Toeplitz matrix.)

Correlators have been studied extensively in many fields (*e.g.* singal processing, communication, and pattern recognition). Besides the attractive feature that the operations needed to calculate the correlation (multiplication and addition) can be done in parallel, another reason why correlators are so popular in the field of optics is the simplicity of their implementations. The hardware requirement are a device to represent the input, a lens (for calculating Fourier transform of the input), another device to represent the kernel, a second lens (for the inverse Fourier transform), and an output detector (Figure I.1.).

The input and the kernel can be represented through various means. One of

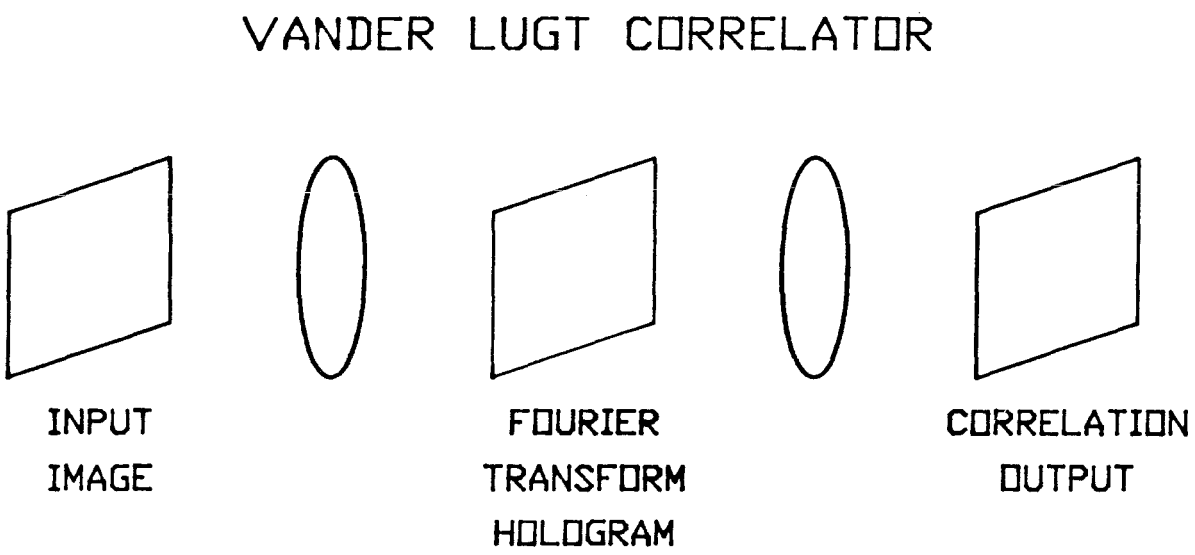


Figure I.1. A Vander Lugt Correlator.

them is through the use of spatial light modulators (SLM). A SLM is a device which modulates the incident light, *e.g.* amplitude, according to the information recorded. The performance of the implementation depends largely on the quality of the SLMs used.

Intuitively, a "good" SLM means that the modulation introduced by the SLM resembles the function recorded closely. If this is the case, we expect the real correlation output to resemble the correlation we try to implement. By the same token, a "poor" SLM will yield a correlation output which we did not plan for. It is, however, important to note that typically, the optical implementation of correlators are aimed at accomplishing certain goals (*e.g.* pattern recognition). It is therefore unimportant whether the correlations come out to be exact so long as the desired goal is accomplished. For example, a correlator is often used to do template matching. In this case, the kernel is the template. When a pattern is presented at the input plane, the cross-correlation between the input pattern and the template will be obtained at the output plane. If the input pattern resembles the template, the cross- correlation will contain a strong peak. By the presence (or absence) of such a peak, patterns can be classified. It is evident that the essential element in such a system is the presence or absence of a correlation peak but not the fidelity with which the correlation function is computed. Thus, so long as the kernel and the input correlate strongly when so desired, it does not really matter how close do they represent the respective functions. This observation allows us to use effectively, non-ideal devices.

Currently, commercially available SLMs suffer from various drawbacks. Some

of them are: low resolution, poor optical quality, and few gray levels per pixel. All of these factors affect the actual correlation output. At least one of them, few gray levels, however, does not degrade the "detectability" of correlation peaks (should they be present) significantly. The goal of my thesis is to demonstrate that this is indeed the case.

The detectability of correlation peaks is a quality measured by comparision: the strength of correlation peaks can only be determined by also measuring their background. In this sense, the signal to side-lobe ratio, SSR, is a useful figure of merit to quantify the perfomance of various correlators. It can be shown that the matched filter, MF, ($h(x,y) = f(x,y)$) is the optimum linear filter in terms of SSR if f a white Gaussian random process (Figure I.2.).

Suppose we want to implement the MF. If the kernel is affected by limited dynamic range (another way of saying few gray levels per pixel) of the SLM that it is written onto, we expect the SSR to be less than that of the genuine MF. We also expect that as the dynamic range is reduced, the SSR is also degraded. The worst performance is expected when the SLM has only a one-bit (two gray levels) dynamic range. By this virtue, the model that I will use the most in this thesis is a device with only two gray levels. The results on perfomance of various correlators using this model should provide useful lower bounds for correlators implemented with devices having larger but still limited dynamic range. By comparing the results produced by my model and those produced by devices with infinitely large dynamic ranges, the importance of dynamic range on information processing of the cases I considered can be realized.

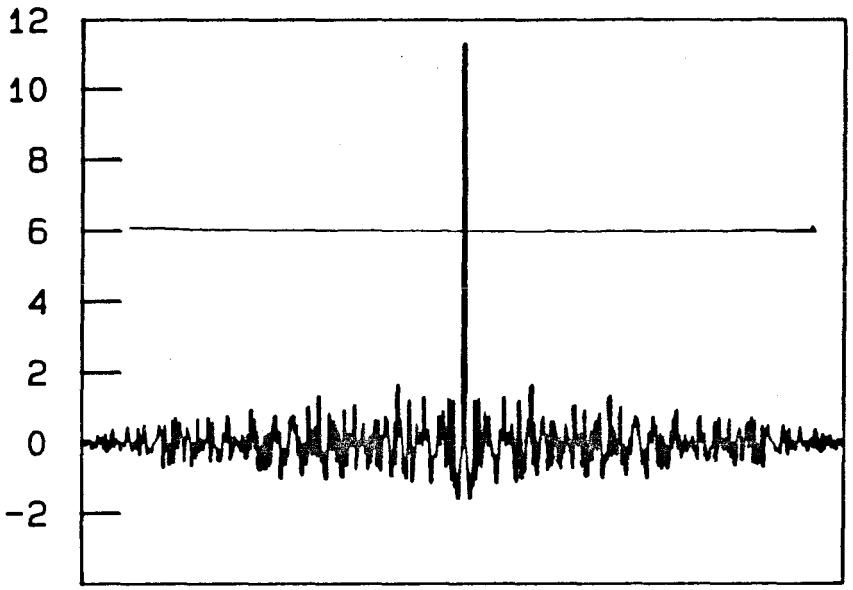


Figure I.2. The autocorrelation of an independent Gaussian random sequence.

In Chapter II, we investigate the degradation in perfomance of various forms of the MF with only one bit of dynamic range. The detectability of the correlation peak with and without the influence of noise are examined. Specifically, the SSR and the SNR (signal to noise ratio) are calculated. The SSR and the SNR of the MF can be shown to be proportional to N^2, the number of pixels in h or f. Comparing to the MF, the SSR and the SNR of the one-bit correlators considered are shown to decrease by constant factors. In other words, the correlation peaks of all the one-bit correlators considered will be detectable given enough information is contained in the patterns to be identified. Another standard problem in pattern recognition is the two-category classification. In this problem, patterns are assigned to one of two categories by comparing the inner-products (a special case of correlation) of the patterns with the two representation patterns of the two categories —— a pattern is assigned to one category if its inner-product with that category's representation pattern is greater than that of the other category (Figure I.3.).

Matched filters (one for each category) are the optimum representation patterns in terms of probabiliy of error if the patterns are white Gaussian random processes. Respective qualitative treatment of the 1-bit correlators are given. It is shown that the general behaviors of the MF and the 1-bit correlators are the same as the number of pixels increases, *i.e.* the probability of errors decreases.

Suppose the features (in forms of vectors) of the patterns belonging to two different classes are known. Then pattern classifers using these features can be constructed. One way to construct such a classifier is to build correlation filters out of a linear combination of the known features. Patterns can be identified by computing the inner-products of the patterns and the filters. (The classifier can be

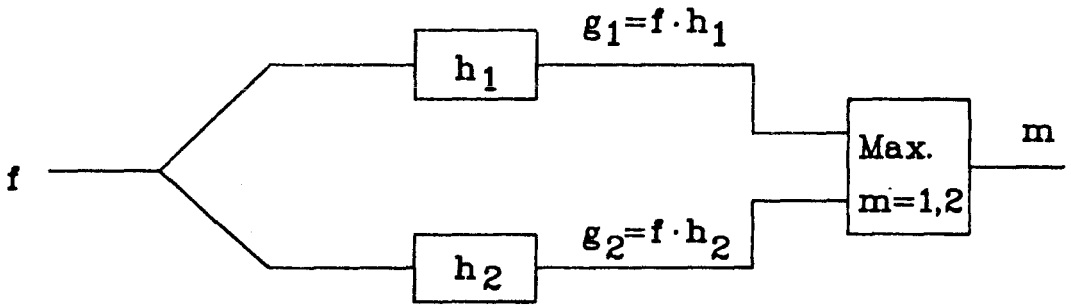


Figure I.3. A two-category classifier.

made insensitive to translations in inputs if the correlations are computed.) The appropriate figure of merit in this case is the number of independent features that the filters can sustain given a certain number of pixels without exceeding a certain probability of error. We refer to this figure of merit as the capacity of correlators. The quantitative value of the capacity provides a good idea for the number of patterns that correlators can handle. When these correlators are implemented on devices with limited dynamic range, we expect a decrease in capacity. In Chapter III, I derived the capacity of correlators with a one-bit dynamic range. The capacity of correlators with an infinite dynamic range are also derived for comparison. It is shown that the loss in capacity due to the two gray levels allowed is only a constant factor. (An interesting way to look at this type of two-category classifier is to view them as a one-bit holographic associative memory.)

Correlators made of linear combinations of features is only one way to construct a multiple-pattern two-category classifier. Another way to obtain a multiple-pattern two-category classifier is to train the kernel by the patterns to be classified. A number of training algorithms were developed in the past years. A famous example is the perceptron (Figure I.4.) developed in the '60s.

However, most of these algorithms implicitly assume that the number of gray levels is inexhaustible. Though this assumption is good when the algorithms are implemented in digital computers, it falls apart when the implementations are carried out by optics. Depending on the architecture of the implementations, the degree of effectiveness of different algorithms suffered from limited dynamic ranges varies — while the original algorithms may always succeed in finding a kernel that classifies all the training patterns correctly, the modified algorithms due to limited dynamic

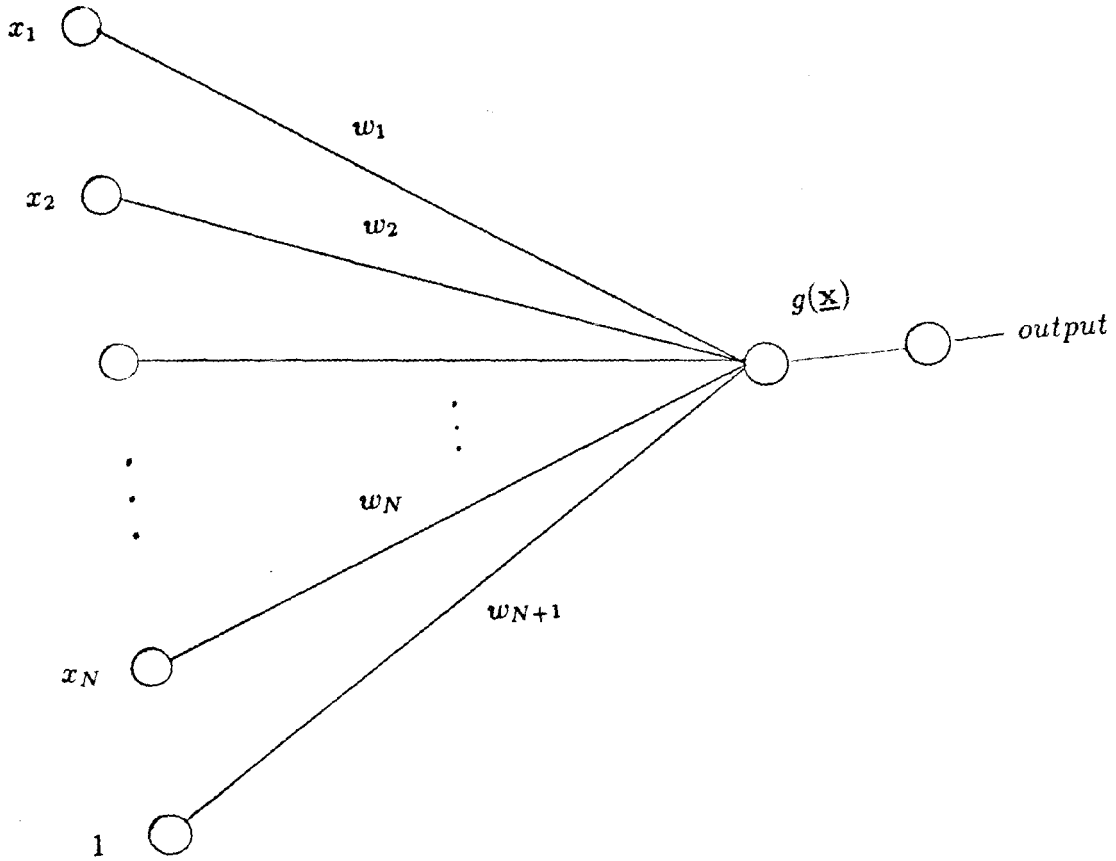


Figure I.4. A perceptron.

ranges may not. In Chapter IV, the effects of quantizing the training kernel of various algorithms are examined. In particular, the dynamics of the training procedures are studied. The dependence of the effectiveness of different algorithms are investigated. The results are quantified by comparing the probability of success in training the kernels of different algorithms. The average time required to arrive at proper kernels are also compared.

In Chapter V, optical experimental results of the implementations of various correlators on binary devices are presented. The experiments include the implementation of the matched filter on a magneto-optic device (MOD, a binary device); a real-time programmable correlator built with an acousto-optic device (AOD), a MOD, and a charge coupuled device (CCD); and a rotation invariant filter realized by a MOD. The experimental results agree with the theoretical results obtained in the previous chapters.

References for Chapter I

1. N. Nilsson, *Learning Machines: Foundations of Trainable Pattern-Classifying Systems*, McGraw-Hill, New York (1965).
2. M. Bongard, *Pattern Recognition*, Spartan Books, Washington, D.C. (1970).
3. R. Duda and P. Hart, *Pattern Classification and Scene Analysis*, John Wiley and Sons, New York (1973).
4. D. Rumelhart, J. McClelland, and the PDP Research Group, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Vol.1: Foundations*, The MIT Press, Massachusetts (1986).
5. J.J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities," *Proc. Natl. Acad. Sci.,* USA, vol. 79, pp.2554-2558 (1983).
5. D.Casasent, "Coherent Optical Pattern Recognition," *Proc. IEEE*, **67**, 813 (1979).
6. G. Turin,"An Introduction to Matched Filters," *IRE Trans. Inform. Theory*, **IT-6**, 311 (1960).
7. A. Oppenheim, "The Importance of Phase in Signals," *Proc. IEEE*, **69**, 529 (1981).
8. T. Huang, and J. Burnett, "The Importance of Phase in Image Processing Filters," *IEEE Trans. Acoust., Speech Signal Processing*, **ASSP-23**, 167 (1979).
9. B. Cooper, "Correlators With Two-Bit Quantization," *Aust J. of Phys,* **23**, 521 (1969). H. Brown, and B. Markevitch, "Application of the Liquid Crystal Light Valve to Real-Time Optical Data Processing," *Opt. Eng.*, **17**, 371 (1978).
10. B. Horowitz and F. Corbett, "The PROM — Theory and Applications for the

Pockels Readout Optical Modulator," *Opt. Eng.*, **17**, 353 (1978).

11. C. Warde, A. Weiss, A. Fisher, and J. Thachara, "Optical Information Processing Characteristics of the Microchannel Spatial Light Modulator," *Appl. Opt.,* **20**, 2066 (1981).
12. W. Colburn and E. Tompkins, "Improved Thermoplastic Photoconductor Devices for Holographic Recording," *Appl. Opt.*, **13**, 2934 (1974).
13. A. Vander Lugt, "Signal Detection by Complex Spatial Filtering," *IEEE Trans Inf Th,* **IT-10**, 139 (1964).
14. T. Cover, "Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition," *IEEE Trans. Elec. Comp.,* **EC-14**, 326 (1965).
15. A. Oppenheim and R. Schafer, *Digital Signal Processing*, Prentice-Hall, New Jersey (1975).
16. M. Minsky and S. Papert, *Perceptrons: An Introduction to Computational Geometry,* MIT Press, Cambridge, Mass. (1969).

II. Binary Correlation Filters

II.0. Introduction

A correlator computes the correlation of an input pattern $f(x,y)$ and a reference pattern $h(x,y)$. The cross-correlation $g(x,y)$ is given as

$$g(x,y) = \int\int f(\xi + x, \eta + y) h(\xi, \eta) d\xi d\eta. \qquad (II.0.1.)$$

Here, we assume h and f are both real functions. When $h(x,y)$ is identical to $f(x,y)$, we say that the input is matched. It can be shown, by invoking the Schwartz inequality, that the center value of the correlation output, $g(0,0)$, of a matched filter is never less than any side-lobe value. Thus, matched filtering can be applied to pattern recognition.

To measure the performance of these filters, various figures of merit can be used:

a. the processing gain, defined as the ratio of the energy level of the correlation peak to the average energy level of the side-lobe. It can be shown that the processing gain of a matched filter is proportional to the space bandwidth product (i.e., number of pixels in the discrete case).

b. the signal-to-noise ratio, defined as the ratio of the energy level of the correlation peak to the average energy level of the side-lobe, under the condition that the input pattern is contaminated by additive gaussian white noise. It can be shown that its matched filter is the linear filter which has the highest signal-to-noise ratio.

c. the probability of error, defined as the probability that a misclassification may occur.

In this chapter, we will investigate the effect on pattern identification due to thresholding the matched filter at the input plane; thresholding the input pattern; and thresholding the matched filter at the fourier plane. Through the computations of the three figures of merit mentioned in the previous paragraph, the performance of these correlators will be investigated. Their relative performances, compared to that of the matched filter, will also be examined.

II.1. Image Plane Binary Matched Filter

In this section, we analyze the performance (subjected to the figures of merit defined earlier) when the reference filter is thresholded. By thresholding the filter, we mean that the function of the filter is mapped to a function having only binary and bipolar values, i.e., +1 and -1. Specifically, the rule of obtaining the reference filter can be given as

$$h(x,y) = \beta[f(x,y)] \qquad (II.1.0.1.)$$

where

$$\beta[x] = \begin{cases} +1 & \text{if } x \geq 0; \\ -1 & \text{if } x < 0. \end{cases}$$

Note that this algorithm preserves the zero-crossings information of the filter. For future reference, we denote this modified form of matched filter as the binary (and bipolar to be specific) image space filter or simply as BIMF.

Suppose, without loss of generality, the reference origin of the pattern to be

matched coincides with the origin of the input plane. Thus, a correlation peak at the output plane origin is expected. To demonstrate that this requirement is satisfied, consider eq.(II.0.1.) in the case when the reference is thresholded, consider eq.(II.0.1.). Substituting eq.(II.1.0.1) into eq.(II.0.1.), we have

$$g(x,y) = \int\int f(\xi + x, \eta + y)\beta[f(\xi,\eta)]\, d\xi d\eta \qquad (II.1.0.2.)$$

By invoking the triangle inequality, we obtain

$$g(x,y) \leq \int\int \Big|f(\xi + x, \eta + y)\Big|\Big|\beta[f(\xi,\eta)]\Big| d\xi d\eta. \qquad (II.1.0.3.)$$

Since the reference filter can only assume values of +1 and -1, hence $|\beta[f(\xi,\eta)]| = 1$. Thus

$$\begin{aligned} g(x,y) &\leq \int\int \Big|f(\xi + x, \eta + y)\Big|\Big| d\xi d\eta \\ &= \int\int \Big|f(\xi,\eta)\Big| d\xi d\eta \\ &= g(0,0) \end{aligned} \qquad (II.1.0.4.)$$

In other words, the correlation level at the center is higher then any other side-lobe level.

II.1.1. Processing Gain of BIMF

Even though we have shown that a correlation peak exists, this result alone does not qualify a BIMF as a pattern detector. One reason is that we have no knowledge of how outstanding this correlation peak is. An objective measure of the

"outstandingness" of the correlation peak is the magnitude of the relative size of the peak to the side-lobe values, or equivalently, the processing gain of the correlator. For ease of computation, we will consider the discrete analog of a BIMF correlator. Let $f(i,j)$, $h(i,j)$, and $g(i,j)$ be the sampled versions of $f(x,y)$, $h(x,y)$, and $g(x,y)$, respectively. In its discrete form, the correlation operation becomes

$$g(i',j') = \sum_{i}^{N} \sum_{j}^{N} f(i+i', j+j')h(i,j) \qquad (II.1.1.1.)$$

where N^2 is the number of samples in f and h and

$$h(i,j) = \beta[f(i,j)]. \qquad (II.1.1.2.)$$

Assume that $f(i,j)$ is a sample realization of a discrete sequence of independent, identically distributed gaussian random variables with zero mean and variance σ_f^2.

Now, the definition of the processing gain, PG, is given as

$$PG = \frac{E^2[g(0,0)]}{var[g(i',j']}, \qquad (II.1.1.3.)$$

where $E[.]$ stands for the expected value operator and $var[.]$ stands for the variance operator. The expected value of $g(0,0)$ can be obtained by considering eq.(II.1.1.1.) and eq.(II.1.1.2.):

$$\begin{aligned} E[g(0,0)] &= E\Big[\sum_{i}^{N} \sum_{j}^{N} f(i,j)\beta[f(i,j)]\Big] \\ &= \sum_{i}^{N} \sum_{j}^{N} E\Big[|f(i,j)|\Big] \end{aligned} \qquad (II.1.1.4.)$$

We assumed earlier that the probability density function of $f(i,j)$ is given as

$$p_f(\alpha) = \frac{1}{\sqrt{2\pi}\sigma_f} exp(\frac{-\alpha^2}{2\sigma_f^2}) \qquad -\infty < \alpha < \infty.$$

By simple transformation, it can be shown that the probability density function of $|f(i,j)|$ is given as

$$p_{|f|} = \begin{cases} \frac{2}{\sqrt{2\pi}\sigma_f} exp(\frac{-\alpha^2}{2\sigma_f^2}) & \text{for } 0 \geq \alpha < \infty; \\ 0 & \text{for } -\infty < \alpha < 0. \end{cases} \qquad (II.1.1.5.)$$

Using the above equation,

$$E\Big[|f(i,j)|\Big] = \int_0^\infty \frac{2}{\sqrt{2\pi}\sigma_f} exp(\frac{-\alpha^2}{2\sigma_f^2})\alpha d\alpha$$
$$= \sqrt{\frac{2}{\pi}}\sigma_f. \qquad (II.1.1.6.)$$

Thus,

$$E[g(0,0)] = N^2\sqrt{\frac{2}{\pi}}\sigma_f \qquad (II.1.1.7.)$$

The variance of $g(i',j')$ is given as

$$var[g(i',j')] = E[g^2(i',j')] - E^2[g(i',j')].$$

For $i' \neq 0, j' \neq 0$,

$$E[g(i',j')] = \sum_i^N \sum_j^N E[f(i+i',j+')]\, E[\beta[f(i,j)]]$$
$$= 0 \qquad (II.1.1.8.)$$

and

$$E[g^2(i',j')] = \sum_i^N \sum_j^N \sum_{i_1}^N \sum_{j_1}^N E[f(i+i',j+j')$$
$$\cdot \beta[f(i,j)]f(i_1+i',j_1+j')\beta[f(i_1,j_1)]$$
$$= \sum_i^N \sum_j^N E[f^2(i+i',j+j')] + \sum_i^N \sum_j^N \sum_{\substack{i_1 \\ i_1 \neq i}}^N \sum_{\substack{j_1 \\ j_1 \neq j}}^N E[f(i+i',j+j')$$
$$\cdot \beta[f(i,j)]f(i_1+i',j_1+j')\beta[f(i_1,j_1)]$$
$$= N^2\sigma_f^2 \qquad (II.1.1.9.)$$

Her we have used the fact that $f(i+i',j+j')$ and $f(i_1+i',j_1+j')$ are statistically independent to evaluate

$$E[f(i+i',j+j')\beta[f(i,j)]f(i_1+i',j_1+j')\beta[f(i_1,j_1)]=0 \qquad i\neq i_1, j\neq j_1.$$

Thus, according to eq.(II.1.1.3.), the processing gain of the BIMF, GP_1, is given as

$$GP_1 = N^2\sqrt{\frac{2}{\pi}}. \qquad (II.1.1.10.)$$

Note that GP_1 is a linear function of the number of samples, or discrete space-bandwidth product. This result is encouraging, for it means the relative size of the correlation peak and side-lobe level can be made sufficient large by increasing the information content in the reference filter. It can be shown easily that the processing gain of the matched filter is equal to N^2 under the same statistical assumptions. As a comparison, the processing gain of the BIMF suffers only a modest loss of $\sqrt{\frac{2}{\pi}}$. The result we obtained here does not only apply to input sequences of independent gaussian random variables. The order of GP_1 remains to be N^2 for other input sequences of independent random variables with finite (non-zero) variances having symmetric probability density functions about the origin.

II.1.2. Signal-to-Noise Ratio

We have verified that the BIMF possesses image identification potential comparable to the conventional matched filter. Next, we will investigate the capability of the BIMF in recognizing patterns resembling the reference image stored in the filter. To facilitate the analysis, we will model the patterns resembling the reference

image by contaminating the input image by noise. The noise chosen here is an additive gaussian white random process with mean zero and variance σ_n^2. Thus, instead of $f(i,j)$, the input in this case is

$$f(i,j) + n(i,j)$$

where $n(i,j)$ is the sampled sequence of the gaussian white noise. The reference filter is still given by eq.(II.1.1.2.). The correlation output becomes

$$g(i',j') = \sum_{i}^{N} \sum_{j}^{N} f(i+i',j+j')h(i,j) + \sum_{i}^{N} \sum_{j}^{N} n(i+i',j+j')h(i,j). \qquad (II.1.2.1.)$$

Unlike what we have shown in the last section, we can no longer guarantee that the center correlation level will be the highest among all output data points due to the presence of noise. A reasonable assumption is that the peak value should still be at the center when the noise level is low, whereas peaks may appear at other locations when the noise level is high. An objective measure, similar to the one used in the last section, is hence required to qualify the "outstandingness" of the center correlation level. To differentiate from the processing gain defined earlier, we define a new quantity known as the signal-to-noise ratio:

$$SNR = \frac{E^2[g(0,0)]}{var[g(i',j']} \qquad (II.1.2.2.)$$

The expected value of $g(i',j')$ can be calculated as follows:

$$E[g(i',j')] = \sum_{i}^{N} \sum_{j}^{N} E[f(i+i',j+j')h(i,j)] + \sum_{i}^{N} \sum_{j}^{N} E[n(i+i',j+j')h(i,j)].$$

$(II.1.2.3.)$

The expected value of the first term on the RHS is identical to that in eq.(II.1.1.4.) and the answer is given in eq.(II.1.1.6.). As for the second term on the RHS, we may use the fact that n and h are independent to arrive at the answer zero. Combining these results, we have

$$g(i',j') = \begin{cases} N^2\sqrt{\frac{2}{\pi}}\sigma_f & \text{for } i'=0, j'=0; \\ 0 & \text{for } i' \neq 0, j' \neq 0. \end{cases} \qquad (II.1.2.4.)$$

The evaluation of the variance is presented below:

$$\begin{aligned} E[g^2(i',j')] &= \sum_{i}^{N}\sum_{j}^{N}\sum_{i_1}^{N}\sum_{j_1}^{N} E\Big[f(i+i',j+j') \\ &\quad \cdot \beta[f(i,j)]f(i_1+i',j_1+j')\beta[f(i_1,j_1)]\Big] \\ &\quad + 2\sum_{i}^{N}\sum_{j}^{N}\sum_{i_1}^{N}\sum_{j_1}^{N} E\Big[f(i+i',j+j')\beta[f(i,j)] \\ &\quad \cdot n(i_1+i',j_1+j')\beta[f(i_1,j_1)]\Big] \\ &\quad + \sum_{i}^{N}\sum_{j}^{N}\sum_{i_1}^{N}\sum_{j_1}^{N} E\Big[n(i+i',j+j') \\ &\quad \cdot \beta[f(i,j)]n(i_1+i',j_1+j')\beta[f(i_1,j_1)]\Big] \\ &= \sum_{i}^{N}\sum_{j}^{N} E\Big[f^2(i+i',j+j')\Big] \\ &\quad + \sum_{i}^{N}\sum_{j}^{N}\sum_{\substack{i_1 \\ i_1\neq i}}^{N}\sum_{\substack{j_1 \\ j_1\neq j}}^{N} E\Big[f(i+i',j+j') \\ &\quad \cdot \beta[f(i,j)]f(i_1+i',j_1+j')\beta[f(i_1,j_1)]\Big] \end{aligned}$$

$$+\sum_{i}^{N}\sum_{j}^{N} E\Big[n^2(i+i',j+j')\Big]$$
$$+\sum_{i}^{N}\sum_{j}^{N}\sum_{\substack{i_1\\ i_1\neq i}}^{N}\sum_{\substack{j_1\\ j_1\neq j}}^{N} E\Big[n(i+i',j+j')$$
$$\cdot\beta[f(i,j)]n(i_1+i',j_1+j')\beta[f(i_1,j_1)]\Big]$$
$$= N^2(\sigma_f^2+\sigma_n^2) \qquad (II.1.2.5.)$$

Again, we used the fact that f and n are statistically independent to deduce that

$$E[f(i+i',j+j')\beta[f(i,j)]n(i_1+i',j_1+j')\beta[f(i_1,j_1)] = 0.$$

Substituting eq.(II.1.2.4.) and eq.(II.1.2.5.) into eq.(II.1.2.2.), it follows that the SNR of the BIMF is given as

$$SNR_1 = N^2\sqrt{\frac{2}{\pi}}\frac{\sigma_f^2}{\sigma_f^2+\sigma_n^2} \qquad (II.1.2.6.)$$

Defining a new quantity $SNR_{in} = \frac{\sigma_f^2}{\sigma_n^2}$, which is a reflection of the relative strength of the input signal and noise energy, eq.(II.1.2.6¿) can be rewritten as

$$SNR_1 = N^2\sqrt{\frac{2}{\pi}}\frac{SNR_{in}}{SNR_{in}+1}. \qquad (II.1.2.7.)$$

From the above equation we observe that for large SNR_{in}, i.e., the input signal level is high compared to the noise level, SNR_1 is approximately equal to PG_1. As for small SNR_{in},

$$SNR_1 \sim SNR_{in}PG_1, \qquad (II.1.2.8.)$$

In comparison, the SNR_M of the conventional matched filter is given:

$$SNR_M = N^2\frac{SNR_{in}}{SNR_{in}+1}. \qquad (II.1.2.9.)$$

Here we note that the loss in SNR_1 is also a constant factor of $\frac{2}{\pi}$ compared to SNR_M. This indicates that the performance of the BIMF is comparable to that of the matched filter in identifying distorted images.

II.1.3. Probability of Error

Up to this point, we have only addressed the intra-class pattern identification capability of the BIMF. We have little knowledge of how the BIMF would perform if subjected to patterns other than the one that is stored. If we assume that the pattern stored is uncorrelated with all other possible input patterns, they appear to be noise to the filter. Their output levels are expected to be zero. A performance measure similar to the signal to noise ratio can be defined to quantify the inter-class discrimination capability of the BIFM. A result similar to SNR_1 would be obtained. Instead of doing just that, we will calculate the probability of error to signify the performance of the BIFM of the concerned aspect. The problem that we will tackle concerns discrimination of two classes of patterns. An error is said to occur if a pattern belonging to one class is identified as a member of another class. The probability of error is defined as the probability that such an error may occur.

Suppose we have two classes of patterns, Ω_1 and Ω_2 to be classified. Each of these classes will be represented by a single pattern: Ω_1 by $f_1(i,j)$ and Ω_2 by $f_2(i,j)$. The other elements in $\Omega_m, m = 1,2$ are distorted versions of $f_m(i,j)$. The distortion is modeled by an additive gaussian white noise of mean zero and variance σ_n^2:

$$f'_m(i,j) = f_m(i,j) + n(i,j) \qquad m = 1,2. \qquad (II.1.3.1.)$$

The machine (Fig.II.1.3.1.) used to classify Ω_1 and Ω_2 consists of two BIFMs.

Each of the BIFM is generated from the algorithm described by eq.(II.1.1.2.) using the class representation pattern as the reference pattern. Thus BIFM1 and BIFM2 are such that

$$h_1(i,j) = \beta[f_1(i,j)] \qquad (II.1.3.2a.)$$

and

$$h_2(i,j) = \beta[f_2(i,j)]. \qquad (II.1.3.2b.)$$

When $f'(i,j)$, randomly chosen from the union of Ω_1 and Ω_2, is applied to the machine, two outputs g_1 and g_2 are available, each corresponding to a different BIFM:

$$g_1 = \sum_i^N \sum_j^N f'(i,j)h_1(i,j) \qquad (II.1.3.3a.)$$

and

$$g_2 = \sum_i^N \sum_j^N f'(i,j)h_2(i,j) \qquad (II.1.3.3b.)$$

To determine which class f' originated from, g_1 and g_2 are compared. f' is classified to be in Ω_m, if $g_m(0,0)$ is a maximum. An error occurs if $g_m(0,0) > g_{m'}(0,0)$ $(m = 1,2; m' = 2,1)$ when f' actually comes from $\Omega_{m'}$, and vice versa.

g_1 and g_2 can also be expressed as

$$g_1 = \sum_i^N \sum_j^N [f(i,j)h_1(i,j) + n(i,j)h_1(i,j)] \qquad (II.1.3.4a.)$$

and

$$g_2 = \sum_i^N \sum_j^N [f(i,j)h_2(i,j) + n(i,j)h_2(i,j)] \qquad (II.1.3.4b.)$$

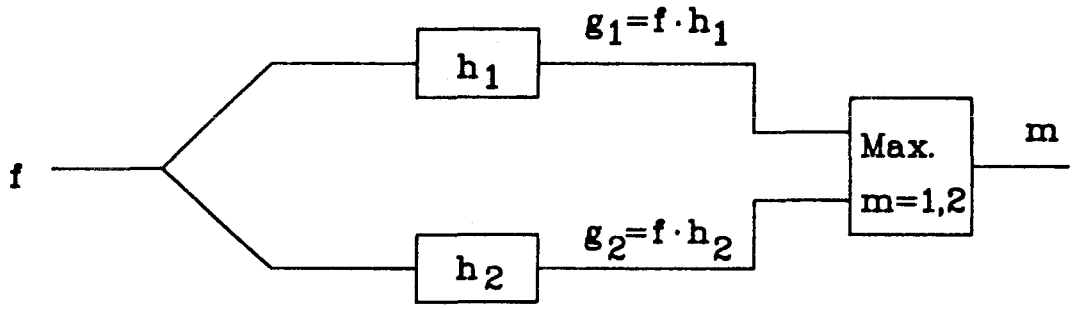


Figure.II.1.3.1. Two-category classifier consists of two BIFMs.

where f can either be f_1 or f_2. To simplify the following discussion, we may assume $f = f_1$ with no loss of generality. By inspecting eq.(II.1.3.4), the cause of possible errors becomes visible. While the first term on the RHS is the uncontaminated correlation level, the second term may diminish $g_1(0,0)$ enough so that it is less than $g_2(0,0)$.

Since n is independent of f_m, when it passes the filters h_m, all it sees are collections of +1 and -1 which have practically no effect on its probability density functions. In other words, the probability density functions of $n(i,j)$ remain gaussian. The summation over i and j is such that the effective noise term in the output is also a gaussian distribution (the sum of any number of independent gaussian random variables is also gaussian). As a consequence, we have two gaussian random variables, $g_1(0,0)$ and $g_2(0,0)$, with means $f \cdot h_1$ and $f \cdot h_2$, and variances $var[n \cdot h_m]$, $m = 1,2$.

The decision in favor of Ω_m, $m = 1,2$, is made whenever the dot-product of h_m and f' is the maximum. Geometrically (Fig.II.1.3.2.), it can be interpreted as which h_m f' is closer to.

The equality of the two dot-products,

$$\underline{\mathrm{h}}_1 \cdot \underline{\mathrm{f}'} = \underline{\mathrm{h}}_2 \cdot \underline{\mathrm{f}'},$$

defines a planar surface, P1 equidistant from the binary patterns $\underline{\mathrm{h}}_1$ and $\underline{\mathrm{h}}_2$. P1 effectively divides the N-dimensional space into two decision regions, R_1 and R_2. Other than comparing the dot-products, the location of $\underline{\mathrm{f}'}$ with respect to P1 will also equivalently determine which class $\underline{\mathrm{f}'}$ belongs to. If $\underline{\mathrm{f}'}$ is closer to $\underline{\mathrm{h}}_1$, then $\underline{\mathrm{f}'}$ is classified to Ω_1 and vice versa. The effect of the noise is such that it may put the $\underline{\mathrm{f}'}$

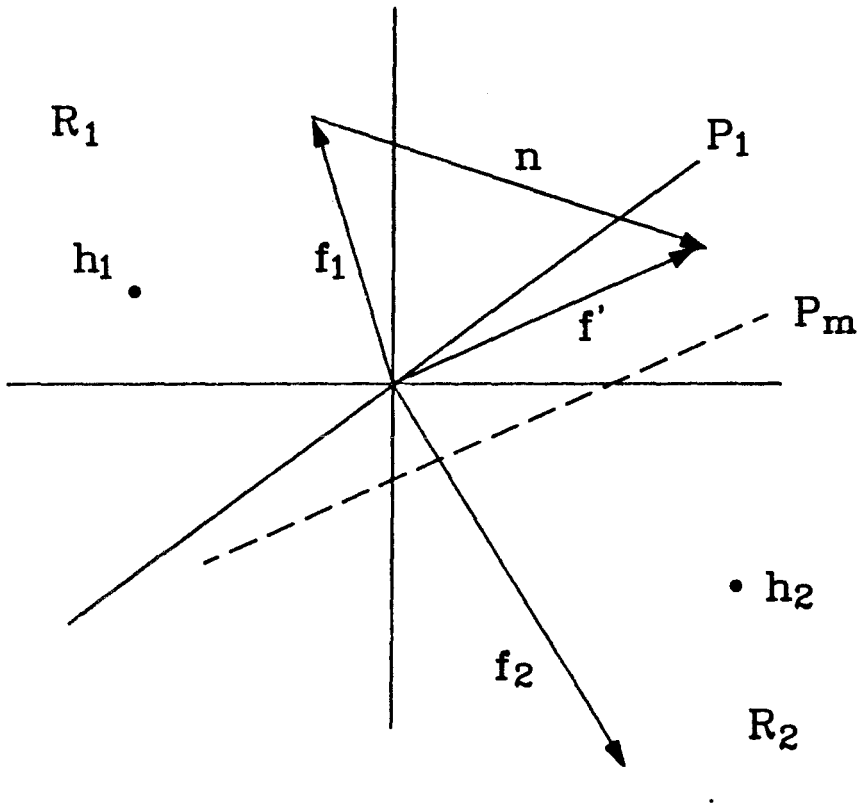


Figure II.1.3.2. Geometric representation of decision surfaces in a two-dimensional space.

in the wrong decision region. Note that P1 is the optimum binary decision surface. It is because

$$\underline{\mathrm{f}} \cdot \underline{\mathrm{h}} + \underline{\mathrm{n}} \cdot \underline{\mathrm{h}} \leq |\underline{\mathrm{f}}| + \underline{\mathrm{n}} \cdot \underline{\mathrm{h}},$$

with the equality holds when $\underline{\mathrm{h}} = \beta[\underline{\mathrm{f}}]$. P1, however, is not the optimum decision surface. The optimum decision surface, by definition, is the surface which will result in the minimum probability of error. It can be shown that the optimum decision surface is a plane equidistant from f_1 and f_2. By inspection, the optimum decision surface is achieved if the matched filters are used.

Since the mean of f' is either f_1 or f_2, the probability of error is one-half the sum of the probabilities that the gaussian noise may put f' in R_2, given that the input is actually f_1 or f' in R_1 given f_2:

$$P_\epsilon = \frac{1}{2}\left\{ \int \cdots \int_{R_2} G(f_1, \sigma_n^2)\, d\underline{\mathrm{x}} + \int \cdots \int_{R_1} G(f_2, \sigma_n^2)\, d\underline{\mathrm{x}} \right\} \qquad (II.1.3.5.)$$

where $G(.,.)$ is the joint gaussian density function with mean f_m, $m = 1, 2$ and variance σ_n^2. The factor $1/2$ comes from the fact that f' can be from Ω_1 or Ω_2 with equal probability. P_ϵ is never less than the probability of error when the MF is used. The reason is that P1 usually does not coincide with the optimum decision surface.

A more detailed form of P_ϵ will be given in the following. Another way to express the plane defining the the decision surface is

$$(\underline{\mathrm{h}}_1 - \underline{\mathrm{h}}_2) \cdot \underline{\mathrm{f}}' = \underline{0}$$

The unit vector normal to this plane is given as

$$\underline{\mathrm{h}}_u = \frac{(\underline{\mathrm{h}}_1 - \underline{\mathrm{h}}_2)}{\|\underline{\mathrm{h}}_1 - \underline{\mathrm{h}}_2\|} \qquad (II.1.3.6.)$$

By transformation of coordinates, the evaluation of $P_{\epsilon 1}$ can be reduced to a one-dimensional problem. The transformation is such that one of the new coordinates, say α, is parallel to $\underline{\mathbf{h}}_u$, whereas the rest are perpendicular to $\underline{\mathbf{h}}_u$. Since the mid point of $(\underline{\mathbf{h}}_1 - \underline{\mathbf{h}}_2)$ is also the intersection point of the decision plane and $(\underline{\mathbf{h}}_1 - \underline{\mathbf{h}}_2)$ (because $\underline{\mathbf{h}}_1$ and $\underline{\mathbf{h}}_2$ are of equal length), the origin of α can always be chosen to be the mid-point of $(\underline{\mathbf{h}}_1 - \underline{\mathbf{h}}_2)$.

The projection of $\underline{\mathbf{n}}$ onto α is given as

$$\underline{\mathbf{h}}_u \cdot \underline{\mathbf{n}} = \sum_i \sum_j h_u(i,j)n(i,j) \qquad (II.1.3.7.)$$

The mean of the projection of the noise is zero. The variance of the projection of the noise is equal to

$$\begin{aligned} \sigma_{np}^2 &= E\Big[[\sum_i \sum_j h_u(i,j)n(i,j)]^2\Big] \\ &= \sum_i \sum_j E[h_u^2(i,j)]E[n^2(i,j)] \\ &= \sigma_n^2 \sum_i \sum_j E[h_u^2(i,j)] \end{aligned} \qquad (II.1.3.8.)$$

Suppose l bits of $\underline{\mathbf{h}}_1$ and $\underline{\mathbf{h}}_2$ are different. By virtue of eq.(II.1.3.6.), the magnitude of each non-zero component of $\underline{\mathbf{h}}_u$ is equal to $1/\sqrt{l}$. Due to the square in $E[h_u^2(i,j)]$, the sign of $h_u(i,j)$ has effect in the evaluation of the double summation in the last equation. Thus, eq.(II.1.3.8.) becomes

$$\sigma_{np}^2 = \sigma_n^2 \sum_i \sum_j \frac{1}{l} \qquad (II.1.3.9.)$$

But of the N^2 components of $n(i,j)$, only l of them have non-zero projections onto $\underline{\mathbf{h}}_u$, therefore

$$\sigma_{np}^2 = \sigma_n^2 \qquad (II.1.3.10.)$$

The projections of $\underline{f}_1$ and $\underline{f}_2$ on $\underline{h}_u$ are given as

$$\mu_1 = \underline{f}_1 \cdot \underline{h}_u \qquad (II.1.3.11a.)$$

and

$$\mu_2 = \underline{f}_2 \cdot \underline{h}_u \qquad (II.1.3.11b.)$$

Note that $|\mu_1| = |\mu_2| = \mu$.

Substituting eq.(II.1.3.7) through eq.(II.1.3.11.) in eq.(II.1.3.5.), the probability of error becomes

$$P_{\epsilon 1} = \int_0^\infty \frac{1}{\sqrt{2\pi}\sigma_n} exp\Big(- \frac{(\alpha - \mu)^2}{2\sigma_n^2}\Big) d\alpha \qquad (II.1.3.12.)$$

As a comparison, the probability of error of a matched filter is derived : The decision surface in this case is defined by the plane (Fig.II.1.3.2.)

$$(\underline{f}_1 - \underline{f}_2) \cdot \underline{f} = 0$$

The unit vector normal to the decision plane is

$$\frac{(\underline{f}_1 - \underline{f}_2)}{\|\underline{f}_1 - \underline{f}_2\|} \qquad (II.1.3.13.)$$

The variance of the projection of the noise is

$$\sigma_m^2 = \frac{\sigma_n^2}{\|\underline{f}_i - \underline{f}_2\|^2} \sum_{i,j}^{N^2} \Big((f_1(i,j) - f_2(i,j)\Big)^2 = \sigma_n^2 \qquad (II.1.3.14.)$$

The distance from either $\underline{f}_1$ or $\underline{f}_2$ to the decision plane is equal to

$$\mu_m = \frac{1}{2}\|\underline{f}_i - \underline{f}_2\| \qquad (II.1.3.15.)$$

The probability of error is therefore given as

$$P_{\epsilon m} = \int_{-\infty}^{0} \frac{1}{\sqrt{2\pi}\sigma_n} exp(- \frac{(\alpha - \mu_m)^2}{2\sigma_n^2}) \, d\sigma \qquad (II.1.3.16.)$$

In the expressions for P_ϵ (eq.II.1.3.12.) and $P_{\epsilon m}$ (eq.II.1.3.16.), the only difference is the means in the arguments of the integrands. In the case of P_ϵ, μ is the distance of one of the binary filters to the optimum binary decision surface, whereas in the other case, μ_m is the distance of one of the matched filters to the optimum decision surface. Note that P_ϵ ($P_{\epsilon m}$) decreases as the distance μ (μ_m) increases. Assume that the matched filter and the binary filter have the same energy content (i.e. $f_{1,2}$ always lie on the N^2 dimensional hypersphere centering at the origin and having a radius of N^2). There are situations for which one probability is greater than the other and vice versa. Vaguely speaking, P_ϵ and $P_{\epsilon m}$ are of the "same order of magnitude."

The only difference between P_ϵ and $P_{\epsilon m}$ is the distance of $f_{1,2}$ to the decision surface $P_{\epsilon m} \rightarrow 0$ as the ratio $\frac{\mu_m^2}{\sigma_m^2} \rightarrow \infty$ regardless of the exact positions of $\underline{\mathbf{f}}_1$ and $\underline{\mathbf{f}}_2$. There are only two factors which can cause $\frac{\mu_m^2}{\sigma_m^2}$ to increase. One of them is the input signal-to-noise ratio. The other is the discrete space-bandwidth product. From the definition of P1, we see that $(\underline{\mathbf{f}}_i - \underline{\mathbf{f}}_2)$ always has a non-zero component in the direction of $\underline{\mathbf{h}}_u$. In general, as $\frac{\mu_m^2}{\sigma_m^2}$ increases, so does $\frac{\mu_1^2}{\sigma_{np}^2}$ and $\frac{\mu_2^2}{\sigma_{np}^2}$. In other words, as $P_{\epsilon m} \rightarrow 0$, $P_{\epsilon 1}$ usually tends to 0 as well. It is only in the extreme situation (Fig.II.1.3.3.) that the direction of increase of $\frac{\mu_m^2}{\sigma_m^2}$ is orthogonal to $\underline{\mathbf{h}}_u$ will $P_{\epsilon m} \rightarrow 0$ while $P_{\epsilon 1}$ remains a non-zero constant.

This happens when the increase of $\frac{\mu_m^2}{\sigma_m^2}$ is due to the increase in magnitudes of only a fraction, but not all, of the components of $\underline{\mathbf{f}}_m, m = 1, 2$. If the increase in

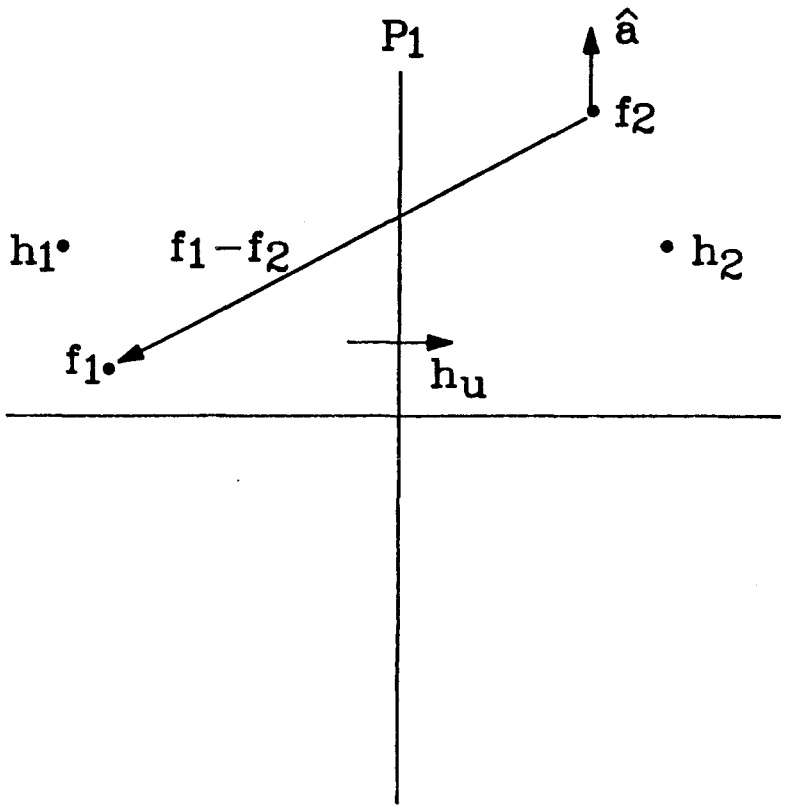


Figure II.1.3.3. Special situation in which the probability of error of the matched filter tends to zero but not that of the binary matched filter.

$\frac{\mu_m^2}{\sigma_m^2}$ is caused by an increase in N^2, the extreme situation will not occur, because we expect the input signal-to-noise ratio is roughly the same in each component of $\underline{f}_m$.

In this section, we have calculated the processing gain and the signal-to-noise ratio, and heuristically analyzed the probability of error. All of them indicate the same thing: the thresholding of the image plane matched filter does not significantly degrade the performance of the resultant filter in the task of pattern identification.

II.2. Binary Input Binary Filter

In this section, we investigate the effects of thresholding the input plane. If the input image is binary and essentially noise-free (text recognition, for example), then, of course, thresholding the input plane does not present any problem. If the input image is not binary but noise free, then the performance of the correlator will be identical to what we have shown in the last section, should a filter matched to the image before thresholding be used. On the other hand, if both the input and the filter are obtained by thresholding a noise-free image, we would have the equivalent of the matched filter for a binary image. One interesting case that we have not yet studied is a correlator implemented with a BIMF (binary image plane matched filter; section II.1.1.) in which the input is a thresholded noisy image.

Again, let $f(i,j)$ be the image we wish to identify. Let the contamination of f be the additive noise $n(i,j)$. The statistics of f and n are given in section II.1.1.

The thresholded input image is defined as

$$b(i,j) = \begin{cases} +1 & \text{if } f(i,j) + n(i,j) \geq 0; \\ -1 & \text{if } f(i,j) + n(i,j) < 0. \end{cases} \qquad (II.2.0.1.)$$

The filter $h(i,j)$ is the BIMF described in eq.II.1.1.2. To write it down again,

$$h(i,j) = \begin{cases} +1 & \text{if } f(i,j) \geq 0; \\ -1 & \text{if } f(i,j) < 0. \end{cases} \qquad (II.2.0.2.)$$

The correlation between the thresholded input and the BIMF is then given by

$$g(i',j') = \sum_{i}^{N} \sum_{j}^{N} b(i+i', j+j')h(i,j) \qquad (II.2.0.3.)$$

When the input image is noise free, the correlation above becomes the auto-correlation of the thresholded pattern of f. Naturally, a correlation peak identifying the presence of f appears at the center of the output plane. In the presence of noise, however, there is no guarantee that such a peak will appear. To observe the effect of thresholding a noisy image on the correlation peak, the signal-to-noise ratio will be calculated. Note that in the absence of noise, the processing gain of the correlation given by eq.(II.2.0.3.) is N^2, which should serve as a useful check for the SNR in the limit.

II.2.1. Signal-to-Noise Ratio

The expected value and the variance of the correlation in eq.II.2.0.3. can be calculated as follows:

$$E\Big[g(i',j')\Big] = \sum_{i}^{N} \sum_{j}^{N} E\Big[b(i+i', j+j')h(i,j)\Big]. \qquad (II.2.1.1.)$$

For $i' \neq 0, j' \neq 0$, $b(i+i', j+j')$ and $h(i,j)$ are uncorrelated and

$$E\Big[b(i+i', j+j')h(i,j)\Big] = E\Big[b(i+i', j+j')\Big]E\Big[h(i,j)\Big]$$
$$= 0 \qquad (II.2.1.2.)$$

For $i' = 0, j' = 0$, $b(i,j)$ and $h(i,j)$ have different signs if either

$$f(i,j) \geq 0 \text{ and } f(i,j) + n(i,j) < 0$$

or

$$f(i,j) < 0 \text{ and } f(i,j) + n(i,j) \geq 0$$

The probability that $b(i,j)$ and $h(i,j)$ have different signs is therefore

$$P_d = P\Big[f(i,j) \geq 0, f(i,j) + n(i,j) < 0\Big] + P\Big[f(i,j) < 0, f(i,j) + n(i,j) \geq 0\Big]$$

$$(II.2.1.3.)$$

Applying the Bayes rule, we have

$$P_d = P\Big[f(i,j) + n(i,j) < 0 | f(i,j) \geq 0\Big] P\Big[f(i,j) \geq 0\Big]$$
$$+ P\Big[f(i,j) + n(i,j) \geq 0 | f(i,j) < 0\Big] P\Big[f(i,j) < 0\Big]$$
$$= \frac{1}{2}\Big\{ P\Big[f(i,j) + n(i,j) < 0 | f(i,j) \geq 0\Big]$$
$$+ P\Big[f(i,j) + n(i,j) \geq 0 | f(i,j) < 0\Big]\Big\}$$

because $P[f(i,j) \geq 0] = P[f(i,j) < 0] = 1/2$. $P\Big[f(i,j) + n(i,j) < 0 | f(i,j) \geq 0\Big]$ can also be expressed as

$$P\Big[n(i,j) < -f(i,j) | f(i,j) \geq 0\Big]$$

Now $n(i,j)$ is a gaussian distributed random variable and for $f(i,j)$ fixed, the above probability is equal to the shaded area depicted in Fig.II.2.1.1. which can also be expressed as

$$A_d = \int_{-\infty}^{-\alpha} \frac{1}{\sqrt{2\pi}\sigma_n} exp(\frac{-\gamma^2}{2\sigma_n^2})\, d\gamma$$

Since $f(i,j)$ can attain values between 0 and ∞ determined by a gaussian distribution,

$$P\Big[f(i,j)+n(i,j)<0\,\Big|\,f(i,j)\geq 0\Big]$$
$$= \int_0^{\infty}\int_{-\infty}^{-\alpha} \frac{1}{\sqrt{2\pi}\sigma_n} exp(\frac{-\gamma^2}{2\sigma_n^2})\, d\gamma \frac{1}{\sqrt{2\pi}\sigma_f} exp(\frac{-\alpha^2}{2\sigma_f^2})\, d\alpha \qquad (II.2.1.4.)$$

The above equation can also be expressed as

$$P\Big[f(i,j)+n(i,j)<0\Big|f(i,j)\geq 0\Big] = \int_0^{\infty} Q(\frac{\alpha}{\sigma_n}) \frac{1}{\sqrt{2\pi}\sigma_f} exp(\frac{-\alpha^2}{2\sigma_f^2})\, d\alpha$$

where

$$Q\Big(\frac{\alpha}{\sigma_n}\Big) = \int_{\frac{\alpha}{\sigma_n}}^{\infty} \frac{1}{\sqrt{2\pi}} exp(\frac{-\gamma^2}{2})\, d\gamma. \qquad (II.2.1.5.)$$

Differentiating eq.(II.2.1.5.) with respect to σ_n, we have

$$\frac{\partial}{\partial\sigma_n} P\Big[f(i,j)+n(i,j)<0\Big|f(i,j)\geq 0\Big]$$
$$= \int_0^{\infty} \frac{\partial}{\partial\sigma_n}\Big(Q(\frac{\alpha}{\sigma_n})\Big) \frac{1}{\sqrt{2\pi}\sigma_f} exp\Big(\frac{-\alpha^2}{2\sigma_f^2}\Big)\, d\alpha$$
$$= \frac{1}{2\pi\sigma_f\sigma_n^2} \int_0^{\infty} exp\Big(-\alpha^2(\frac{1}{2\sigma_n^2}+\frac{1}{2\sigma_f^2})\Big)\, \alpha d\alpha$$
$$= \frac{1}{2\pi}\frac{\sigma_f}{\sigma_f^2+\sigma_n^2} \qquad (II.2.1.6.)$$

Integrating the above equation with respect to σ_n results in

$$P\Big[f(i,j)+n(i,j)<0\Big|f(i,j)\geq 0\Big] = \frac{1}{2\pi} tan^{-1}(\frac{\sigma_n}{\sigma_f}) + C$$

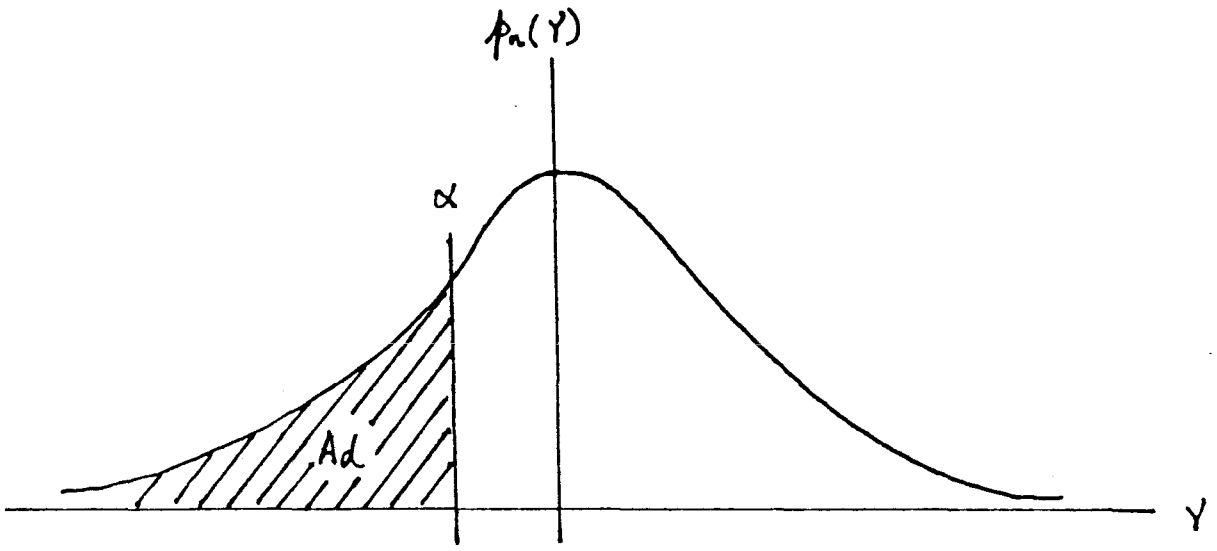


Figure II.2.1.1. A Gaussian distribution. The shaded area is equal to A_d.

where C is the integration constant. By matching the boundary condition that

$$P\Big[f(i,j)+n(i,j)<0|f(i,j)\geq 0\Big]\rightarrow 0 \quad \text{as} \quad \frac{\sigma_n}{\sigma_f}\rightarrow 0$$

we find that $C=0$. Now by symmetry

$$P\Big[f(i,j)+n(i,j)<0|f(i,j)\geq 0\Big]=P\Big[f(i,j)+n(i,j)\geq 0|f(i,j)<0\Big].$$

Therefore,

$$P_d=\frac{1}{2\pi}tan^{-1}(\frac{\sigma_n}{\sigma_f}). \qquad (II.2.1.7.)$$

The probability that $b(i,j)$ and $h(i,j)$ are of the same sign is $1-P_d$. The expected value of the product of $b(i,j)$ and $h(i,j)$ when they have the same sign is 1. The expected value of the product of $b(i,j)$ and $h(i,j)$ when they have different signs is -1. Thus

$$\begin{aligned} E[b(i,j)h(i,j)] &= 1-2P_d \\ &= 1-\frac{1}{\pi}tan^{-1}(SRN_{in}^{-1}) \end{aligned} \qquad (II.2.1.8.)$$

where $SRN_{in}=\sigma_f/\sigma_n$. Substituting eq.(II.2.1.8.) in eq.(II.2.1.1.) gives the following result:

$$E[g(0,0)]=N^2\Big(1-\frac{1}{\pi}tan^{-1}(SRN_{in}^{-1})\Big)$$

The variance of $g(i',j')$ can be obtained as follows:

$$\begin{aligned} var[g(i',j')] &= \sum_{i}^{N}\sum_{j}^{N}\sum_{i_1}^{N}\sum_{j_1}^{N} E[b(i'+i,j'+j)h(i,j)b(i'+i_1,j'+j_1)h(i_1,j_1)] \\ &= \sum_{i}^{N}\sum_{j}^{N} E[f^2(i'+i,j'+j)h^2(i,j)] \\ &= N^2 \end{aligned} \qquad (II.2.1.9.)$$

By using eq.(II.2.1.8.) and eq.(II.2.1.9.), the signal-to-noise ratio, defined by eq. (II.1.2.2.), of the BIMF, when the input is also thresholded, is given as

$$SNR_2 = N^2\left(1 - \frac{1}{\pi}tan^{-1}(SRN_{in}^{-1})\right)^2 \qquad (II.2.1.10.)$$

In order to appreciate the effects of thresholding, we compare this ratio with the signal-to-noise ratio that is obtained if the matched filter is used. Writing eq.(II.1.2.9.) again,

$$SNR_m = \frac{N^2 SNR_{in}^2}{1 + SNR_{in}^2}.$$

The two ratios, SNR_2 and SNR_m, are plotted in Figure.II.2.1.2. as a function of SNR_{in}^2. Notice that the output SNR_2 is larger for low SNR_{in}. Both ratios asymptotically converge N^2 as SNR_{in} gets large. Starting at the initial point of $1/2$, SNR_2, however, increases at a lower rate with SNR_{in}, and thus there is a crossover at approximately $SNR_{in} = 2$. At this point, the output SNRs exceeds $N^2/2$ (one-half of the number of pixels of the input image), and thus the loss in output SNR for $SNR_{in} > 2$ will not degrade appreciably the probability of detection in a typical application in which N^2 is a large number.

The conclusions we can draw from this analysis are that, under the stated assumptions, thresholding the input image before it is applied to the correlator actually improves the performance if the input SNR is low, and that the performance is only marginally degraded if the input SNR is high compared to a correlator using a matched filter. Recall from section II.1.2 that the SNR of a correlator using a BIMF is $2/\pi$ of that of the matched filter. Comparing SNR_1 and SNR_2, we see that thresholding the input actually reduces the negative effect of noise and thus it

is always preferable as far as the SNR is concerned. The assumptions used in the analyses are that the input image and the additive noise are both independent.

II.2.2. Probability of Error of Two-Pattern Classification

The probability of error for the problem of two-pattern classification discussed in section II.1.3 will be analyzed here for contrast. Note that the deviation from section II.1.3 is that the input image is thresholded here. Recall that the machine consists of BIMF1 and BIMF2 given by

$$h_1(i,j) = \beta[f_1(i,j)] \qquad (II.2.2.1a.)$$

and

$$h_2(i,j) = \beta[f_2(i,j)]. \qquad (II.2.2.1b.)$$

But here, the input is

$$b(i,j) = \beta[f(i,j) + n(i,j)] \qquad (II.2.2.2.)$$

where $f(i,j)$ may belong either to Ω_1 or Ω_2. Recall from section II.1.3. that only the dot-products, g_1 and g_2 are considered:

$$g_1 = \sum_i^N \sum_j^N b(i,j)h_1(i,j) \qquad (II.2.2.3a.)$$

and

$$g_2 = \sum_i^N \sum_j^N b(i,j)h_2(i,j). \qquad (II.2.2.3b.)$$

The decision rule remains the same: f is classified to Ω_1 if $g_1 > g_2$ and vice versa. The effect of thresholding is such that the dot products are always between binary

patterns. Thus, the decision rule can be modified to one that compares the hamming distances between b and the two reference patterns, h_1 and h_2.

Suppose $x_{i,j}$ stands for the (i,j)-th variable of a N^2 dimensional space. The N^2 planes $\{x_{i,j} = 0\}$ divides the N^2-dimensional space into 2^{N^2} regions. We shall denote such region by the letter R (in 2-D, R is known as a quadrant, and in 3-D, an octant). Within each R, all points have the same sign, i.e., all points will be mapped to the same binary number when thresholded. In view of this property, it is clear that all points in the same R will be classified to Ω_1 (Ω_2) if the binary string of that R is closer in hamming distance to h_1 (h_2). (See Figure II.2.2.1.).

The decision region of Ω_1, R_1, therefore is a union of all the disjointed $R's$ which have binary points closer to h_1 in hamming distance than h_2. R_2, the decision region of Ω_2, is defined in a similar fashion. An error is said to occur if b lands in R_1 (R_2) given the condition that f belongs to Ω_2 (Ω_1).

There may be binary numbers that have the same dot-products with both $\underline{\mathrm{h}}_1$ and $\underline{\mathrm{h}}_2$. (The $R's$ containing these binary numbers are known as the ambiguity regions. An error is said to occur when the noisy input falls into any one of these ambiguity regions.) Let $\underline{\mathrm{b}}$ be a binary vector such that

$$\underline{\mathrm{b}} \cdot \underline{\mathrm{h}}_1 = \underline{\mathrm{b}} \cdot \underline{\mathrm{h}}_2,$$

or

$$\underline{\mathrm{b}} \cdot (\underline{\mathrm{h}}_1 - \underline{\mathrm{h}}_2).$$

In other words, $\underline{\mathrm{b}}$ is orthogonal to $\underline{\mathrm{h}}_1 - \underline{\mathrm{h}}_2$. Suppose $\underline{\mathrm{h}}_1$ differs $\underline{\mathrm{h}}_2$ by l bits. Thus $\underline{\mathrm{h}}_1 - \underline{\mathrm{h}}_2$ has only l non-zero components. To count the number of ambiguous regions,

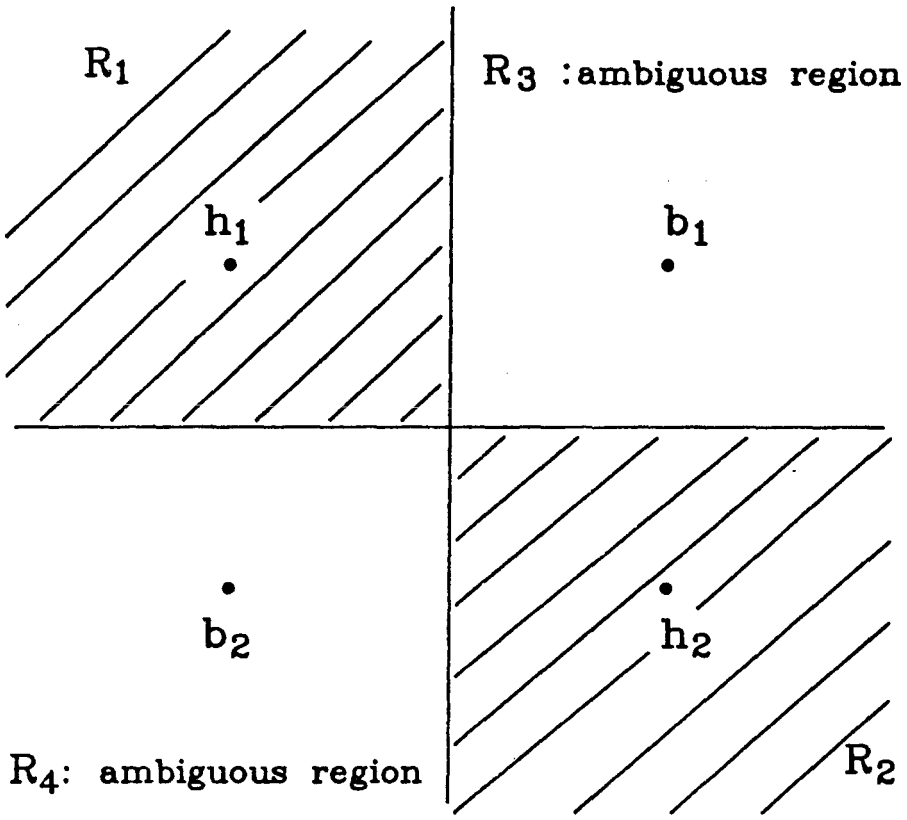


Figure II.2.2.1. Decision regions of a binary input binary filter in two-dimensional space.

we can therefore confine our attention to the l dimensional subspace projected by the l non-zero components of $\underline{h}_1 - \underline{h}_2$. To every binary vector in a l dimensional space, there are only $2(l-1)$ orthogonal binary vectors. (The factor 2 comes from the fact that if $\underline{r}$ is orthogonal to $\underline{s}$, so is $-\underline{r}$.) Since l is upperbounded by N^2, the maximum possible number of ambiguous regions is $2(N^2-1)$ which is negligible compared to 2^{N^2}, the total number of $R's$, for large N^2.

Since the joint probability density function of the noise is gaussian, it is also symmetric about the origin. This means no R is more preferable to the others. It is therefore safe to neglect the contribution of the ambiguity regions to the probability of error. The probability of error is then equal to

$$P_\epsilon = \frac{1}{2}\Big\{ P(f+n \in R_1 | f \in \Omega_2) + P(f+n \in R_2 | f \in \Omega_1) \Big\} \qquad (II.2.2.4.)$$

Here, we assume that $P(\Omega_1) = P(\Omega_2) = 1/2$. The additive noise has no effect on the $N^2 - l$ components for which h_1 and h_2 are identical. It is because the contributions of those $N^2 - l$ components of b to $b \cdot h_1$ and $b \cdot h_2$ are the same. As a result, only l noise components need to be considered. We may thus ignore those $N^2 - l$ bits of h_1 and h_2 in the evaluation of the probability of error. Let h_1' and h_2' be the l sub-parts of h_1 and h_2 respectively. Note that $h_1' = -h_2'$. By invoking the symmetry of the probability density functions (which are gaussian) of the noise components, and the complementary property of h_1' and h_2', we have

$$P(f+n \in R_1 | f \in \Omega_2) = P(f+n \in R_2 | f \in \Omega_1)$$

We may express P_ϵ as follows:

$$\begin{aligned} P_\epsilon &= P(f+n \in R_1 | f \in \Omega_2) \\ &= \int \cdots \int_{R_1} G(\{n(i,j)\}; \{f_2(i,j)\}) d\{n(i,j)\}. \end{aligned}$$

The l dimensional subspace has a total of 2^l $R's$. That means R_1 has 2^{l-1} $R's$. (It also means the L.H.S. of the above expression for the probability of error is a sum of 2^{l-1} integrals. For each of these R, the range of integration of each dummy variable (l of them) may either be $R_1^- = (-\infty, 0]$ or $R_1^+ = [0, \infty)$. Without loss of generality, assume $\beta[f_2(i,j)] = 1, \forall i,j$. Under this assumption, any R in R_1 must have more R_1^- than R_1^+. Otherwise, the binary vector representing that particular R would have a greater d ot product with h_2 than h_1 (in which case there is no error). Of the 2^{l-1} $R's$, there are one R with no R_1^+ and l R_1^-; $\binom{l}{1}$ sign-ants with 1 R_1^+ and $\binom{l}{l-1}$ R_1^-; $\cdots$; $\binom{l}{m}$ sign-ants with m R_1^+, and $\binom{l}{l-m}$ R_1^- where m is the greatest integer less than l. Since the noise components are statistically independent, we may consider the contributions due to individual components separately. For the range of integration of a dummy variable equals to R_1+, the contribution of that noise component is given by

$$P_{i,j}^+ = \int_0^\infty \frac{1}{\sqrt{2\pi}\sigma_n} exp(-\frac{(\alpha_{i,j} - f_2(i,j))^2}{2\sigma_n^2})\, d\alpha_{i,j} \qquad (II.2.2.5a.)$$

For R_1^-, the contribution of the corresponding component is given by

$$P_{i,j}^- = \int_{-\infty}^0 \frac{1}{\sqrt{2\pi}\sigma_n} exp(-\frac{(\alpha_{i,j} - f_2(i,j))^2}{2\sigma_n^2})\, d\alpha_{i,j} \qquad (II.2.2.5b.)$$

The probability of error is therefore

$$P_\epsilon = \sum^{2^{l-1}} \prod_{i,j}^{l} P_{i,j}^+ P_{i,j}^- . \qquad (II.2.2.6.)$$

In general, the above expression is tedious to evaluate. However, we may gain some insight by assuming $\{f_2(i,j)\}$ to be independent gaussian random variables with mean zero and variance σ_f^2, like we did in previous sections. Under the condition

that $\beta[f_2(i,j)] = 1$, the probability density function of $f_2(i,j)$ is

$$p_f(\gamma) = \frac{2}{\sqrt{2\pi}\sigma_f} exp(-\frac{\gamma^2}{2\sigma_f^2})$$

i.e., γ can only be positive. Using this statistical assumption on $f_2(i,j)$, we have

$$P_{i,j}^{+} = P^{+} = \int_0^{\infty} \left[\int_0^{\infty} \frac{1}{\sqrt{2\pi}\sigma_n} exp(-\frac{(\alpha-\gamma)^2}{2\sigma_n^2})\, d\alpha \right] \frac{1}{\sqrt{2\pi}\sigma_f} exp(-\frac{\gamma^2}{2\sigma_f^2})\, d\gamma \qquad (II.2.2.7a.)$$

and similarly

$$P_{i,j}^{-} = P^{-} = \int_{-\infty}^{0} \left[\int_{-\infty}^{0} \frac{1}{\sqrt{2\pi}\sigma_n} exp(-\frac{(\alpha-\gamma)^2}{2\sigma_n^2})\, d\alpha \right] \frac{1}{\sqrt{2\pi}\sigma_f} exp(-\frac{\gamma^2}{2\sigma_f^2})\, d\gamma \qquad (II.2.2.7b.)$$

The terms inside the brackets in the above equations are the probabilities conditioned on γ, whereas the integrand outside the brackets is the probability density function of γ. Note that the expressions in eq.(II.2.2.7a.) and eq.(II.2.2.7b.) are similar to that in eq.(II.2.1.4.). We may follow the same procedure from eq.(II.2.1.5.) to eq.(II.2.1.7.), i.e., first differentiate the RHS of the expressions with respect to σ_n, then evaluate the integrals, and finally integrating with respect to σ_n and matching the boundary conditions, and show that

$$P^{+} = 1 - \frac{1}{\pi} tan^{-1}(SNR_{in}^{-1}) \qquad (II.2.2.8a.)$$

and

$$P^{-} = \frac{1}{\pi} tan^{-1}(SNR_{in}^{-1}) \qquad (II.2.2.8b.)$$

where $SNR_{in} = \sigma_f/\sigma_n$. We may now analyze the asymptotic behavior by substituting eq.(II.2.2.8.) into eq.(II.2.2.6.).

For small SNR_{in}, i.e., $\sigma_n \gg \sigma_f$,

$$P^- \rightarrow \frac{1}{2} \qquad P^+ \rightarrow \frac{1}{2}$$

Eq.(II.2.2.6.) becomes

$$\begin{aligned} P_\epsilon &= (\frac{1}{2})^l + \binom{l}{1}(\frac{1}{2})^l + \cdots + \binom{l}{m}(\frac{1}{2})^l \\ &\approx \frac{1}{2}(1+1)^l(\frac{1}{2})^l \\ &= \frac{1}{2} \end{aligned} \qquad (II.2.2.9.$$

This result agrees with intuition: when the noise level is high, using the machine to classify two patterns is just as good as flipping a coin.

For large SNR_{in}, i.e., $\sigma_n \ll \sigma_f$,

$$P^- \rightarrow 0 \qquad P^+ \rightarrow 1$$

Eq.(II.2.2.6.) can be approximated by the following expression:

$$\begin{aligned} P_\epsilon &\approx \binom{l}{m}(P^+)^m(P^-)^{l-m} \\ &\approx \binom{l}{m}(P^-)^{l-m} \end{aligned} \qquad (II.2.2.10.)$$

Substituting $m \approx l/2$ for large l and invoking Stirling's formula, the above equation becomes

$$P_\epsilon \approx \sqrt{\frac{2}{\pi l}}(\frac{2}{\pi})^{\frac{l}{2}}(2tan^{-1}(SNR_{in}^{-1}))^{\frac{l}{2}} \qquad (II.2.2.11.)$$

which tends to 0 as either $SNR_{in} \rightarrow \infty$ or $l \rightarrow \infty$.

II.3. Binary Fourier Plane Correlator

The correlation $g(x', y')$ between $f(x, y)$ and $h(x, y)$ is given by eq.(II.0.1.) The correlation theorem states that the Fourier transform of $G(\xi, \eta)$ is given by the product of $F(\xi, \eta)$, the fourier transform of f, and $H^*(\xi, \eta)$, the complex conjugate of the fourier transform of h:

$$G(\xi, \eta) = F(\xi, \eta)H^*(\xi, \eta)$$

where

$$F(\xi, \eta) = \int \int f(x, y)e^{-j2\pi(x\xi+y\eta)}\, dxdy$$

$$H(\xi, \eta) = \int \int h(x, y)e^{-j2\pi(x\xi+y\eta)}\, dxdy \qquad (II.3.1.)$$

In accordance,

$$g(x\prime, y') = \int \int G(\xi, \eta)e^{j2\pi(x'\xi+y'\eta)}\, d\xi d\eta \qquad (II.3.2.)$$

We can see from eq.(II.3.1.) that the reference filter of a correlator can be realized in the fourier (frequency) plane. In such case, $H^*(\xi, \eta)$ is known as the frequency plane filter of which $h(x, y)$ is the impulse response.

From eq.(II.3.1.), the matched filter in Fourier plane is expressed as

$$H^*(\xi, \eta) = F^*(\xi, \eta)$$

When f is applied to the correlator, the autocorrelation at the output plane can be expressed as

$$g(x', y') = \int \int |F(\xi, \eta)|^2 e^{j2\pi(x'\xi+y'\eta)}\, d\xi d\eta \qquad (II.3.3.)$$

Suppose we want to modify the frequency plane filter by performing a thresholding operation on the filter. Immediately, we are confronted with the problem

that $H^*(\xi,\eta)$ is in general complex. This means that it is not possible to conserve the sign information of both the real and imaginary components of H by using only one bit of data.

Consider the following algorithm for generating the binary frequency plane filter:

$$\begin{aligned} B(\xi,\eta) &= \beta\Big\{ Re[H^*(\xi,\eta)] \Big\} \\ &= \beta\Big\{ Re[H(\xi,\eta)] \Big\} \end{aligned} \qquad (II.3.5.)$$

In this algorithm, all the information associated with H_i, the imaginary component of H, is discarded. Only the phase information of H_r, the real component of H, is conserved. In the following we examine the consequences of this information loss.

The correlation $g(x',y')$ between B and F can be expressed as

$$\begin{aligned} g(x',y') = &\int\int F_r(\xi,\eta)B(\xi,\eta)e^{j2\pi(x'\xi+y'\eta)}\,d\xi d\eta \\ &+\int\int F_i(\xi,\eta)B(\xi,\eta)e^{j2\pi(x'\xi+y'\eta)}\,d\xi d\eta \end{aligned} \qquad (II.3.6.)$$

where F_r and F_i are the real and imaginary parts of F respectively. If h is chosen to match f, then the first term in the above summation becomes

$$\int\int |F_r(\xi,\eta)|e^{j2\pi(x'\xi+y'\eta)}\,d\xi d\eta$$

By invoking the Schwartz inequality, it can be shown that the above term is always less than

$$\int\int |F_r(\xi,\eta)|d\xi d\eta$$

which is the output correlation at the center. The expected value of the second term in eq.(II.3.6.) is zero, assuming that the real and imaginary components of F are uncorrelated. We therefore have a correlation peak at $x' = 0, y' = 0$. This occurs if and only if $F = H$, and therefore this type of filter exhibits a property useful for pattern recognition similar to the matched filter. We shall call the filter generated by the algorithm described by eq.(II.3.5.) binary fourier (frequency) plane matched filter, or simply as BFMF. To further examine the property of BFMF, we will investigate the PG, SNR and P_ϵ of the correlator in the following sections.

II.3.1. Processing gain

To simplify computations, we will again use the discrete analogs of input, filter and correlation output. We take $f(n,m), n, m = 0, \ldots, N-1$ to be a sample realization of a white, gaussian process. Assume $f(n,m) = 0$ for $N \leq n, m \leq 2N-1$. The DFT of f is defined as

$$F(k,l) = \frac{1}{2N}\Big\{ \sum_{n=0}^{2N-1} \sum_{m=0}^{2N-1} f(n,m) cos(\frac{2\pi}{2N}(nk+ml)) - j \sum_{n=0}^{2N-1} \sum_{m=0}^{2N-1} f(n,m) sin(\frac{2\pi}{2N}(nk+ml)) \Big\} \qquad (II.3.1.1.)$$

Note that the discrete bandwidth of the DFT is taken to be $4N^2$. The reason for so doing is to accommodate the full size of a NxN by NxN correlation which will result in an output pattern of $2N$x$2N$. The first term in eq.(II.3.1.1.) is the real part and the second term is the imaginary part of F. In accordance, the IDFT of

$F(k,l)$ is given as

$$f(n,m) = \frac{1}{2N}\left\{ \sum_{k=0}^{2N-1} \sum_{l=0}^{2N-1} F(k,l) exp(\frac{2\pi}{2N}(nk+ml)) \right\}. \qquad (II.3.1.2.)$$

Here we analyze the statistics of F_r and F_i. The linear transformation of a sequence of jointly gaussian random variables will always result in another sequence of jointly gaussian random variables. Because the DFT is a linear transformation, $\{F_r(k,l)\}$ and $\{F_i(k,l)\}$ are jointly gaussian. Furthermore, it can be shown that

$$\begin{aligned} E[F_r(k,l)] &= 0, \\ E[F_i(k,l)] &= 0 \end{aligned} \qquad (II.3.1.3a.)$$

and

$$\begin{aligned} E[F_r(k,l)F_r(k_1,l_1)] &= 0 \qquad k \neq k_1, l \neq l_1 \\ E[F_i(k,l)F_i(k_1,l_1)] &= 0 \qquad k \neq k_1, l \neq l_1 \\ E[F_i(k,l)F_r(k_1,l_1)] &= 0 \qquad \forall k, k_1, l, l_1 \end{aligned} \qquad (II.3.1.3b.)$$

which means $\{F_r(k,l)\}$ and $\{F_i(k,l)\}$ are statistically independent since they are jointly gaussian. It can also be shown that

$$E[F_r^2(k,l)] = \sigma_r^2 = \begin{cases} \frac{\sigma_f^2}{2} & \text{for } (k,l) = (0,0),(N,N); \\ \frac{\sigma_f^2}{4} & \text{otherwise} \end{cases} \qquad (II.3.1.4a.)$$

and

$$E[F_i^2(k,l)] = \sigma_r^2 = \begin{cases} 0 & \text{for } (k,l) = (0,0),(N,N); \\ \frac{\sigma_f^2}{4} & \text{otherwise} \end{cases} \qquad (II.3.1.4b.)$$

The correlation $g(n', m')$ is equal to

$$g(n', m') = \frac{1}{2N}\Big\{ \sum_{k=0}^{2N-1} \sum_{l=0}^{2N-1} G(k,l) exp(\frac{2\pi}{2N}(n'k + m'l)) \Big\}$$

where

$$\begin{aligned} G(k,l) &= B(k,l)F(k,l) \\ &= |F_r(k,l)| + jB(k,l)F_i(k,l) \end{aligned} \qquad (II.3.1.5.)$$

Recall that

$$B(k,l) = \beta[F_r(k,l)].$$

The first term on the RHS of the above equation is the signal term. The second term is considered to be the noise term because no information of F_i is stored in B so the second term can add an undesirable effect to the output. The mean of $G(k,l)$ can be evaluated as follows:

$$\begin{aligned} E[G(k,l)] &= E[|F_r(k,l)|] + jE[B(k,l)]E[F_i(k,l)] \\ &= E[|F_r(k,l)|]. \end{aligned} \qquad (II.3.1.6.)$$

Here we used the fact that B and F_i are uncorrelated. The probability density function of $|F_r|$ is a one-sided gaussian distribution with variance σ_r^2 given by eq.II.3.1.4a. Hence

$$\begin{aligned} E[|F_r(k,l)|] &= \int_0^\infty \sqrt{\frac{2}{\pi\sigma_r^2}} exp(-\frac{\alpha^2}{2\sigma_r^2}) \alpha d\alpha \\ &= \sqrt{\frac{2}{\pi}}\sigma_r \end{aligned} \qquad (II.3.1.7.)$$

Substituting the result in eq.(II.3.1.4a.) and the above equation into eq.(II.3.1.6.), we have

$$E[G(k,l)] = \begin{cases} \frac{\sigma_f}{\sqrt{\pi}} & \text{for } (k,l) = (0,0),(N,N); \\ \frac{\sigma_f}{\sqrt{2\pi}} & \text{otherwise.} \end{cases} \qquad (II.3.1.8.)$$

The expected value of $g(n',m')$ is equal to the expected value of the inverse fourier transform of $G(k,l)$:

$$\begin{aligned} E[g(n',m')] &= \frac{1}{2N}\Big\{ \sum_{k=0}^{2N-1} \sum_{l=0}^{2N-1} E[G(k,l)]\, exp(\frac{2\pi}{2N}(n'k+m'l)) \\ &= \frac{1}{2N} E[G(k,l)] \Big[\sum_{k=0}^{2N-1} exp(\frac{2\pi}{2N}n'k)\Big] \Big[\sum_{l=0}^{2N-1} exp(\frac{2\pi}{2N}m'l)\Big] \end{aligned}$$

Using eq.(II.3.1.8.), the above expression can also be written as

$$\begin{aligned} E[g(n',m')] = \frac{1}{2N}\Big\{ & \frac{\sigma_f}{\sqrt{2\pi}} \sum_{k=0}^{2N-1} exp(\frac{2\pi}{2N}n'k)\Big[\sum_{l=0}^{2N-1} exp(\frac{2\pi}{2N}m'l)\Big] \\ & + \sigma_f \frac{(\sqrt{2}-1)}{\sqrt{2\pi}}(1+exp(\frac{j2\pi}{2Nm'l}) \sum_{k=0}^{2N-1} exp(\frac{2\pi}{2N}n'k) \\ & + \sigma_f \frac{(\sqrt{2}-1)}{\sqrt{2\pi}}(1+exp(\frac{j2\pi}{2Nn'k}) \sum_{l=0}^{2N-1} exp(\frac{2\pi}{2N}m'l) \\ & + \sigma_f \frac{(\sqrt{2}-1)}{\sqrt{2\pi}}(1+exp(\frac{j2\pi}{2Nm'l})\sigma_f \frac{(\sqrt{2}-1)}{\sqrt{2\pi}}(1+exp(\frac{j2\pi}{2Nn'k})\Big\} \end{aligned} \qquad (II.3.1.9.)$$

In view of the fact that

$$\sum_{k=0}^{2N-1} exp(\frac{2\pi}{2N}n'k) = \begin{cases} 2N & \text{if } n' \text{ is a multiple of 2N;} \\ 0 & \text{otherwise} \end{cases}$$

the second, third, and fourth term on the RHS of eq.(II.3.1.9.) are negligible compared to the first term for large enough N. As a result

$$E[g(n',m')] = \begin{cases} \sqrt{\frac{2}{\pi}} N\sigma_f & \text{for } (n',m') = (0,0); \\ 0 & \text{otherwise.} \end{cases} \qquad (II.3.1.10.)$$

The expected value of $G^2(k,l)$ may be obtained as follows:

$$\begin{aligned} E[G^2(k,l)] &= E\Big[\Big||F_r(k,l)| + jB(k,l)F_i(k,l)\Big|^2\Big] \\ &= E\Big[F_r^2(k,l)\Big] + E\Big[B^2(k,l)F_i^2(k,l)\Big] \\ &= \frac{\sigma_f^2}{2} \end{aligned}$$

where the last equality is a result of applying eq.II.3.1.4. The expected value of the square of $g(n',m')$ can be expressed as

$$\begin{aligned} E\Big[g^2(n',m')\Big] &= \frac{1}{4N^2}\sum_{k=0}^{2N-1}\sum_{l=0}^{2N-1}\sum_{k_1=0}^{2N-1}\sum_{l_1=0}^{2N-1} E\Big[G(k,l)G(k_1,l_1)\Big] \\ &\quad \cdot exp(\frac{j2\pi}{2N}(n'k+m'l))exp(\frac{-j2\pi}{2N}(n'k_1+m'l_1)) \\ &= \frac{1}{4N^2}\sum_{k=0}^{2N-1}\sum_{l=0}^{2N-1} E\Big[G^2(k,l)\Big]exp(\frac{2\pi}{2N}(n'k+m'l)) \\ &= \frac{1}{4N^2}\sum_{k=0}^{2N-1}\sum_{l=0}^{2N-1}\frac{\sigma_f^2}{2} \\ &= \frac{\sigma_f^2}{2} \qquad (II.3.1.11.) \end{aligned}$$

Substituting eq.(II.3.1.10.) and eq.(II.3.1.11.) into eq.(II.1.1.3.), we have the processing gain, PG_3, of the BFMF:

$$PG_3 = \frac{N^2}{\sqrt{2\pi}}$$

Comparing with PG_1 given by eq.II.1.1.10.,

$$PG_3 = 1/over2PG_1.$$

The explanation is straightforward: half of the information in f is discarded in formation of the BFMF. This is so because we assumed that the information content

in f_s is the same as in f_a. PG_3 will be identical to PG_1 in the limiting situation $f = f_s$, i.e., no phase information will be lost in obtaining the BFMF.

II.3.2. Signal-to-Noise Ratio

When a sampled white gaussian white noise pattern, $\eta(n,m)$, is added to $f(n,m)$, the pattern to be recognized, the distorted input image becomes

$$s(n,m) = f(n,m) + \eta(n,m)$$

Suppose the fourier transform of $\eta(n,m)$ is $\mathcal{N}(k,l)$. When s is applied to the BFMF given by eq.(II.3.5.), the fourier transform of the output is given as

$$G(k,l) = F(k,l)B(k,l) + \mathcal{N}(k,l)B(k,l) \qquad (II.3.2.1.)$$

The linearity of the filter allows us to consider the first and the second terms in the above equation separately. In other words, the output due to the sum of f and η is equal to the sum of the outputs due to f and η individually. The statistics of the output due to f have been analyzed in the last section. We now examine the effect of the noise on the output.

Let

$$G_{\mathcal{N}}(k,l) = \mathcal{N}(k,l)B(k,l)$$

$$G_F(k,l) = F(k,l)B(k,l)$$

and

$$g_\eta(n',m') = FT^{-1}[G_{\mathcal{N}}(k,l)]$$

$$g_f(n',m') = FT^{-1}[G_F(k,l)]$$

The mean of $G_{\mathcal{N}}(k,l)$ is equal to

$$E[G_{\mathcal{N}}(k,l)] = E[\mathcal{N}(k,l)]\,[H(k,l)] = 0$$

Thus the mean of $g_\eta(n',m')$ is equal to

$$E[g_\eta(n',m')] = FT^{-1}\Big\{ E[G_{\mathcal{N}}(k,l)] \Big\} = 0 \qquad (II.3.2.2.)$$

Since $\mathcal{N}$ and F are uncorrelated (f and η are uncorrelated), the variance of g is equal to the sum of the variances of g_η and g_f. The variance of g_f is given by eq.(II.3.1.1.) The variance of g_η can be computed as follows:

$$\begin{aligned}
&E[g_\eta^2(n',m')] \\
&= \frac{1}{4N^2} \sum_{k=0}^{2N-1} \sum_{l=0}^{2N-1} \sum_{k_1=0}^{2N-1} \sum_{l_1=0}^{2N-1} E[\mathcal{N}(k,l)B(k,l)\mathcal{N}^*(k_1,l_1)B(k_1,l_1)] \\
&\quad \cdot exp(\frac{j2\pi}{2N}(n'k+m'l))exp(\frac{-j2\pi}{2N}(n'k_1+m'l_1)) \\
&= \frac{1}{4N^2} \sum_{k=0}^{2N-1} \sum_{l=0}^{2N-1} E[|\mathcal{N}(k,l)|^2] \\
&\quad + \frac{1}{4N^2} \sum_{k=0}^{2N-1} \sum_{l=0}^{2N-1} \sum_{\substack{k_1=0 \\ k_1\neq k}}^{2N-1} \sum_{\substack{l_1=0 \\ l_1\neq l}}^{2N-1} E[\mathcal{N}(k,l)B(k,l)\mathcal{N}^*(k_1,l_1)B(k_1,l_1)] \\
&\quad \cdot exp(\frac{j2\pi}{2N}(n'k+m'l))exp(\frac{-j2\pi}{2N}(n'k_1+m'l_1)) \\
&= E[|\mathcal{N}(k,l)|^2]
\end{aligned}$$

where

$$\begin{aligned}
E[|\mathcal{N}(k,l)|^2] &= \frac{1}{4N^2} \sum_{n=0}^{2N-1} \sum_{m=0}^{2N-1} \sum_{n_1=0}^{2N-1} \sum_{m_1=0}^{2N-1} E[\eta(n,m)\eta(n_1,m_1)] \\
&\quad \cdot exp(\frac{j2\pi}{2N}(nk+ml))exp(\frac{-j2\pi}{2N}(n_1k+m_1l)) \\
&= \frac{1}{4N^2} \sum_{n=0}^{2N-1} \sum_{m=0}^{2N-1} E[\eta^2(n,m)] \\
&= \sigma_n^2 \qquad (II.3.2.3.)
\end{aligned}$$

Substituting eq.(II.3.2.3.) and eq.II.(3.1.10.) into eq.(II.1.2.2.) the signal-to-noise ratio of the BFMF is given as

$$SNR_3 = \frac{N^2}{\sqrt{2\pi}} \frac{SNR_{in}^2}{1 + SNR_{in}^2}$$

where $SNR_{in} = \sigma_f / \sigma_n$. Note that SNR_3 is one half of SNR_1.

II.3.3. The Probablity of Error of Two-pattern Classification

We will consider the probability of error of the classification of two patterns in this section. Two BFMFs, each corresponding to the pattern to be matched, will be used to implement two correlators. A noisy pattern, which may belong to either Ω_1 or Ω_2, is applied to these two correlators. Again, only the data point at the center of the output plane will be monitored. The two output data points of the two correlators are then compared in magnitude. The input is classified to $\Omega_m, m = 1, 2$ if and only if the output data point of correlator m is maximum. The effect of the noise is such that the output data point of correlator $m (m = 1, 2)$ is maximum when the input is actually a distorted member of $\Omega_{m'}$ $(m' = 2, 1)$. An error is declared if this happens. An expression for the probability of error will be derived.

Since the binary filter is realized in the frequency plane, we will examine the probability of error by considering f, h, and n in the Fourier domain. Let $\underline{\mathbf{F}}_1$ and $\underline{\mathbf{F}}_2$ be the Fourier transforms of the two patterns to be recognized. Let $\underline{\mathbf{H}}_1$ and $\underline{\mathbf{H}}_2$ be the BFMFs generated from $\underline{\mathbf{F}}_1$ and $\underline{\mathbf{F}}_2$ according to eq.(II.3.1.1). Suppose we wish to identify a distorted pattern $\underline{\mathbf{f}}^{(\prime)}$. $\underline{\mathbf{f}}^{(\prime)}$ can be modeled as a pattern $\underline{\mathbf{f}}$ ($\underline{\mathbf{f}}$ may

be an element of either Ω_1 or Ω_2) contaminated by an additive gaussian white noise with mean zero and variance σ_n^2:

$$\underline{\mathbf{f}}^{(\prime)} = \underline{\mathbf{f}} + \underline{\mathbf{n}} \qquad (II.3.3.1.)$$

Let $\underline{\mathbf{F}}^{(\prime)}$, $\underline{\mathbf{F}}$, and $\underline{\mathcal{N}}$ be the Fourier transforms of $\underline{\mathbf{f}}^{(\prime)}$, $\underline{\mathbf{f}}$, and $\underline{\mathbf{n}}$ respectively. The output data points are given by the inverse fourier transform of the products of $\underline{\mathbf{F}}^{(\prime)}$ and $\underline{\mathbf{H}}_1$, and $\underline{\mathbf{f}}^{(\prime)}$ and $\underline{\mathbf{H}}_2$, respectively. But since only the center pixel of the output plane are detected, the effective output points become

$$g_1(0,0) = \underline{\mathbf{F}}^{(\prime)} \cdot \underline{\mathbf{H}}_1 \qquad (II.3.3.2a.)$$

and

$$g_2(0,0) = \underline{\mathbf{F}}^{(\prime)} \cdot \underline{\mathbf{H}}_2 \qquad (II.3.3.2b.)$$

Recall that $\underline{\mathbf{H}}_m, m = 1,2$ is obtained by thresholding the real parts of $\underline{\mathbf{F}}_m\, m = 1,2$. It can be shown that the real parts of the fourier transform of any real signal are symmetric. It follows that $\underline{\mathbf{H}}_m$ is also symmetric. It can also be shown that the imaginary parts of the fourier transform of any real siganl is antisymmetric. Since the dot product of a symmetric signal and an antisymmetric signal is always zero, we may rewrite eq.(II.3.3.2.) as

$$g_1(0,0) = \underline{\mathbf{F}}'_r \cdot \underline{\mathbf{H}}_1 \qquad (II.3.3.3a.)$$

and

$$g_2(0,0) = \underline{\mathbf{F}}'_r \cdot \underline{\mathbf{H}}_2 \qquad (II.3.3.3b.)$$

where $\underline{\mathbf{F}}'_r$ is the real part of $\underline{\mathbf{F}}^{(\prime)}$. In view of eq.(II.3.3.1.)

$$\underline{\mathbf{F}}'_r = \underline{\mathbf{F}}^{(r)} + \underline{\mathcal{N}}^{(r)}$$

where $\underline{\mathbf{F}}_r$ and $\underline{\mathcal{N}}^{(r)}$ are the real parts of $\underline{\mathbf{F}}$ and $\underline{\mathcal{N}}$, respectively.

The decision surface is defined to be the surface at which

$$g_1(0,0) - g_2(0,0) = 0$$

We call this surface P3. By equation (II.3.3.2.), P3 can also be defined by the expression

$$(\underline{\mathbf{H}}_1 - \underline{\mathbf{H}}_2) \cdot \underline{\mathbf{F}}'_r = 0 \qquad (II.3.3.4.)$$

We shall denote the unit vector in the direction of $(\underline{\mathbf{H}}_1 - \underline{\mathbf{H}}_2)$ by $\underline{\mathbf{H}}_u$. $\underline{\mathbf{F}}'_r$ is classified to Ω_1 if it is on the positive side of P3, i.e., $g_1(0,0) > g_2(0,0)$. $\underline{\mathbf{F}}'_r$ is classified to Ω_2 if it is on the negative side of P3. An error occurs when the addition of $\underline{\mathcal{N}}^{(r)}$ takes $\underline{\mathbf{F}}^{(r)}$ to the opposite side of P3.

In retrospect, the formalism defined in this section bears a one-to-one correspondence to that in section II.1.3: $\underline{\mathbf{H}}_m$ $m = 1,2$, corresponds to $\underline{\mathbf{h}}_m$; $\underline{\mathbf{F}}'_r$ corresponds to $\underline{\mathbf{f}}^{(l)}$; $\underline{\mathcal{N}}_r$ corresponds to $\underline{\mathbf{n}}$; and P3 corresponds to P1. Because of this resemblance, the results in section II.1.3 are also applicable here. The expression for $P_{\epsilon 1}$ described in eq.(II.1.3.12.) can therefore be adapted to express the probability of error here. With minimal modification, the probability of error of the BFMF is given as

$$P_{\epsilon 3} = \frac{1}{2}\int_{-\infty}^{0} \frac{1}{\sqrt{2\pi}\sigma_{np}} exp\Big(-\frac{(\alpha-\mu_1)^2}{2\sigma_{np}^2}\Big) + \frac{1}{2}\int_{0}^{\infty} \frac{1}{\sqrt{2\pi}\sigma_{np}} exp\Big(-\frac{(\alpha-\mu_2)^2}{2\sigma_{np}^2}\Big)$$

$$(II.3.3.5.)$$

where μ_m, $m = 1,2$, is the projection of $\underline{\mathbf{F}}_{rm}$, $m = 1,2$, onto $\underline{\mathbf{H}}_u$ and σ_{np} is the variance of the projection of $\underline{\mathcal{N}}_r$ onto $\underline{\mathbf{H}}_u$. In addition to having an expression

for $P_{\epsilon 3}$, the conclusion drawn in section II.1.3 must also be valid here too, i.e., in general, $P_{\epsilon 3} \rightarrow 0$ as $P_{\epsilon m} \rightarrow 0$.

II.4. Conclusion

We have analyzed the effect of thresholding the image plane matched filter, the input pattern, and the fourier plane matched filter. The criteria for performance used in the analyses were processing gain, signal-to-noise ratio, and probability of error in a two-pattern classification problem. Under the influence of additive gaussian white noise, the performance degradation in the various situations examined was shown to be minimal. The various correlators analyzed, in general, exhibit similar properties as demonstrated by the conventional matched filter. From the evaluations of all the objective performance measures for these correlators, it was clear that gray levels are unimportant compared to the degree of freedom (the discrete space-bandwidth product). Given enough pixels to work with, we may conclude that thresholding the input and the matched filter do not induce great loss in performance.

References for Chapter II

1. A. Vander Lugt, "Signal Detection by Complex Spatial Filtering," *IEEE Trans Inf Th,* **IT-10**, 139 (1964).
2. G. Turin,"An Introduction to Matched Filters," *IRE Trans. Inform. Theory,* **IT-6**, 311 (1960).
3. B. Cooper, "Correlators With Two-Bit Quantization," *Aust J. of Phys,* **23**, 521 (1969).
4. A. Oppenheim, "The Importance of Phase in Signals," *Proc. IEEE,* **69**, 529 (1981).
5. T. Huang and J. Burnett, "The Importance of Phase in Image Processing Filters," *IEEE Trans. Acoust., Speech Signal Processing,* **ASSP-23**, 167 (1979).
6. A. Papoulis, *Probability, Random Variables, and Stochastic Processes,* McGraw-Hill, New York (1965).
7. J. Lee, *Numerical Analysis for Computers,* Reinhold Publishing Corp., New York (1966).
8. J. M. Wozencraft and I. M. Jacobs, *Principles of Communication Engineering,* Wiley and Sons, New York (1965).
9. A. Oppenheim and R. Schafer, *Digital Signal Processing,* Prentice-Hall, New Jersey (1975).
10. W. Feller, *An Introduction to Probability Theory and Its Applications,* Wiley and Sons, New York (1970).
11. S. Venkatesh, "Linear Maps with Point Rules: Applications to Pattern Classification and Associative Memory," *Ph.D Dissertation,* California Institute of Technology (1987).

12. D. Psaltis, E. Paek, and S. Venkatesh, "Optical Image Correlation with a Binary Spatial Light Modulator," *Opt. Eng.*, **23**, 698 (1984).

III. Capacity of Optical Correlators

III.0 Introduction

In the field of pattern recognition, there are many applications where categorizing of dissimilar objects (small mutual inner products) into the same class are desirable. Some examples are: recognition of the same object of different scales, orientations, and perspective angles. Usually, the solutions are application dependent. However, a good general concept can still be obtained by considering the two-category classification problem.

A two-category classifier can be described as follows. Suppose we have a set of dissimilar objects. Each one of the objects is assigned to one of two categories arbitrarily. Our goal here is to classify all the objects correctly without any a priori knowledge of their true identities. Any object of the set can be an input to the classifier. A binary number (0 or 1) returned by the classifier in turn determines which of the two classes the object belongs to.

One measure of performance is the number of dissimilar objects that a two-category classifier can handle without exceeding a certain margin of error. It is generally true that as a classifier tries to handle more objects, the accuracy of the decision of the classfier degrades. We call this performance measure of a classifier its *capacity* or M_c. We define M_c to be the maximum number of arbitrarily selected dissimilar objects that a classifier can identify with an arbitrarily small probability of error, given the degree of freedom, N (the number of input pixels), is large

enough.

Note that a C-category classifier can be built by adjoining K (a function of C; discussed in the following) 2-category classifiers. So instead of one output plane, there will be K output planes. Each output plane will result in one binary bit (1 or 0) when any pattern is input to the classifier. In essence, the output of the C-category classifier can be thought of as a K-bit binary vector. Since 2^K distinct numbers can be represented, a minimum of $K = log_2 C$ 2-category classifiers is required to arbitrarily classify any M objects into C categories. A 2-category classifier can also be viewed as a one-bit output associative memory (a pattern resembling any pattern stored will result in the same response). By the same token, a C-category classifier can be viewed as a K-bit output holographic memory. The classification capacity M_c, discussed earlier, therefore, can also be used to characterize the storage capacity of such a K-bit output memory.

In this chapter, we will investigate the storage capability of a 2-category classifier implemented by a correlation filter. Heuristically, the capacity of a classifier depends on a number of factors:

i. the degrees of freedom, N;

ii. the statistics of the objects;

iii. the algorithm of generating the correlation filter; and

iv. the complexity of the processing unit (an array of neurons) at the output plane, e.g., the capacity would be higher if the output neurons are of the multi-level thresholding type.

In the following sections, we will study what we refer to as the multiple exposure

correlator (MEC). A MEC is a correlator having the multiple exposure hologram (MEH) as its reference filter. MEH is generated as follows:

$$\underline{\mathrm{w}} = \sum_{m} \underline{\mathrm{f}}^{(m)}$$

where $\{\underline{\mathrm{f}}^{(m)}\}$ is the set of patterns to be stored and $\underline{\mathrm{w}}$ is the weight vector of the MEH. The capacity of the MEC when the output is confined to one pixel (center pixel) will be derived. In such case, the correlator is not shift-invariant, since all input patterns must be aligned at the origin. When more output pixels are monitored, the capacity of the MEC is expected to decrease. The capacity of the MEC, when translated input patterns are also allowed, will also be derived. And the reduction in capacity, due to the incorporation of shift-invariance into the correlator, will be addressed. When the MEH is thresholded to ± 1, we have a binary multiple exposure hologram, or BMEH. The storage capacity of the binary multiple exposure correlator (BMEC) is expected to be lower than that of the MEC. The capacities (with and without shift-invariance) of the BMEC will also be derived.

III.1 Inner-product Filter

III.1.0 Basic Model

The output of the inner-product filter, g, is given as

$$g = \underline{\mathrm{w}} \cdot \underline{\mathrm{f}} \qquad (III.1.0.1)$$

where $\underline{\mathrm{w}}$ is the reference pattern, and $\underline{\mathrm{f}}$ is the input pattern. At the output, g is

subjected to thresholding. Whether g exceeds the threshold level or not determines to which class $\underline{\mathbf{f}}$ belongs.

Eq(III.1.0.1) is also known as a linear discriminant function. It was shown by Cover that any classifier which implements a linear discriminant function has a storage capacity of $2N$. It can also be shown that a linear discriminator which stores vectors within its capacity can always be obtained through some training algorithms using the vectors to be stored as the training set. The perceptron training algorithm and the adaline are examples of such training algorithms. It can be shown by inspecting these two algorithms that a trained weight vector is a linear combination of the training vectors:

$$\underline{\mathbf{w}} = \sum_m a_m \underline{\mathbf{f}}^{(m)} \qquad (III.1.0.2)$$

However, algorithms such as the perceptron and adaline typically require many iterations to train a weight vector. Therefore, a simpler and special case of the previous algorithm is often used:

$$a_m = \begin{cases} 1 & \text{if } \underline{\mathbf{f}}^{(m)} \text{ belongs to class 1;} \\ 0 & \text{if } \underline{\mathbf{f}}^{(m)} \text{ belongs to class 2.} \end{cases} \qquad (III.1.0.3)$$

As a matter of fact, the above equation is a precise definition of the MEH. Physically, the reference hologram can be realized by multiply- exposing a hologram to all the patterns in class 1. In this section, the capacity of the multiple exposure inner-product filter will be derived. We shall begin the derivation with some formulation.

III.1.1. Definitions and Assumptions

Let

$$\left\{ \begin{array}{ll} \Omega_1 = \{\underline{\hat{\mathbf{f}}}^{(\hat{m})}\} & \hat{m} = 1, \ldots, M, \\ \Omega_2 = \{\underline{\check{\mathbf{f}}}^{(\check{m})}\} & \check{m} = 1, \ldots, M \end{array} \right\}$$

be two sets of images to be classified. For simplicity, these images are assumed to be one-dimensional vectors each with N components. Suppose $\hat{f}_i^{(\hat{m})}$ and $\check{f}_j^{(\check{m})}$ are the *i-th* and the *j-th* component of $\underline{\hat{\mathbf{f}}}^{(\hat{m})}$ and $\underline{\check{\mathbf{f}}}^{(\check{m})}$, respectively, where $i, j = 1, \ldots, N$. Assume $\{\}$ to be statistically independent random variables, i.e., $\hat{f}_i^{(\hat{m})}$ and $\check{f}_j^{(\check{m})}$ are dependent if and only if the superscripts and the subscripts are identical (in which case, the two random variables are identical). Furthermore, assume that they have identical symmetric probability density functions with mean 0 and variance equal to σ^2.

III.1.2 Formulation for the Probability of Correctness

Let $\Phi = \{\underline{\mathbf{f}}^{(m)}\}$, $m = 1, \ldots, 2M$, be the set of input vectors. Define $E_j^{(m)}$ as the event that $\underline{\mathbf{f}}^{(m)}$ is classified to Ω_j, $j = 1$ or 2, given that $\underline{\mathbf{f}}^{(m)}$ actually belongs to Ω_j. Note that $E^{(m)}$ is equivalent to the event $g_0^{(m)} > \tau$, given $\underline{\mathbf{f}}^{(m)} \in \Omega_1$ or the event $g_0^{(m)} < \tau$, given $\underline{\mathbf{f}}^{(m)} \in \Omega_2$ where $g_0^{(m)}$ is the inner-product of $\underline{\mathbf{f}}^{(m)}$ and $\underline{\mathbf{w}}$ and τ is the thresholding level. We define the probability of correctness, P_c, as

$$P_c = P\Big[\bigcap_m E_j^{(m)}\Big], \qquad (III.1.2.1.)$$

i.e., the probability that all vectors in Φ are detected correctly. It can be shown that P_c is independent of the order of which $\underline{\mathbf{f}}^{(m)}$ is presented to the correlator. Hence, without loss of generality, we may assume the first M vectors in Φ to be

originally from Ω_1 while the last M vectors are from Ω_2, i.e., assume

$$\underline{\mathbf{f}}^{(1)} = \hat{\underline{\mathbf{f}}}^{(1)}; \cdots; \underline{\mathbf{f}}^{(M)} = \hat{\underline{\mathbf{f}}}^{(M)},$$

$$\underline{\mathbf{f}}^{(M+1)} = \check{\underline{\mathbf{f}}}^{(1)}; \cdots; \underline{\mathbf{f}}^{(2M)} = \check{\underline{\mathbf{f}}}^{(M)}.$$

And subsequently, the probability of correctness becomes

$$P_c = P(g_0^{(1)} > \tau; \ldots, g_0^{(M)} > \tau; g_0^{(M+1)} < \tau; u \ldots, g_0^{(2M)} < \tau). \qquad (III.1.2.2.)$$

In the following sections, it will be shown that the events $\{E_j^{(m)}\}$ are statistically independent. P_c can then be written as j

$$P_c = \prod_{m=1}^{M} P[E_1^{(m)}] \prod_{m=M+1}^{2M} P[E_2^{(m)}], \qquad (III.1.2.3a.)$$

where

$$P[E_1^{(m)}] = P[g_0^{(m)} > \tau] \quad m = 1, \ldots, M;$$
$$P[E_2^{(m)}] = P[g_0^{(m)} < \tau] \quad m = M+1, \ldots, 2M. \qquad (III.1.2.3b.)$$

The joint cumulative probability functions of $\underline{\mathbf{g}}^{(m)}$ corresponding to the MEC will be shown to be approximately jointly gaussian with diagonal covariance matrices in the limit of large N. P_c can then be calculated using eq.(III.1.2.3.). The capacity of the MEC correlators can be derived by forcing P_c to one in the limit that N goes to infinity.

III.1.3. Multivariate Central Limit Theorem

The multivariate central limit theorem will be used for deriving our results and, for this reason, we begin by quoting the theorem.

Multivariate Central Limit Theorem (MVCLT).

Let $\underline{z} = (z_1, \ldots, z_M)$ be defined as

$$\underline{z} = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} \underline{x}_i.$$

Assume that each $\underline{x}_i$ is a N-component vector that is statistically independent of all others†:

$$p_{\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_N} = \prod_{i}^{N} p_{\underline{x}_i}.$$

Also, assume that each $\underline{x}_i$ has the same density function, say $p_{\underline{x}}$, with mean $\underline{0}$, covariance matrix $\underline{\underline{\Lambda}}_{\underline{x}}$ and characteristic function $F_{\underline{x}}$:

$$\begin{aligned} p_{\underline{x}_i} &= p_{\underline{x}}; \\ E[\underline{x}_i] &= \underline{0}; \\ E[\underline{x}_i{}^t \underline{x}_i] &= \underline{\underline{\Lambda}}_{\underline{x}}; \\ F_{\underline{x}}(\underline{\nu}) &\triangleq \int p_{\underline{x}}(\underline{\alpha}) e^{j\underline{\alpha}\underline{\nu}^t} \, d\underline{\alpha}. \end{aligned}$$

Then

$$\lim_{N \to \infty} F_{\underline{z}}(\underline{\nu}) = exp\left(-\frac{1}{2} \underline{\nu} \underline{\underline{\Lambda}}_{\underline{z}} \underline{\nu}^t \right),$$

---

† The statistical independence of vectors means that any component of any vector $\underline{x}_i$ is statistically independent of all components of the remaining $N - 1$ vectors.

where $\underline{\underline{\Lambda}}_{\underline{z}}$ is the covariance matrix and $F_{\underline{z}}(\nu)$ is the characteristic function of $\underline{z}$:

$$\underline{\underline{\Lambda}}_{\underline{z}} = \underline{\underline{\Lambda}}_{\underline{x}};$$
$$F_{\underline{z}}(\underline{\nu}) \triangleq \int p_{\underline{z}}(\underline{\alpha}) e^{j\underline{\alpha}\underline{\nu}^t} \, d\underline{\alpha}.$$

The probability density function of $\underline{z}$, $p_{\underline{z}}$ is determined by taking the inverse Fourier transform of $F_{\underline{z}}$. As in the case of the one-dimensional central limit theorem, $p_{\underline{z}}$ does not always converge to gaussian form as N gets large. Whenever $p_{\underline{x}}$ does not contain impulses, the convergence occurs. The cumulative function of $\underline{z}$, however, always becomes gaussian (provided $\underline{\underline{\Lambda}}_{\underline{x}}$ exists). Since it is the cumulative function (integration of the probability density function) that we are going to use for deriving the probability of correctness (or error), we need not be concerned with the limitation of the theorem on the probability density function.

III.1.4. Joint Probability Density Function

The output $g_0^{(m)}$ can be expressed as

$$g_0^{(m)} = \frac{1}{\sqrt{N}} \sum_l^M \underline{f}^{(l)^t} \underline{f}^{(m)} \qquad (III.1.4.1)$$

by substituting eq.(III.1.2.1b.) into eq.(III.1.2.1a.). From the definition of $E_j^{(n)}$, it is apparent that to show $\{E_j^{(m)}\}$ to be statistically independent is equivalent to showing that $\{g_0^{(m)}\}$ are statistically independent. Aside from the necessity of showing the required independence, the probability density functions of $\{g_0^{(m)}\}$ also have to be derived in order to calculate the probability of correctness. All these can be accomplished by invoking the MVCLT. To prepare $\{g_0^{(m)}\}$ for the application of

the MVCLT, define a vector $\underline{z}$:

$$\underline{z} = (g_0^{(1)}, \ldots, g_0^{(m)}, \ldots, g_0^{(2M)}).$$

In addition, define a set of vectors $\{\underline{x}_i\}$, $i = 1, \ldots, N$:

$$\underline{x}_i = (f_i^{(1)} w_i, \ldots, f_i^{(m)} w_i, \ldots, f_i^{(2M)} w_i).$$

With these definitions we write

$$\underline{z} = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} \underline{x}_i. \qquad (III.1.4.2.)$$

The relations among $\underline{z}$, $\underline{x}^{(i)}$, $g_0^{(m)}$, and $f_i^{(m)} w_i$ are tabulated in Table III.1.4.1.

In the table, the first entry of each column is equal to $\frac{1}{\sqrt{N}}$ times the sum of the remaining entries of the same column,

$$g_0^{(m)} = \frac{1}{\sqrt{N}} \underline{f}^{(m)} \cdot \underline{w} = \frac{1}{\sqrt{N}} (f_1^{(m)} w_1 + \cdots + f_i^{(m)} w_i + \cdots + f_N^{(m)} w_N).$$

The first entry of each row is a row vector of which the components are given by the remaining entries of the same row, e.g., $\underline{x}_i = (f_i^{(1)} w_i, \ldots, f_i^{(m)} w_i, \ldots, f_i^{(2M)} w_i)$. By inspecting Table III.1.4.1., we learn that components from different rows (or $\underline{x}_i$), are statistically independent. (Consider components from different rows and different columns. They are clearly statistically independent. Consider components from different rows but from the same column, e.g., $f_i^{(m)} w_i$ and $f_j^{(m)} w_j$. They are also statistically independent since the subscripts are different.) The only possible dependent terms are confined in the same rows. As pointed out in section III.1.3., statistical dependent terms within the same vector do not affect the statistical independence of $\{\underline{x}_i\}$. In other words, the joint probability density function of all

$\underline{z}_0$	$g_0^{(1)}$	$\cdots$	$g_0^{(m)}$	$\cdots$	$g_0^{(2M)}$
$\underline{x}_1$	$f_1^{(1)} \cdot w_1$	$\cdots$	$f_1^{(m)} \cdot w_1$	$\cdots$	$f_1^{(2M)} \cdot w_1$
$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$\vdots$
$\underline{x}_m$	$f_i^{(1)} \cdot w_i$	$\cdots$	$f_i^{(m)} \cdot w_i$	$\cdots$	$f_i^{(2M)} \cdot w_i$
$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$\vdots$
$\underline{x}_N$	$f_N^{(1)} \cdot W_N$	$\cdots$	$f_N^{(m)} \cdot w_N$	$\cdots$	$f_N^{(2M)} \cdot w_N$

Table III.1.4.1. The first entry of each row stands for a row vector of which the components are given by the remaining entries of the same row, e.g., $\underline{x}_i = (f_i^{(1)} w_i, \ldots, f_i^{(m)} w_i, \ldots, f_i^{(2M)} w_i)$.

the vectors $\{\underline{\mathbf{x}}_i\}$ is equal to the product of the probability density functions of each individual $\underline{\mathbf{x}}_i$:

$$p_{\underline{\mathbf{x}}_1,\cdots,\underline{\mathbf{x}}_i,\cdots,\underline{\mathbf{x}}_N} = p_{\underline{\mathbf{x}}_1}\cdots p_{\underline{\mathbf{x}}_i}\cdots p_{\underline{\mathbf{x}}_N}.$$

Because the probability density functions of $\{f_i^{(m)}\}$ are identical for each i, the probability density functions of $\{w_i\}$ are also identical. Thus the probability density functions of $\{f_i^{(m)} w_i\}$ are also identical. This in turn implies that the probability density functions of the vectors $\{\underline{\mathbf{x}}_i\}$ are also identical:

$$p_{\underline{\mathbf{x}}_i} = \cdots = p_{\underline{\mathbf{x}}_i} = \cdots = p_{\underline{\mathbf{x}}_N} = p_{\underline{\mathbf{x}}}$$

Having all the required conditions statisfied, the MVCLT is invoked to declare that the joint cumulative probability function of $\underline{\mathbf{z}} = (g_0^{(1)}, \ldots, g_0^{(m)}, \ldots, g_0^{(2M)})$, or $P_{\underline{\mathbf{z}}}(\underline{\alpha})$, converges to gaussian form in the limit. If the probability density function of $f_i^{(m)}$ is continuous, the joint probability density function of $\underline{\mathbf{z}}$, $p_{\underline{\mathbf{z}}}(\underline{\beta})$, is also gaussian in the limit, i.e.,

$$p_{\underline{\mathbf{z}}}(\underline{\beta}) = \frac{1}{(2\pi)^M |\underline{\underline{\mathbf{\Lambda}}}_{\underline{\mathbf{z}}}|^{\frac{1}{2}}} exp[-\frac{1}{2}(\underline{\beta} - \underline{\mu}_z)\underline{\underline{\mathbf{\Lambda}}}^{-1}(\underline{\beta} - \underline{\mu}_z)^t],$$

where $\underline{\underline{\mathbf{\Lambda}}}_z$ is the covariance matrix and $\underline{\mu}_z$ is the mean vector of $\underline{\mathbf{z}}$. Regardless of the continuity of $f_i^{(m)}$, the cumulative probability function of $\underline{\mathbf{z}}$, $P_{\underline{\mathbf{z}}}(\underline{\alpha})$ can always be written as

$$P_{\underline{\mathbf{z}}}(\underline{\alpha}) = \int_{-\infty}^{\underline{\alpha}} \frac{1}{(2\pi)^M |\underline{\underline{\mathbf{\Lambda}}}_z|^{\frac{1}{2}}} exp[-\frac{1}{2}(\underline{\beta} - \underline{\mu}_z)\underline{\underline{\mathbf{\Lambda}}}^{-1}(\underline{\beta} - \underline{\mu}_z)^t] d\underline{\beta}, \qquad (III.1.4.3)$$

in the limit. In general, in order to calculate the probability of correctness (or error), an integral of the appropriate joint probability density function over some region will be evaluated. In other words, only the cumulative probability density function

is of interest. Hence, whether the probability density function conforms strictly to gaussian is immaterial in our investigation of the capacity (as was pointed out in the last section).

III.1.5. Evaluation of the Mean Vector and the Covariance Matrix

Let $\underline{\mu}_{x_i}$ be the mean of $\underline{\mathbf{x}}_i$. Recall that $p_{\underline{\mathbf{x}}_i} = p_{\underline{\mathbf{x}}}$. Therefore,

$$\underline{\mu}_{x^i} = \underline{\mu}_x.$$

By the definition of $\underline{\mathbf{z}}$,

$$\begin{aligned} \underline{\mu} &= E[\frac{1}{\sqrt{N}} \sum_{i=1}^{N} \underline{\mathbf{x}}_i] \\ &= \frac{1}{\sqrt{N}} \sum_{i=1}^{N} E[\underline{\mathbf{x}}_i] \\ &= \sqrt{N}\underline{\mu}_x. \end{aligned} \qquad (III.1.5.1)$$

The $m\text{-}th$ component of $\underline{\mu}_x$, μ_m is given as

$$\mu_m = E[f_i^{(m)} w_i]. \qquad (III.1.5.2.)$$

Due to the fact that the probability density functions of $\{f_i^{(m)} w_i\}, m = 1, \ldots, M$, are identical, the first M components in $\underline{\mu}_x$ must also be identical. Similarly, the last M components in $\underline{\mu}_x$ are also identical. Thus, we need only consider two cases:

i. $m = 1, \ldots, M$.

$$\mu_m = E[f_i^{(m)} \sum_{l=1}^{N} f_i^{(l)}]$$

$$= E[(f_i^{(m)})^2 + \sum_{l \neq m} f_i^{(m)} f_i^{(l)}]$$
$$= (\sigma^2 + \sum_{l \neq m} E[f_i^{(m)}] E[f_i^{(l)}])$$
$$= \sigma^2. \qquad (III.1.5.3.)$$

ii. $m = M+1, \ldots, 2M.$

$$\mu_m = \sqrt{N+1} E[f_i^{(m)} \sum_{l=1}^{N} f_i^{(l)}]$$
$$= \sqrt{N+1} \sum_{l=1}^{N} E[f_i^{(m)}] E[f_i^{(l)}]$$
$$= 0. \qquad (III.1.5.4.)$$

It follows that the first M elements of $\underline{\mu}_x$ are identical and equal to σ^2 while the last M elements are equal to 0:

$$\underline{\mu}_x = (\sigma^2, \ldots, \sigma^2, 0, \ldots, 0). \qquad (III.1.5.5.)$$

From eq.(III.1.5.1.) and eq.(III.1.5.5.), we have

$$\underline{\mu}_z = (\sqrt{N}\sigma^2, \ldots, \sqrt{N}\sigma^2, 0, \ldots, 0). \qquad (III.1.5.6.)$$

Let $\underline{\underline{\Lambda}}_{\underline{x}}$ be the covariance matrix of $p_{\underline{x}}$. The normalization of $\underline{z}$ by the factor $\frac{1}{\sqrt{N}}$ in eq.(III.1.4.2) is such that

$$\underline{\underline{\Lambda}}_{\underline{z}} = E[(\underline{z} - \underline{\mu}_z)^{\mathrm{T}} (\underline{z} - \underline{\mu}_z)]$$
$$= \frac{1}{N} E\Big[(\sum_i (\underline{x}_i - \underline{\mu}_x))^T (\sum_j (\underline{x}_j - \underline{\mu}_x)) \Big]$$

$$= \frac{1}{N} E\Big[\sum_i (\underline{\mathbf{x}}_i - \underline{\mu}_x)^T (\underline{\mathbf{x}}_i - \underline{\mu}_x) + \sum_i \sum_{j \neq i} (\underline{\mathbf{x}}_i - \underline{\mu}_x)^T (\underline{\mathbf{x}}_j - \underline{\mu}_x) \Big]$$

$$= \frac{1}{N} \sum_i E[(\underline{\mathbf{x}}_i - \underline{\mu}_x)^T (\underline{\mathbf{x}}_i - \underline{\mu}_x)]$$

$$= \underline{\underline{\mathbf{\Lambda}}}_{\mathbf{x}}, \qquad (III.1.5.7.)$$

because

$$E[(\underline{\mathbf{x}}^{(i)} - \underline{\mu}_x)^T (\underline{\mathbf{x}}^{(j)} - \underline{\mu}_x)] = E[(\underline{\mathbf{x}}^{(i)} - \underline{\mu}_x)^T] E[(\underline{\mathbf{x}}^{(j)} - \underline{\mu}_x)] = \underline{\underline{0}}; \qquad j \neq i.$$

$\underline{\underline{\mathbf{\Lambda}}}_x$ can also be expressed as

$$\underline{\underline{\mathbf{\Lambda}}}_x = E[\underline{\mathbf{x}}^T \underline{\mathbf{x}}] - \underline{\mu}_{\mathbf{x}}^{\mathbf{t}} \underline{\mu}_{\mathbf{x}}, \qquad (III.1.5.8.)$$

where the superscript i in $\underline{\mathbf{x}}_i$ is dropped because its statistics are invariant with respect to i. The second term on the LHS in eq.(III.1.5.8.) can be obtained by simply taking the outer-product of $\underline{\mu}_x$ given by eq.(III.1.5.6.). Due to the configuration of $\underline{\mu}_x$, all the elements except those with both their row count and column count less than or equal to M in $\underline{\mu}_x^T \underline{\mu}_x$ are zero:

$$\underline{\mu}_x^t \underline{\mu}_x = \begin{pmatrix} \sigma^4 & \dots & \sigma^4 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \sigma^4 & \dots & \sigma^4 & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & \dots & 0 \end{pmatrix} \qquad (III.1.5.9.)$$

The expected value of the outer-product of $\underline{\mathbf{x}}$ can be evaluated by taking care of its diagonal and off-diagaonal elements separately. Let ϱ_{m_1,m_2} be the element at the m_1-th row and the m_2 column of $E[\underline{\mathbf{x}}^T \underline{\mathbf{x}}]$.

$$\varrho_{m_1,m_2} = E\Big[(f_i^{(m_1)} w_i) (f_i^{(m_2)} w_i) \Big]. \qquad (III.1.5.10.)$$

For convenience, all the i subscripts in the above equation will be dropped in the process of calculating ϱ_{m_1,m_2}. It is legitimate to do so because the statistics of all $f_i^{(m)}$ are identical. For the off-diagonal (row not equal to column) elements, calculation can further be divided into two cases:

i. elements with either their row count or column count (or both) greater than M, i.e., $m_1 > M$ or $m_2 > M$ and $m_1 \neq m_2$. Without loss of generality, assume $m_1 > M$. Since w is composed only of $f^{(m)}$ with $m \leq M$,

$$\begin{aligned} \varrho_{m_1,m_2} &= E\Big[f^{(m_1)}\Big] E\Big[w^2 f^{(m_2)}\Big] \\ &= 0. \end{aligned} \qquad (III.1.5.11.)$$

ii. elements with both their row count and column count no greater than M, i.e. $m_1 \leq M$ and $m_2 \leq M$ and $m_1 \neq m_2$.

$$\begin{aligned} \varrho_{m_1,m_2} &= E\Big[(f^{(m_1)} \sum_{l_1} f^{(l_1)})(f^{(m_2)} \sum_{l_2} f^{(l_2)})\Big] \\ &= E\Big[\Big((f^{(m_1)})^2 + \sum_{l_1 \neq m_1} f^{(m_1)} f^{(l_1)}\Big)\Big((f^{(m_2)})^2 + \sum_{l_2 \neq m_2} f^{(m_2)} f^{(l_2)}\Big)\Big] \\ &= E\Big[(f^{(m_1)})^2\Big] E\Big[(f^{(m_2)})^2\Big] + \sum_{l_2} E\Big[f^{(l_2)}\Big] E\Big[(f^{(m_1)})^2\Big] E\Big[f^{(m_2)}\Big] \\ &\quad + \sum_{l_1} E\Big[f^{(l_1)}\Big] E\Big[(f^{(m_2)})^2\Big] E\Big[f^{(m_1)}\Big] \\ &\quad + \sum_{l_1 \neq m_1} \sum_{l_2 \neq m_2} E\Big[f^{(l_1)} f^{(l_2)}\Big] E\Big[f^{(m_1)}\Big] E\Big[f^{(m_2)}\Big] \\ &= \sigma^4. \end{aligned} \qquad (III.1.5.12.)$$

As for the calculation of $\varrho_{m,m}$, it can also be divided into two cases:

i. $m > M$.

$$\begin{aligned}\varrho_{m,m} &= E\Big[(f^{(m)}\sum_{l_1} f^{(l_1)})(f^{(m)}\sum_{l_2} f^{(l_2)})\Big] \\ &= E\Big[(f^{(m)})^2\Big]E\Big[\sum_{l_1}\sum_{l_2} f^{(l_1)}f^{(l_2)}\Big] \\ &= \sigma^2\Big(\sum_{l_1} E\Big[(f^{(l_1)})^2\Big] + \sum_{l_1}\sum_{l_2\neq l_1} E\Big[f^{(l_1)}\Big]E\Big[f^{(l_2)}\Big]\Big) \\ &= M\sigma^4. \qquad (III.1.5.13.)\end{aligned}$$

ii. $m \leq M$.

$$\begin{aligned}\varrho_{m,m} &= E\Big[(f^{(m)}\sum_{l_1} f^{(l_1)})(f^{(m)}\sum_{l_2} f^{(l_2)})\Big] \\ &= E\Big[(f^{(m)})^4\Big] + \sum_{l_1\neq m} E\Big[(f^{(m)})^3\Big]E\Big[f^{(l_1)}\Big] + \sum_{l_2\neq m} E\Big[(f^{(m)})^3\Big]E\Big[f^{(l_2)}\Big] \\ &\quad + \sum_{l_1\neq m}\sum_{l_2\neq m} E\Big[(f^{(m)})^2\Big]E\Big[f^{(l_1)}f^{(l_2)}\Big] \\ &= E\Big[(f^{(m)})^4\Big] + \sum_{l_1\neq m} E\Big[(f^{(m)})^2\Big]E\Big[(f^{(l_1)})^2\Big] \\ &\quad + \sum_{l_1\neq m}\sum_{\substack{l_2\neq m\\ l_2\neq l_1}} E\Big[(f^{(m)})^2\Big]E\Big[f^{(l_1)}\Big]E\Big[f^{(l_2)}\Big] \\ &= E\Big[(f^{(m)})^4\Big] + (M-1)\sigma^4. \qquad (III.1.5.14.)\end{aligned}$$

Let ϑ_{m_1,m_2} be the element at the m_1-th row and m_2-th column of $\underline{\underline{\Lambda}}_{\mathbf{x}}$. From eq.(III.1.5.9), eq.(III.1.5.11.) and eq.(III.1.5.12.), it is clear that all the off-diagonal ϑ_{m_1,m_2} s are equal to 0. As for $m_1 = m_2 = m$, refering to eq. (III.1.4.9.), eq. (III.1.4.13.) and eq. (III.1.4.14.),

$$\vartheta_{m,m} = \begin{cases} E\Big[(f^{(m)})^4\Big] + (M-2)\sigma^4 & \text{if } m = 1,\ldots,M; \\ M\sigma^4 & \text{if } m = M+1,\ldots,2M. \end{cases} \qquad (III.1.5.15.)$$

Assuming $E\left[(f^{(m)})^4\right]$ finite, the above equation becomes

$$\vartheta_{m,m} \approx M\sigma^4 \qquad m = 1, \ldots, M, \qquad (III.1.5.16.)$$

for large enough M. The covariance matrix $\underline{\underline{\Lambda}}_z$ therefore is given as

$$\underline{\underline{\Lambda}}_z = \begin{pmatrix} M\sigma^4 & 0 & \ldots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & M\sigma^4. \end{pmatrix} \qquad (III.1.5.17.)$$

III.1.6. Evaluation of the Capacity

Having shown that $\underline{\underline{\Lambda}}_z$ diagonal, we have actually shown that $\{g_0^{(m)}\}$ are statistically independent in the limit that N tends to infinity. Specifically, if $P_{g_0^{(m)}}(\alpha_m)$ stands for the cumulative probability function of the random variable $g_0^{(m)}$, then

$$P_{\underline{z}}(\underline{\alpha}) = \prod_m P_{g_0^{(m)}}(\alpha_m),$$

and

$$P_{g_0^{(m)}}(\alpha_m) = \begin{cases} \int_{-\infty}^{\alpha_m} \frac{1}{\sqrt{2\pi M\sigma^2}} exp[\frac{-(\beta_m - N\sigma^2)^2}{2M\sigma^4}]\, d\beta_m & \text{if } m = 1, \ldots, M; \\ \\ \int_{-\alpha_m}^{\infty} \frac{1}{\sqrt{2\pi M\sigma^2}} exp[\frac{-\beta_m^2}{2M\sigma^4}]\, d\beta_m & \text{if } m = M+1, \ldots, 2M. \end{cases}$$

$$(III.1.5.1)$$

Refering to eq.(III.1.2.2.) and eq.(III.1.6.1.),

$$P[E_1^{(m)}] = P[g_0^{(m)} > \tau]$$

$$= 1 - P_{g_0^{(m)}}(\tau)$$

$$= 1 - \int_{-\infty}^{\tau} \frac{1}{\sqrt{2\pi M\sigma^2}} exp[\frac{-(\beta_m - N\sigma^2)^2}{2M\sigma^4}]\, d\beta_m, \qquad (III.1.6.2)$$

$$m = 1, \ldots, M.$$

And

$$P[E_2^{(m)}] = P[g_0^{(m)} < \tau]$$

$$= P_{g_0^{(m)}}(\tau)$$

$$= \int_{-\infty}^{\alpha_m} \frac{1}{\sqrt{2\pi M\sigma^2}} exp[\frac{-\beta_m^2}{2M\sigma^4}]\, d\beta_m, \qquad (III.1.6.3)$$

$$m = M+1, \ldots, 2M.$$

It is apparent that

$$P[E_1^{(1)}] = \cdots = P[E_1^{(M)}] = P_1 \qquad (III.1.6.4a)$$

and

$$P[E_2^{(M+1)}] = \cdots = P[E_2^{(2M)}] = P_2, \qquad (III.1.6.4b)$$

where

$$P_1 = 1 - \int_{-\infty}^{\tau} \frac{1}{\sqrt{2\pi M\sigma^2}} exp[\frac{-(\beta - N\sigma^2)^2}{2M\sigma^4}]\, d\beta \qquad (III.1.6.5a)$$

and

$$P_2 = \int_{-\infty}^{\tau} \frac{1}{\sqrt{2\pi M\sigma^2}} exp[\frac{-\beta^2}{2M\sigma^4}]\, d\beta. \qquad (III.1.6.5b)$$

Substituting eq.(III.1.6.5.) and eq.(III.1.6.6.) into eq.(III.1.2.2.), we have

$$P_c = (P_1 P_2)^M. \qquad (III.1.6.6.)$$

Let p_1 and p_2 represent the integrands of P_1 and P_2 (eq.(III.1.6.5.)), respectively. Note that p_1 and p_2 have exactly the same shape, except p_1 is shifted to the right by $N\sigma^2$ (Fig.III.6.1.).

Geometrically, P_1 is equal to the area under p_1 to the left of τ whereas P_2 is equal to the area under p_2 to the right of τ. It can be shown easily that P_1P_2 is maximum when τ equals one-half of the difference of the means of p_1 and p_2. We therefore choose $\tau = \frac{1}{2}N\sigma^2$ to maximize P_c. In this instance,

$$P_1 = P_2 = 1 - \int_{-\infty}^{\tau} \frac{1}{\sqrt{2\pi M\sigma^2}} exp[\frac{-(\beta - N\sigma^2)^2}{2M\sigma^4}]\, d\beta. \qquad (III.1.6.7.)$$

And eq.(III.1.6.6.) can be written as

$$P_c = \left(1 - Q(\sqrt{\frac{N}{2M}})\right)^{2M}, \qquad (III.1.6.8.)$$

where

$$Q(\sqrt{\frac{N}{2M}}) = \int_{\sqrt{\frac{N}{2M}}}^{\infty} \frac{1}{\sqrt{2\pi}\sigma}\, exp\left(\frac{-\alpha^2}{2\sigma^2}\right)\, d\alpha$$

which is also known as the cumulative gaussian function.

Recall that the goal of this section is to obtain a relation between M and N under the condition that P_c be arbitrarily close to 1 as $N \to \infty$. Assume P_c to be very close to 1. Also assume that $N/M \to \infty$ as $N \to \infty$. (These assumptions, of course, have to be justified later.) Taking logarithms on both sides of eq.(III.1.6.8.), we have

$$ln(P_c) = 2M ln(1 - Q(\sqrt{\frac{N}{2M}})). \qquad (III.1.6.9.)$$

For P_c to be very close to 1, $Q(\sqrt{\frac{N}{2M}})$ has to be very small. We may thus invoke the approximation

$$ln(1-u) \approx -u \qquad \text{for small } u.$$

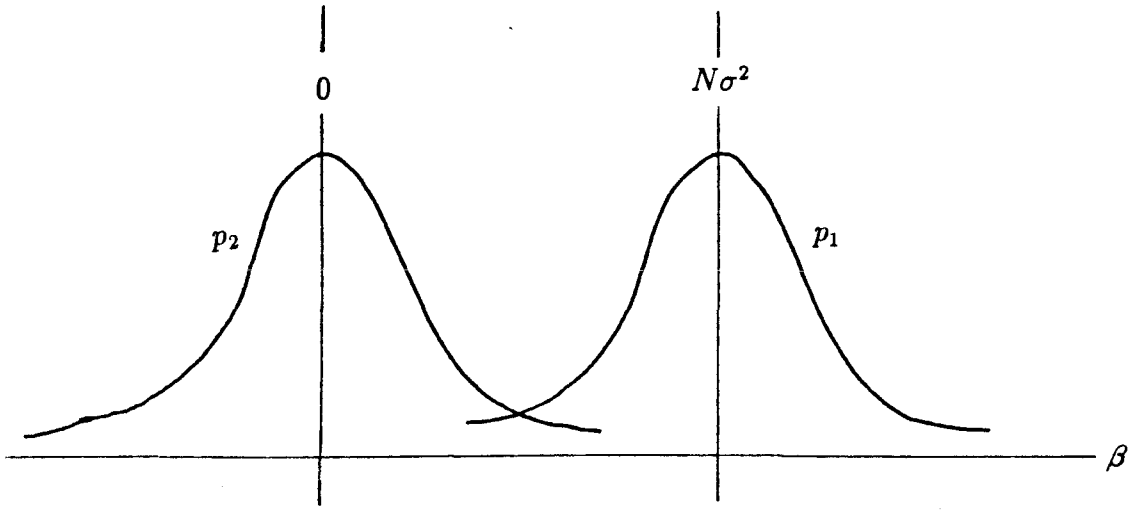


Fig. III.6.1. P_1 is equal to the area under p_1 from τ to ∞ whereas P_2 is equal to the area under p_2 from $-\infty$ to τ.

Therefore, eq.(III.1.6.9.) becomes

$$ln(P_c) \approx -2MQ(\sqrt{\frac{N}{2M}}).$$

Using the approximation

$$Q(v) \approx \frac{1}{\sqrt{2\pi}} \frac{exp(-v^2/2)}{v} \qquad \text{for large } u,$$

eq.(III.1.6.9.) is reduced to

$$ln(P_c) \approx -2M \frac{1}{\sqrt{2\pi}} \frac{exp(-N/4M)}{\sqrt{N/2M}}.$$

Taking logarithms on both sides of the above equation again, we have

$$ln(c) \approx ln(M\sqrt{\frac{M}{N}}) - \frac{N}{4M}, \qquad (III.1.6.10.)$$

where

$$c = -\frac{\sqrt{\pi}}{2} ln(P_c).$$

Assuming further that $|ln(c)| \ll ln(M\sqrt{\frac{M}{N}})$, a transcendental equation relating M and N is obtained:

$$M \approx \frac{N}{8ln(M\sqrt{\frac{M}{N}})}. \qquad (III.1.6.11.)$$

Since $M \leq N$, a lower bound for M can be obtained by substituting N for M on the RHS of eq.(III.1.6.11.):

$$M \gtrsim \frac{N}{8lnN}.$$

Substituting this lower bound of M for M on the RHS of eq.(III.1.6.11.), an upper bound for M is obtained:

$$M \lesssim \frac{N}{8lnN - 8ln(8lnN)}. \qquad (III.1.6.12.)$$

We can of course substitute this upper bound into eq.(III.1.6.11.) to obtain a tighter lower bound and continue in this fashion to derive a series solution. But doing so does not provide a return comparable to the effort involved. We shall hence invoke the large N approximation once again. For large enough N, the second term, $8ln(8lnN)$, in the denominator on the RHS of eq.(III.1.6.12.) becomes negligible. In other words, the upper bound for M approaches the lower bound as $N \rightarrow \infty$. Consequently, we have an approximate expression relating M and N:

$$M \approx \frac{N}{8lnN} \qquad \text{as } N \rightarrow \infty. \qquad (III.1.6.13.)$$

Now that we have an expression for M, we have to use it to justify our previous assumptions. The assumption that $N/M \rightarrow \infty$ as $N \rightarrow \infty$ is clearly valid. The assumption that $P_c \rightarrow 1$ as $N \rightarrow \infty$ can be verified easily by substituting expression (III.1.6.13.) into eq.(III.1.6.8.) and letting N go to infinity. As for the relative sizes of c and $ln(M\sqrt{M/N})$, it can be shown that c is of the order $ln(lnN)$ which is negligible for large enough N. With all the assumptions justified, we can only claim that a legitimate approximate expression relating N and M has been found. An immediate question arises: how good is this approximation? The answer to this question can be found by considering an alternate expression for M:

$$M_a = \frac{(1+\epsilon)N}{8lnN}$$

where ϵ is some arbitrarily small constant. Apparently,

$$M_z > \frac{N}{8lnN}.$$

Substituting M_a in eq.(III.1.6.8.), it can be shown that $P_c \rightarrow 0$ as $N \rightarrow \infty$. This means that the expression obtained earlier is indeed a very good approximation.

That is to say

$$M_{c1} = \frac{N}{8lnN}, \qquad N \to \infty \qquad (III.1.6.14.)$$

is the capacity of the MEC with only one (center) output pixel.

III.2. Image Inner-product Filter

III.2.0. Basic Model

Suppose the medium used to record the MEH has a 1-bit (2 gray levels) dynamic range. The true hologram recorded $\underline{w} = (w_1, \ldots, w_i, \ldots, w_N)$ is then given by

$$w_i = \beta\Big[\sum_{l=1}^{N} f_i^{(l)}\Big], \qquad (III.2.0.1a)$$

where

$$\beta[x] = \begin{cases} 1 & \text{if } x \geq 0; \\ -1 & \text{if } x < 0. \end{cases} \qquad (III.2.0.1b)$$

We refer to this hologram as the binary multiple exposure hologram (BMEH).

Undoubtedly, the capacity of the BMEH is smaller than that of the MEH due to the information lost in the process of recording the MEH on a 1-bit dynamic range medium. However, we learned from Chapter II that thresholding a matched filter will only reduce the processing gain (peak to side-lobe ratio) by a constant factor. Now, the MEH is only a summation of matched filters. We have shown in section III.1 that the probability of error, and hence, the capacity, is a function of the processing gain. We might therefore expect that the capacity is reduced only

by a constant factor when the MEH is thresholded. This is shown indeed to be the case.

III.2.1 Similarities and Variations in Assumptions and Formulation

The definitions of the two-category classification problem and the statistics of the input vectors are exactly the same as those described in the section III.1.2. The method that the two classes, Ω_1 and Ω_2, of vectors are combined to form Φ remains unchanged. The operation performed by the binary classifier is still an inner-product:

$$g_0^{(m)} = \frac{1}{\sqrt{N}} \underline{\mathbf{w}} \cdot \underline{\mathbf{f}}^{(m)}, \qquad m = 1, \ldots, 2M.$$

The only variation is the algorithm under which $\underline{\mathbf{w}}$ is generated. $\underline{\mathbf{w}}$ in this case is the BMEH given in eq.(III.2.0.1). The formulation of the probability of correctness of a dot-product filter described in section III.1.3. was derived with no regard of the exact form of the filter and hence is also applicable here. Thus the equation expressing the probability of correctness (eq.III.1.2.2.) is also valid here.

To derive the capacity of the BMEC, we have to derive the solution for the probability of correctness P_c again. To derive P_c, we have to consider the statistics of $\{g_0^{(m)}\}$. The derivation of the jointly gaussian nature of $\{g_0^{(m)}\}$ in section III.1.3 did not require severe constraint on the exact form of $\underline{\mathbf{w}}$ (Table III.1.4.1.). The only property that was used was the fact that the *i-th* component of $\underline{\mathbf{w}}$ must be generated from only the *i-th* components of $\{\underline{\mathbf{f}}^{(m)}\}$. This requirement is certainly met by the binary weight vector discussed here. Due to the satisfaction of the above

requirement, we may conclude that the joint cumulative probability function of the vector $\underline{\mathbf{z}}$ given as $\underline{\mathbf{z}} = (g_0^{(1)}, \ldots, g_0^{(m)}, \ldots, g_0^{(2M)})$:

$$\underline{\mathbf{z}} = \frac{\mathbf{1}}{\sqrt{\mathbf{N}}} \sum_{\mathbf{i}}^{\mathbf{N}} \underline{\mathbf{x}}_{\mathbf{i}} \qquad (III.2.1.1a.)$$

$$\underline{\mathbf{x}}_i = (w_i f_i^{(1)}, \ldots, w_i f_i^{(m)}, \ldots, w_i f_i^{(2M)}). \qquad (III.2.1.1b.)$$

is also gaussian.

The exact form of the joint cumulative probability function of $\underline{\mathbf{z}}$ is completely defined once we compute its mean vector and covariance matrix. In other words, under the condition that the *i-th* component of $\underline{\mathbf{w}}$ is only generated from the *i-th* component of $\{\underline{\mathbf{f}}^{(m)}\}$, the mean vector and covariance matrix of $\underline{\mathbf{z}}$ are the only two things that are affected by the exact algorithm through which $\underline{\mathbf{w}}$ is created.

III.2.2 Evaluation of the Mean Vector and the Covariance Matrix

We will first evaluate the mean of $\underline{\mathbf{z}}$. Taking the expected value of equation (III.2.1.1a.), we have

$$\underline{\mu}_z = \sqrt{N}\underline{\mu}_x. \qquad (III.2.2.1.)$$

Let the *m-th* component of $\underline{\mu}_x$ be μ_m:

$$\mu_m = E[f_i^{(m)} w_i]. \qquad (III.2.2.2.)$$

For $m = M+1, \ldots, 2M$.

$$\mu_m = E[f_i^{(m)}]\,[w_i] = 0 \qquad (III.2.2.3.)$$

because $f_i^{(m)}$ is independent of w_i.

For $m = 1, \ldots, M$ we have to determine the probability density function of $w_i f_i^{(m)}$, say $p(wf^{(m)})$ (the subscript i is dropped for convenience) in order to evaluate $E[f_i^{(m)} w_i]$. By Bayes rule,

$$p(wf^{(m)} = c) = P(w = 1|f^{(m)} = c)p(f^{(m)} = c) + P(w = -1|f^{(m)} = -c)p(f^{(m)} = -c). \quad (III.2.2.4.)$$

(p stands for the probability density function whereas P stands for the probability.) $w = 1$ if and only if $\sum_{l}^{M} f^{(l)} \geq 0$, hence

$$P(w = 1|f^{(m)} = c) = P(f^{(m)} \geq h|f^{(m)} = c), \quad (III.2.2.5.)$$

where

$$h = -\sum_{l \neq m}^{M} f^{(l)}.$$

By the central limit theorem, the cumulative probability function of h,

$$F_h(c) = \int_{-\infty}^{c} G(\alpha; 0, (M-1)\sigma^2) d\alpha \quad \text{for large M}, \quad (III.2.2.6.)$$

where $G(\alpha; 0, (M-1)\sigma^2)$ is the gaussian distribution of the dummy variable α with mean zero and variance $(M-1)\sigma^2$. The event "$f^{(m)}$ be no less than h under the condition that $f^{(m)} = c$" is equivalent "h be less than c." Therefore $P(f^{(m)} \geq h|f^{(m)} = c)$ is simply equal to the cumulative probability function of h evaluated at c. From eq.(III.2.2.5.),

$$P(w = 1|f^{(m)} = c) = F_h(c). \quad (III.2.2.7a.)$$

Similarly, it is found that

$$P(w = -1|f^{(m)} = -c) = F_h(c). \qquad (III.2.2.7b.)$$

Suppose $p_f(c)$ stands for the probability density function of the random variable $f^{(m)}$. Substituting $p_f(c)$ and eq.(III.2.2.7.) into eq.(III.2.2.4.),

$$p(wf^{(m)} = c) = 2p_f(c)F_h(c). \qquad (III.2.2.8.)$$

μ_m is obtained by integrating over $p(wf^{(m)} = c)$ from $-\infty$ to ∞:

$$\mu_m = 2\int_{-\infty}^{\infty} p_f(c)F_h(c)\,dc, \qquad (III.2.2.9.)$$

where $p_f(c)$ is the probability density function of $f^{(m)}$. To evaluate the above integral, first, we differentiate it with respect to M:

$$\frac{\partial \mu_g}{\partial M} = \frac{-1}{\sqrt{2\pi}\sigma(M-1)^{\frac{3}{2}}}\int_{-\infty}^{\infty} c^2 p_f(c) exp(\frac{-c^2}{2(M-1)\sigma^2})\,dc.$$

Since $var[p_f(c)] = \sigma^2 \ll (M-1)\sigma^2$ for finite σ^2 and large enough M, the exponential part of the above integrand practically equals to 1 the entire domain of $p_f(c)$. Consequently,

$$\begin{aligned}\frac{\partial \mu_g}{\partial M} &\approx \frac{-1}{\sqrt{2\pi}\sigma(M-1)^{\frac{3}{2}}}\int_{-\infty}^{\infty} c^2 p_f(c)\,dc \\ &\approx \frac{-\sigma}{\sqrt{2\pi}M^{\frac{3}{2}}}. \qquad (III.2.2.10.)\end{aligned}$$

Integrating the above expression with respect to M, we obtain

$$\mu_m = \sqrt{\frac{2}{M\pi}}\sigma + \kappa. \qquad (III.2.2.11.)$$

As M approaches infinity, the first term on the R.H.S of the above equation approaches zero. Therefore, $\mu_m = \kappa$. If κ is non-zero, it means that no matter how large the number of vectors stored, w will always have some preferred sign which is clearly nonsense. In other words, κ cannot assume any other value but 0. Thus,

$$\mu_m = \sqrt{\frac{2}{M\pi}}\sigma. \qquad (III.2.2.12.)$$

Substituting eq.(III.2.2.12.) and eq.(III.2.2.3) into eq.(III.2.2.1), we obtain

$$\mu_z = (\sqrt{\frac{2N}{M\pi}}\sigma, \ldots, \sqrt{\frac{2N}{M\pi}}\sigma, 0 \ldots, 0).$$

As for the covariances, consider

$$E[g_0^{(m_1)} g_0^{(m_2)}] = \frac{1}{N} E[\sum_i^N w_i f_i^{(m_1)} \sum_j^N w_j f_j^{(m_2)}]$$
$$= \frac{1}{N} E[\sum_i^N f_i^{(m_1)} f_i^{(m_2)}] + \frac{1}{N} \sum_i^N \sum_{j \neq i}^N E[w_i f_i^{(m_1)}] E[w_j f_j^{(m_2)}]$$

$$(III.2.2.13.)$$

Therefore,

$$var[g_0^{(m_1)} g_0^{(m_2)}] = \frac{1}{N} E[\sum_i^N f_i^{(m_1)} f_i^{(m_2)}].$$

For $m_1 \neq m_2$.

$$var[g_0^{(m_1)} g_0^{(m_2)}] = \frac{1}{N} \sum_i^N E[f_i^{(m_1)}] E[f_i^{(m_2)}] = 0. \qquad (III.2.2.14.)$$

For $m_1 = m_2$,

$$var[(g_0^{(m_1)})^2] = \frac{1}{N} \sum_i^N E[(f_i^{(m_1)})^2] = \sigma^2 \qquad (III.2.2.15.)$$

III.2.3. Evaluation of Capacity

We have shown that $g_0^{(m_1)}$ and $g_0^{(m_2)}$, $m_1 \neq m_2$, are uncorrelated. Since $\{g_0^{(m)}\}$ are jointly gaussian, cross-correlations of $\{g_0^{(m)}\}$ equal to zero means that $\{g_0^{(m)}\}$ are statistically independent. As we have noted earlier, the formulation of the probability of correctness of this section is the same as that in section III.1.3. Thus, showing $\{g_0^{(m)}\}$ to be statistically independent implies that P_c here has the same form as given by eq.(III.1.6.8.):

$$P_c = (1 - Q(.))^{2M}$$

The only difference between the above equation and eq.(III.1.6.8.) is the argument of the cumulative gaussian function $Q(.)$ which is one half of the processing gain of the respective correlator outputs. For the BMEC, the processing gain is $\sqrt{\frac{2N}{\pi M}}$. Therefore,

$$P_c = (1 - Q(\frac{1}{2}\sqrt{\frac{2N}{\pi M}}))^{2M} \qquad (III.2.3.1.)$$

Assuming that P_c is arbitrarily small for some large enough N, the procedure from eq.(III.1.6.9.) through eq.(III.1.6.11.) is repeated. The following transcendental equation is arrived at:

$$M \approx \frac{N}{4\pi \ln(M\sqrt{\frac{M}{N}})} \qquad (III.2.3.2.)$$

Using the upper and lower bound argument presented in eq.(III.1.6.13.), the capacity of the BMEC with one output pixel is found to be

$$M_{c2} = \frac{N}{4\pi \ln N}. \qquad (III.2.3.3.)$$

To verify the theoretical result of M_{c2}, 100 computer trials were averaged to determine the capacity for various N. For each trial, two random vectors were

generated and summed. The sum was then thresholded to form the initial reference filter. Each pattern was tested to determine whether classification was performed correctly. If no error occurred, a new random vector was generated and added to the previous sum. The sum is again thresholded to form a new reference filter. Correlations with all patterns were done. The number of patterns in the reference filter was increased until a misclassification occured. At this point, the capacity was said to be one less than the number of patterns stored in the reference.

The result is shown in Fig.III.2.3.1. The capacity is plotted as a function of the number of pixels, N, in the pattern. The small circles represent the experimental means of the results of the 100 trials. The vertical bars are the experimental standard deviation of the results. The solid curve in the figure represents the theoretical result. Experimental simulations show good agreement with theoretical prediction. Identical computer simulation for the fully analog MEC was also done for comparison. Its results were also shown in Fig.III.2.3.1. Like M_{c2}, the experimental simulations for M_{c1} agree well with the theoretical prediction. It is important to note that because the simulations were performed in the regime of small N, the transcendental equations (eq.(III.1.6.11.) and eq.(III.2.3.1.)) were used to plot the theoretical curves.

Comparing eq.(III.2.3.3.) and eq.(III.1.6.14.), we see that the loss in storage capacity of the MEC of one output pixel due to thresholding the filter is just a modest factor of $\frac{2}{\pi}$. Recall from Chapter II that the ratio of the correlation peak to the variance of the side-lobes of the binary matched filter is also equal to $\frac{2}{\pi}$. We noted earlier that the cause of errors in the MEC is the build-up of the cross-

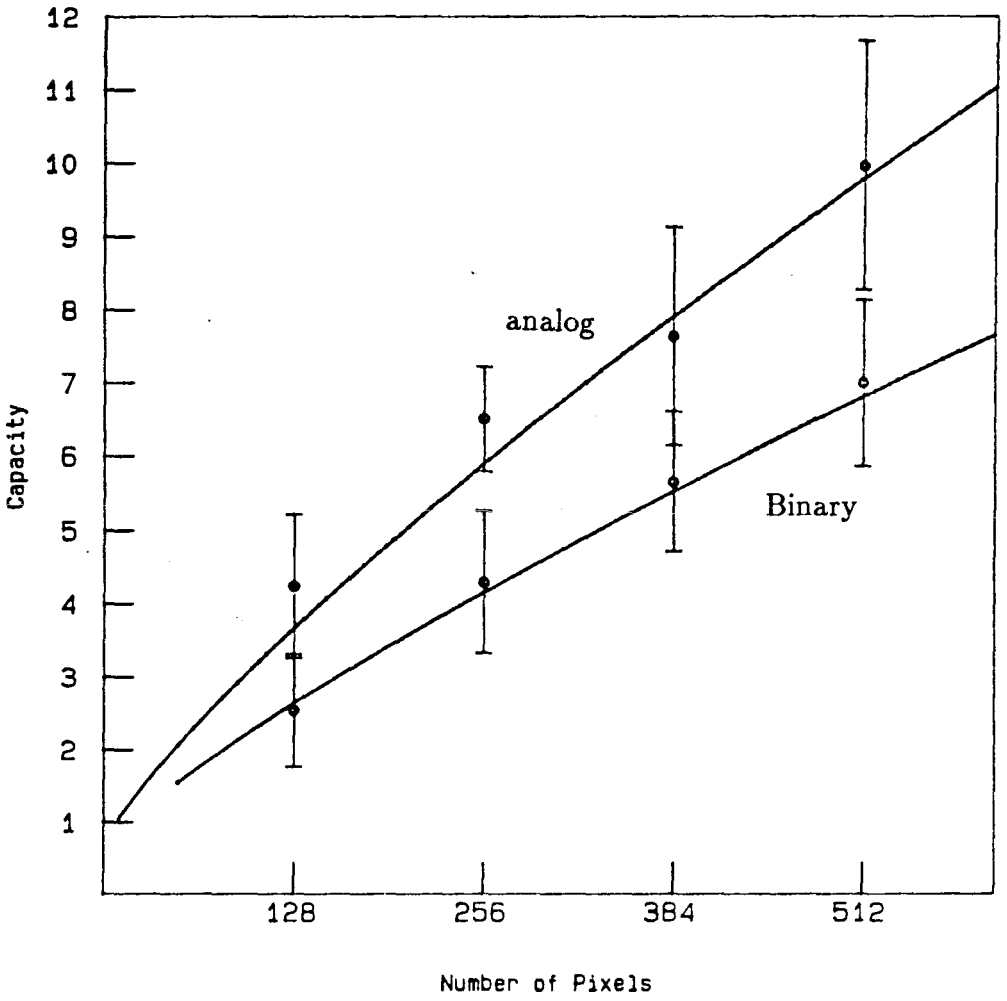


Fig.III.2.3.1. Capacity curves of analog and binary MEC.

talk as more patterns are stored in the reference. The peak-to-variance ratio is exactly the measure of the cross-talk level. Due to the fact that the input patterns are assumed to be statistically independent, the statistics of the output variables become gaussian. This means the chance for an error to occur is completely specified by the peak-to-variance ratio. The appearance of the factor $\frac{2}{\pi}$ in the ratio of the two capacities and the processing gain is therefore no accident. Note also that when the probability of correctness was derived to take on a gaussian form, the exact algorithm under which the filter was created was never used. The only requirement is that any component of the filter is generated only from the same components of the patterns stored. This infers that the probability of correctness of any correlator generated fulfilling the above requirement should take on the form of a gaussian distribution. Should the processing gain of a filter be of the order of $\frac{N}{M}$, the order of the capacity of that filter will also be of $\frac{N}{\ln N}$. The MVCLT also holds for vectors of different joint density functions, so long as the vectors are statistically independent.

III.3. Correlation Filter

A 2-category classifier can be made insensitive to any shift in the input plane by implementing it with a correlation filter rather than a inner-product filter. In such case, we will have an output vector instead of one output point. Mathematically, each component, g_k, of the output vector $\underline{g}$ can be expressed as

$$g_k = \sum_{k}^{N} w_i f_{i+k}\,, \qquad (III.3.0.1.)$$

where w_i is the *i-th* component of the correlation filter $\underline{w}$. In this section, we shall investigate the capacity of the correlation generated by the multiple exposure

algorithm, i.e.,

$$\underline{w} = \sum_m \underline{f}^{(m)}. \qquad (III.3.0.2.)$$

When a vector $\underline{f}$ which was used to form $\underline{w}$ is applied to the correlator, we expect to see a strong correlation peak whose location corresponds to the origin of $\underline{f}$. Whereas when a vector that was not used to form $\underline{w}$ is applied, only weak cross-correlation is expected. However, for a given space bandwidth product N, the cross-correlation and the side-lobes are expected to increase as the number of vectors stored increases. A digital simulation showing this effect is presented in Fig.III.3.1.

In the simulation, one-dimensional vectors composed of 64 statistically independent gaussian random variables, are used. Fig.III.3.1a. shows the autocorrelation of a vector, say $\underline{f}^{(1)}$. The peak at the center of the figure signifies that $\underline{f}^{(1)}$ is recognized. Fig.III.3.1b. shows the cross-correlation of $\underline{f}^{(1)}$ and the reference filter formed by summing $\underline{f}^{(1)}$ and 3 other vectors. Observe that the peak decreases slightly, whereas the side-lobes have increased. Since the peak is still above and the side-lobes are below the threshold level, which is chosen to be 0.5, $\underline{f}^{(1)}$ is still said to be recognized. But in Fig.III.3.1c., as the filter is overloaded, two peaks are detected to be above the threshold level. Because there is no telling whether the output signifies a stored vector that is shifted, or a stored vector that is not shifted, or even a vector that is not stored at all, is presented at the input plane, an error is declared. We shall begin our investigation by stating the necessary definitions and assumptions.

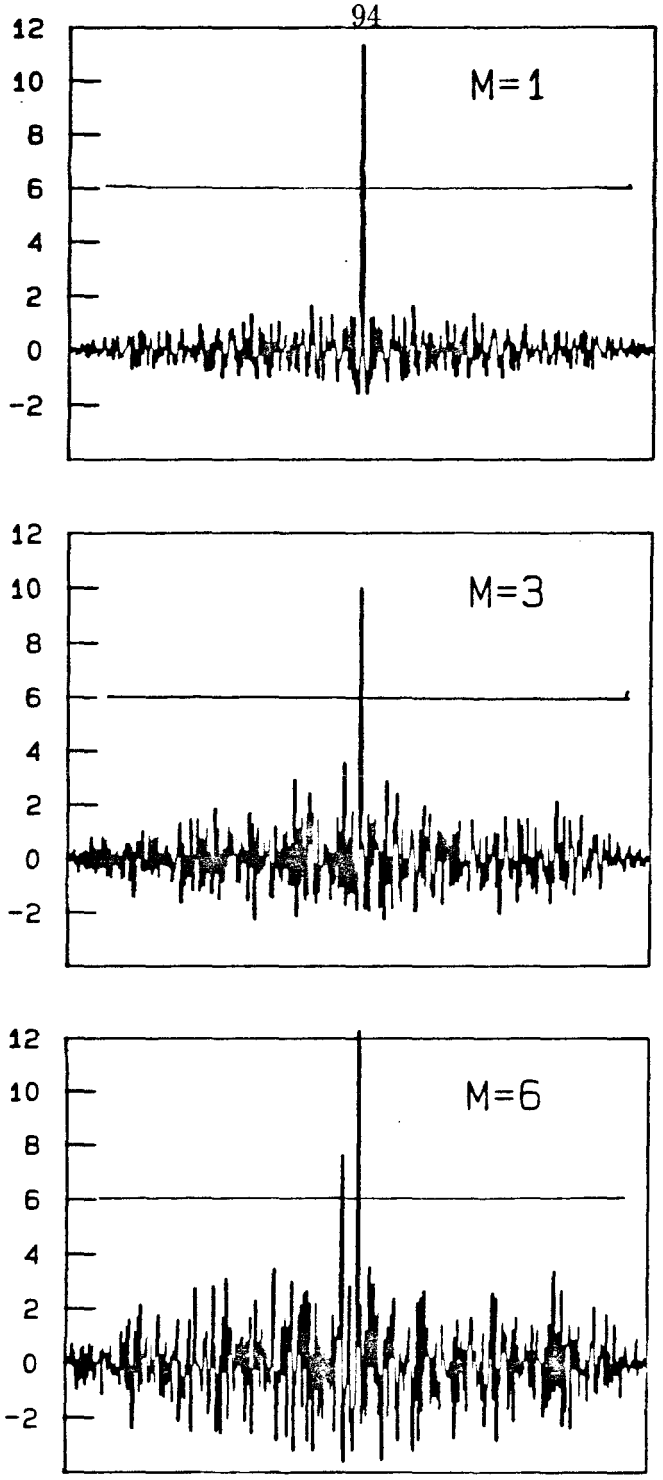


Figure III.3.1. Correlations of $\underline{\mathbf{f}}$ and $\underline{\mathbf{w}}$ with different number of vectors stored.

III.3.1. Definitions and Assumptions

Let

$$\left\{ \begin{array}{ll} \Omega_1 = \{\hat{\underline{f}}^{(\hat{m})}\} & \hat{m} = 1, \ldots, M, \text{ and} \\ \Omega_2 = \{\check{\underline{f}}^{(\check{m})}\} & \check{m} = 1, \ldots, M \end{array} \right\}$$

be two sets of one-dimensional vectors of N components to be classified. Suppose $\hat{f}_i^{(\hat{m})}$ and $\check{f}_j^{(\check{m})}$ are the *i-th* and the *j-th* components of $\hat{\underline{f}}^{(\hat{m})}$ and $\check{\underline{f}}^{(\check{m})}$, respectively, where $i, j = -\frac{N}{2}, \ldots, \frac{N}{2}$. Assume that $\{\hat{f}_i^{(\hat{m})}\}$ and $\{\check{f}_j^{(\check{m})}\}$ are statistically independent for all $i, j, \hat{m}, \check{m}$. Assume furthermore that these random variables are binary and bipolar valued (± 1) with equal probabilities.

Let each and every $\hat{\underline{f}}^{(\hat{m})}$ in Ω_1 and $\check{\underline{f}}^{(\check{m})}$ in Ω_2 be shifted k bits, k being a uniformly distributed random number. Let Φ be the set which contains all these shifted versions of $\hat{\underline{f}}^{(\hat{m})}$ and $\check{\underline{f}}^{(\check{m})}$. (We assume that only N output pixels are monitored, thus the maximum shift allowed in the input is also N. Note that there are a total of $2MN$ vectors in Φ.) As we have pointed out in section III.1.2, the ordering of the vectors in Φ does not affect the evaluation of the probability of correctness. We shall hence assume that the first M vectors in Φ are shifted versions of the vectors from Ω_1, and the last M vectors in Φ are shifted versions of the vectors from Ω_2. For simplicity, we shall rename all the vectors in Φ to be $\underline{f}^{(m)}$, $m = 1, \ldots, 2M$. Thus

$$\underline{f}^{(m)} = \hat{\underline{f}}^{(\hat{m})} \qquad m = 1, \ldots, M; \hat{m} = 1, \ldots, M,$$

and

$$\underline{f}^{(m)} = \check{\underline{f}}^{(\check{m})} \qquad m = M+1, \ldots, 2M; \check{m} = 1, \ldots, M.$$

Suppose $\underline{g}^{(m)} = (g_0^{(m)}, \ldots, g_l^{(m)}, \ldots, g_N^{(m)})$ $m = 1, \ldots, 2M$, represents the out-

put vector corresponding to $\underline{\mathbf{f}}^{(m)}$:

$$g_l^{(m)} = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i f_{i+l}^{(m)}, \qquad (III.3.1.1.)$$

where

$$w_i = \sum_{m=1}^{M} f_i^{(m)}.$$

After $\underline{\mathbf{g}}^{(m)}$ is obtained, every component of $\underline{\mathbf{g}}^{(m)}$ is compared to a threshold level τ. $\underline{\mathbf{f}}^{(m)}$ is classified to Ω_1 if there is one and only one $g_k^{(m)} > \tau$ and $\underline{\mathbf{f}}^{(m)}$ is classified to Ω_2 if there is $g_k^{(m)} < \tau$ for all k.

Now, an error occurs when any of the following happens:

a) $m > M$ but $g_k^{(m)} > \tau$

b) $m \leq M$ but no $g_k^{(m)} > \tau$

c) two or more $g_k^{(m)} > \tau$ for any m.

d) $m \leq M$ and one and only one $g_k^{(m)} > \tau$ but the true shift in $\underline{\mathbf{f}}^{(m)}$ is not k.

The performance of the correlator is said to be flawless if and only if the event $C = \{\forall m = 1, \ldots, M$, one and only one $g_k^{(m)} > \tau$ and the true shift in $\underline{\mathbf{f}}^{(m)}$ is k; and $\forall m = M+1, \ldots, 2M$, $g_k^{(m)} < \tau$ for all k $\}$ is true. C can also be expressed as the conjunction of a group of events. Let $C_m = \{g_k^{(m)} > \tau$ and the true shift in $\underline{\mathbf{f}}^{(m)}$ is $k\}$ if $\underline{\mathbf{f}}^{(m)}$ belongs to $\Omega_1\}$ or $C_m = \{g_k^{(m)} < \tau$ for all k if $\underline{\mathbf{f}}^{(m)}$ belongs to Ω_2 $\}$. Then

$$C = \bigcap_m C_m.$$

C_m can further be elaborated as

$$C_m = \{g_0^{(m)} < \tau; \ldots; g_{k-1}^{(m)} < \tau; g_k^{(m)} > \tau; g_{k+1}^{(m)} < \tau; \ldots; g_N^{(m)} < \tau,$$
$$\text{true shift in } \underline{\mathbf{f}}^{(m)} = k\} \quad m = 1, \ldots, M$$

or

$$C_m = \{g_k^{(m)} < \tau, \forall k\} \quad m = M+1, \ldots, 2M$$

Let $P[C_m]$ denote the probability that C_m is true.

For $m = 1, \ldots, M$: Using Bayes rule,

$$P[C_m] = \sum_{k=0}^{N} P[\{g_0^{(m)} < \tau; \ldots; g_{k-1}^{(m)} < \tau; g_k^{(m)} > \tau; g_{k+1}^{(m)} < \tau; \ldots; g_N^{(m)} < \tau, \\ |\text{true shift in} \underline{\mathbf{f}}^{(m)} = k\}] P[\{\text{true shift in } \underline{\mathbf{f}}^{(m)} = k\}]$$

In the following sections, we will show that the random variables $(g_0^{(m)}, .., g_N^{(m)})$ are statistically independent and jointly gaussian. In such case,

$$P[\{g_0^{(m)} < \tau; \ldots; g_{k-1}^{(m)} < \tau; g_k^{(m)} > \tau; g_{k+1}^{(m)} < \tau; \ldots; g_N^{(m)} < \tau, \\ |\text{true shift in} \underline{\mathbf{f}}^{(m)} = k\}] = p_m$$

for all k. Assuming that the shift in $\underline{\mathbf{f}}^{(m)}$ is equally likely for all positions,

$$P[C_m] = N p_m \frac{1}{N} = p_m$$

We may therefore assume without loss of generality that $k = 0$ and thus

$$P[C_m] = p_m = P[\{g_0^{(m)} > \tau; g_1^{(m)} < \tau; \ldots; g_N^{(m)} < \tau | \text{true shift is } 0\}]$$

The probability of correctness, $P[C]$, is given as

$$P[C] = P[\bigcap_{m}^{2} MC_m]$$

In the later sections, we will show that $\{C_m\}$ are statistically independent. After showing that, we can then write

$$p[C] = \prod_{m}^{2M} P[C_m] \qquad (III.3.1.2.)$$

and

$$P[C_m] = P[g_0^{(m)} > \tau]P[g_1^{(m)} < \tau]\cdots P[g_N^{(m)}] \qquad (III.3.1.3.)$$

The probabilities $P[C_m]$ will also be derived.

III.3.2. Evaluation of the Probability of Correctness

Let $\underline{\mathbf{z}}$ be the vector

$$(g_0^{(1)},\ldots,g_N^{(1)},g_0^{(2)},\ldots,g_l^{(m)}\ \ldots,g_N^{(2M)})$$

Note that this vector is composed of all the output data points due to all the vectors from Φ. Construct a set of vectors $\{\underline{\mathbf{x}}_i\}$ such that

$$\underline{\mathbf{z}} = \frac{1}{\sqrt{N}}\sum_i^N \underline{\mathbf{x}}_i \qquad (III.3.2.1a.)$$

In view of eq.(III.3.0.2.),

$$\underline{\mathbf{x}}_i = (f_i^{(1)}w_i,\ldots,f_{i+N}^{(1)}w_i,f_i^{(2)}w_i\ \ldots,f_{i+l}^{(m)}w_i,\ldots,f_{i+N}^{(2M)}w_i) \qquad (III.3.2.1b.)$$

If we can show that $\{\underline{\mathbf{x}}_i\}$ are statistically independent, then we may invoke the MVCLT to obtain the joint probability density function of $\underline{\mathbf{z}}$. To show that $\{\underline{\mathbf{x}}_i\}$ are statistically independent is equivalent to showing that any component of $\underline{\mathbf{x}}_i$ is independent of all the other components of the remaining vectors $\underline{\mathbf{x}}_j$, $j \neq i$. In other words, we have to show that $f_{i+l}^{(m)}w_i$ is independent of all the elements in the set A where

$$A = \{f_{j+k}^{(n)}w_j\}, i,j = 1,\ldots,N; j \neq i; k,l = 1,\ldots,N; n,m = 1,\ldots,2M.$$

A can be divided into two subsets, A_1 and A_2:

$$A_1 = \{\text{all terms containing } f_i;$$
$$\text{all terms containing } f_{i+l}^{(n)};$$
$$\text{all terms containing } w_i + l\}$$

and $A_2 = A \backslash A_1$. Note that all the terms in A_2 are statistically independent of $f_{i+l}^{(m)} w_i$. To show complete independence, we have to show that $f_{i+l}^{(m)} w_i$ is independent of A_1 and A_2 simultaneously, which can be accomplished by showing the following:

$$p[f_{i+l}^{(m)} w_i | A_1; A_2] = p[f_{i+l}^{(m)} w_i]$$

But $f_{i+l}^{(m)} w_i$ is independent of A_2 conditioned on A_1, thus

$$p[f_{i+l}^{(m)} w_i | A_1; A_2] = p[f_{i+l}^{(m)} w_i | A_1]$$

Let

$$S_1 = \{\text{all terms containing } f_i;\}$$
$$S_2 = \{\text{ all terms containing } f_{i+l}^{(n)};\}$$
$$S_3 = \{\text{ all terms containing } w_i + l\}$$

Thus

$$p[f_{i+l}^{(m)} w_i | A_1] = p[f_{i+l}^{(m)} w_i | S_1; S_2; S_3]$$

Using the fact that $comfimfi + lm = \pm 1$, we invoke the Bayes rule to obtain

$$p[f_{i+l}^{(m)} w_i = c | S_1; S_2; S_3] =$$
$$p[w_i = c | S_1; S_2; S_3; f_{i+l}^{(m)} = 1] P[f_{i+l}^{(m)} = 1 | S_1; S_2; S_3]$$
$$p[w_i = -c | S_1; S_2; S_3; f_{i+l}^{(m)} = -1] P[f_{i+l}^{(m)} = -1 | S_1; S_2; S_3]$$

In view of the definitions of S_1, S_2, and S_3, w_i is independent of S_2 and S_3 conditioned on S_1. Hence the above equation becomes

$$\begin{aligned} p[f_{i+l}^{(m)} w_i = c | S_1; S_2; S_3] = \\ p[w_i = c | S_1] P[f_{i+l}^{(m)} = 1 | S_1; S_2; S_3] \\ + p[w_i = -c | S_1] P[f_{i+l}^{(m)} = -1 | S_1; S_2; S_3] \end{aligned} \qquad (III.3.2.2.)$$

To show that S_1 is independent of w_i is equivalent to showing that

$$p[S_1 | w_i] = p[S_1].$$

Using the definition for S_1, we have

$$p[S_1 | w_i] = p[f_i^{(n)} w_a = u; f_i^{(n)} w_b = v, \ldots | w_i] \qquad (III.3.2.3.)$$

$$a, b, \ldots \neq i$$

By invoking the Bayes rule again,

$$\begin{aligned} p[S_1 | w_i] &= p[w_a = u; w_b = v; \ldots | w_i; f_i^{(n)} = 1] P[f_i^{(n)} = 1 | w_i] \\ &\quad + p[w_a = -u; w_b = -v; \ldots | w_i; f_i^{(n)} = -1] P[f_i^{(n)} = -1 | w_i] \\ &= p[w_a] p[w_b] \ldots \{ P[f_i^{(n)} = 1 | w_i] + P[f_i^{(n)} = -1 | w_i] \} \end{aligned}$$

Here we have used the fact that $p[w_a = u] = p[w_a = -u]$ due to the symmetry in the density function of $f_i^{(m)}$. Since $f_i^{(n)}$ can only be ± 1, the sum within the $\{\}$ in the above equation must be 1. Because $p[w_a]$ is symmetric and w_a is independent of $f_i^{(n)}$,

$$p[w_a] = p[w_a f_i^{(n)}]$$

Substituting all these results in eq.(III.3.2.3.), the following results:

$$p[S_1|w_i] = p[w_a f_i^{(n)}]p[w_b f_i^{(n)}]\ldots$$

$$= p[S_1]$$

In other words, w_i is independent of S_1. Using this result, eq.(III.3.2.2.) can be written as

$$p[f_{i+l}^{(m)} w_i = c|S_1; S_2; S_3] =$$

$$p[w_i = c]P[f_{i+l}^{(m)} = 1|S_1; S_2; S_3]$$

$$+ p[w_i = -c]P[f_{i+l}^{(m)} = -1|S_1; S_2; S_3]$$

$$= p[w_i]$$

$$= p[f_{i+l}^{(m)} w_i]$$

We have indeed shown that $\{\underline{x}_i\}$ are statistically independent.

Observe that eq.(III.3.2.1.) conforms with the expression of the MVCLT. As a consequence $\underline{z}$ approaches jointly gaussian form as N tends to infinity. The mean vector and the covariance matrix of $\underline{z}$ will be calculated in the next section.

III.3.3. Evaluation of the Mean Vector and the Covariance Matrix

The mean of the element $g_l^{(m)}$ is evaluated as follows:

$$E[g_l^{(m)}] = \frac{1}{\sqrt{N}} \sum_i^N E[f_{i+l}^{(m)} w_i] = 0 \quad l \neq 0$$

$$E[g_0^{(m)}] = \frac{1}{\sqrt{N}} \sum_i^N E[f_i^{(m)} w_i] \qquad (III.3.3.1.)$$

Note that eq.(III.3.3.1.) is identical to eq.(III.1.5.1.), therefore,

$$E[g_0^{(m)}] = \sqrt{N} E[(f_i^{(m)})^2] = \sqrt{N}\sigma^2$$

Thus the mean vector of $\underline{z}$ is given as

$$E[\underline{z}] = (\sqrt{N}\sigma^2, 0, \ldots, 0, \sqrt{N}, 0 \ldots, 0)$$

Note that there are $N-1$ zeros between every two $\sqrt{N}\sigma^2$ for the first NM terms (so there are a total of M $\sqrt{N}\sigma^2$ elements). The last NM terms of $E[\underline{z}]$ are all zeros.

The covariances of $\underline{z}$ are computed as follows:

$$\begin{aligned} var[g_l^{(m)} g_k^{(n)}] &= \frac{1}{N} \sum_{i}^{N} \sum_{j}^{N} E[f_{i+l}^{(m)} w_i f_{j+k}^{(n)} w_j] \\ &= \frac{1}{N} \sum_{i}^{N} E[f_{i+l}^{(m)} f_{j+k}^{(n)} (w_i)^2] \\ &+ \frac{1}{N} \sum_{i}^{N} \sum_{\substack{j \\ j \neq i}}^{N} E[f_{i+l}^{(m)} w_i f_{j+k}^{(n)} w_j] \\ &= \begin{cases} E[(w_i)^2] & \text{for } i+l = j+k \text{ and } m = n; \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

where

$$\begin{aligned} E[(w_i)^2] &= \sum_{m}^{M} E[(f_i^{(m)})^2] + \sum_{m}^{M} \sum_{n}^{M} E[f_i^{(m)}] E[f_i^{(n)}] \\ &= M\sigma^2 \end{aligned}$$

From the above equation, we see that the covariance matrix of $\underline{z}$ has only non-zero (and identical) diagonal terms. All the off-diagonal terms are zeros. Since $\underline{z}$ is jointly gaussian, we can therefore say that $\{g_l^{(m)}\}$ are statistically independent for all l and m.

III.3.4. Evaluation of Capacity

Due to the fact that $\{g_l^{(m)}\}$ are statistically independent,

$$P[C_m] = P[g_0^{(m)} > \tau]P[g_1^{(m)} < \tau]\dots P[g_N^{(m)} < \tau] \qquad (III.3.4.1.)$$

Now

$$P[g_1^{(m)} < \tau] = \cdots = P[g_N^{(m)} < \tau] = \int_{-\infty}^{\tau} \frac{1}{\sqrt{2\pi M}} exp(-\frac{\alpha^2}{2M})d\alpha \qquad (III.3.4.2a.)$$

and

$$P[g_0^{(m)} > \tau] = \int_{\tau}^{\infty} \frac{1}{\sqrt{2\pi M}} exp(-\frac{(\alpha^2 - \sqrt{N})}{2M})d\alpha \qquad (III.3.4.2b.)$$

If we choose $\tau = \sqrt{N}/2$ then

$$P[g_0^{(m)} > \tau] = P[g_1^{(m)} < \tau] = 1 - Q(\sqrt{\frac{N}{2M}})$$

where

$$Q(\sqrt{\frac{N}{2M}}) = \int_{\sqrt{\frac{N}{2M}}}^{\infty} \frac{1}{\sqrt{2\pi}} exp(-\frac{\alpha^2}{2})d\alpha \qquad (III.3.4.3.)$$

Substituting the above equation into eq.(III.3.4.1.),

$$P[C_m] = \left(1 - Q(\sqrt{\frac{N}{2M}})\right)^N \qquad (III.3.4.4.)$$

By the statistical independence of $\{g_l^{(m)}\}$, $\{C_m\}$ are also statistically independent. In addition, $P[C_m]$ are identical for all m because of the identical probability density functional forms of $(g_0^{(m)}, \dots, g_N^{(m)})$. From these statistical properties of C_m, eq.(III.3.1.2.) and eq.(III.3.4.4.), it follows that

$$P[C] = \left(1 - Q(\sqrt{\frac{N}{2M}})\right)^{2MN} \qquad (III.3.4.5.)$$

The capacity can be obtained by forcing $P[C]$ arbitrarily close to 1. Following the same procedure described by eq.(III.1.6.9.) through eq.(III.1.6.11.), the transcendental equation

$$M \approx \frac{N}{8ln(M\sqrt{MN})}$$

is arrived at. Using the upper and lower bound argument described by equation (III.1.6.12.), the capacity of the multiple exposure correlation hologram is found to be

$$M = \frac{N}{16ln(N)}$$

Comparing this result with the capacity of multiple exposure dot-product hologram (eq.(III.1.6.14.)), the cost of having shift invariance is only a reduction by a factor of 2 in the storage capacity. This result, however, should not be interpreted as the storage capacity of NM (N shifts) patterns. It is because each shifted pattern cannot be counted as an independent individual pattern. For a shifted pattern is almost totally determined (except for the amount of shift) once the unshifted pattern is defined.

III.4. Binary Correlation Filter

In this section, we will investigate the effect on storage capacity by thresholding the multiple exposure correlation filter described in section III.3. We will denote the binary multiple exposure correlator as BMEC and the reference hologram as BMEH. Apparently, the storage capacity of the BMEC is lower than that of MEC due to thresholding and that of BMED due the addition of shift-invariance. The goal of this section is to derive the sotrage capacity of the BMEC.

III.4.1. Definitions and assumptions

Same as section III.3.1 with the exception that the probability density function of $f_i^{(m)}$ is symmetric about the origin and with variance σ_f^2 here.

III.4.2. Formulation of the Probability of Correctness

Note that the formulation given in section III.3.2 did not place requirement on the form of the correlation filter. We may therefore use the entire formulation here without any modification.

III.4.3. Evaluation of the Probability of Correctness

The BMEH $\underline{\mathbf{w}} = (w_1, \ldots, w_i, \ldots, w_N)$ is defined as

$$w_i = \beta[\sum_{m=1}^{M} \hat{f}_i^{(m)}]$$

which is identical to eq.(III.2.0.1a.). The correlation output is given as

$$g_l^{(m)} = \sum_{i}^{N} f_i^{(m)} w_{i+l}$$

To obtain the probability of correctness, consider the vector $\underline{\mathbf{z}}$:

$$\underline{\mathbf{z}} = (g_0^{(1)}, \ldots, g_N^{(1)}, g_0^{(2)}, \ldots, g_l^{(m)} \ldots, g_N^{(2M)}) \qquad (III.4.3.1.)$$

Construct a set of vectors $\{\underline{\mathbf{x}}_i\}$ such that

$$\underline{\mathbf{z}} = \frac{1}{\sqrt{N}} \sum_{i}^{N} \underline{\mathbf{x}}_i \qquad (III.4.3.2a.)$$

In view of eq.(III.4.3.1.),

$$\underline{\mathbf{x}}_i = (f_i^{(1)} w_i, \ldots, f_i^{(1)} w_i + N, f_i^{(2)} w_i \ldots, f_i^{(m)} w_i + l, \ldots, f_i^{(2M)} w_i + N) \quad (III.4.3.2b.)$$

Compare eq.(III.4.3.2.) with eq.(III.3.2.1.). In eq.(III.3.2.1.), all the variables in f are binary and bipolar, and all the variables in w are symmetric about zero. In contrast, all the variables in f are symmetric about zero, and all the variables in w are binary and bipolar in eq.(III.4.3.2.). Recall that the proof of independence in section III.3. does not require the exact form $\underline{\mathbf{w}}$. The only requirement is that variables in w and f be statistically independent if their subscripts are different. This requirement is certainly satisfied here. Therefore, there is a one-to-one correspondence between the problem in this section and that in section III.3.3. In other words, $\{g_i^{(m)}\}$ must also be jointly gaussian. The mean vector and the covariance matrix of $\underline{\mathbf{z}}$, however, depends on the exact form of $\underline{\mathbf{w}}$. This means the probability of correctness of the BMEC is different from that of the MEC. The mean vector, covariance matrix, and the probability of correctness are derived in the next section.

III.4.4. Evaluation of of the Mean Vector and the Covariance Matrix

The mean of the element $g_l^{(m)}$ is evaluated as follows:

$$E[g_l^{(m)}] = \frac{1}{\sqrt{N}} \sum_i^N E[f_i^{(m)}] E[w_i + l] = 0 \quad l \neq 0$$

$$E[g_0^{(m)}] = \frac{1}{\sqrt{N}} \sum_i^N E[f_i^{(m)} w_i] \qquad (III.4.4.1.)$$

Note that eq.(III.4.4.1.) is identical to eq.(III.2.2.2.), therefore,

$$E[g_0^{(m)}] = \sqrt{N} E[(f_i^{(m)})^2] = \sqrt{\frac{2N}{\pi M}}$$

Thus the mean vector of $\underline{z}$ is given as

$$E[\underline{z}] = (\sqrt{\frac{2N}{\pi M}}, 0, \ldots, 0, \sqrt{\frac{2N}{\pi M}}, 0 \ldots, 0)$$

Note that there are $N-1$ zeros between every two $\sqrt{\frac{2N}{\pi M}}$ for the first NM terms. The last NM terms of $E[\underline{z}]$ are all zeros.

The covariances of $\underline{z}$ are computed as follows:

$$\begin{aligned} var[g_l^{(m)} g_k^{(n)}] &= \frac{1}{N} \sum_i^N \sum_j^N E[f_i^{(m)} w_i + l f_j^{(n)} w_j + k] \\ &= \frac{1}{N} \sum_i^N E[(f_i^{(m)})^2 w_i + l w_j + k] \\ &+ \frac{1}{N} \sum_i^N \sum_{\substack{j \\ j \neq i}}^N E[f_i^{(m)} w_i + l f_j^{(n)} w_j + k] \\ &= \begin{cases} E[(f_i^{(m)})^2] & \text{for } i + l = j + k \text{ and } m = n; \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

where

$$E[(f_i m)^2] = \sigma^2$$

From the above equation, we see that the covariance matrix of $\underline{z}$ has only non-zero (and identical) diagonal terms. All the off-diagonal terms are zeros. Since $\underline{z}$ is jointly gaussian, we can therefore say that $\{g_l^{(m)}\}$ are statistically independent for all l and m.

III.4.4. Capacity

Due to the similar forms of the joint probability density functions of $\underline{z}$ of this section and that of section III.3., and the identical formulations in the probability of

correctness in section III.4. and III.3., the probability of correctness of the BMEC must be given as (refer to eq.III.3.)

$$P[C] = \left(1 - Q(\sqrt{\frac{2N}{\pi M}})\right)^{2MN}$$

where

$$Q(\sqrt{\frac{2N}{\pi M}}) = \int_{\sqrt{\frac{2N}{\pi M}}}^{\infty} \frac{1}{\sqrt{2\pi}} exp(-\frac{\alpha^2}{2}) d\alpha$$

The capacity can be obtained by forcing $P[C]$ arbitrarily close to 1. Following the same procedure described by eq.(III.1.6.9.) through eq.(III.1.6.11.), the transcendental equation

$$M \approx \frac{N}{4\pi ln(M\sqrt{MN})}$$

is arrived at. Using the upper and lower bound argument described by equation (III.1.6.12.), the capacity of the multiple exposure correlation hologram is found to be

$$M = \frac{N}{8\pi ln(N)}.$$

Compare the capacity of the binary correlation filter and that of the fully analog correlation filter, we see that the loss in the binary correlation filter is also $\frac{\pi}{2}$. This loss is identical to that of the inner-product filter suffered from a 1-bit dynamic range.

References for Chapter III

1. Y. C. Lee, G. Doolen, H. H. Chen, G. Z. Sun, T. Maxwell, H. Y. Lee, and Lee Giles, "Machine Learning Using a Higher Order Correlation Network," Snowbird Conference on Associative Memory, Snowbird, UT (1986).
2. J. Hong and D. Psaltis, "Storage Capacity of Holographic Associative Memories," *Opt. Letters*,
3. D. Psaltis and J. Hong, "Shift Invariant Optical Associative Memories,"
4. C. Hester, and D. Casasent, "Multi-variant Technique for Multi-Class pattern Recognition," *Appl. Opt.*, **19**, 1758 (1980).
5. G. Turin, "An Introduction to Matched Filters," *IRE Trans. Inform. Theory*, **IT-6**, 311 (1960).
6. A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, McGraw Hill, New York (1965).
7. T. Anderson and R. Bahadur, "Classification into Two Multivariate Normal Distributions with Different Covariance Matrices," *Ann. Math. Stat.*, **33**, 420 (1962).
8. W. Chaplin and V. Levadi, "A Generalization of the Linear Threshold Decision Algorithm to Multiple Classes," in *Computer and Information Sciences – II,* pp.337-354, J. Tou, ed., Academic Press, New York (1953).
9. S. Gass, *Linear Programming*, Third Edition, McGraw-Hill, New York (1969).
10. M. Minsky and S. Papert, *Perceptrons: An Introduction to Computational Geometry*, MIT Press, Cambridge, Mass. (1969).
11. J. Lee, *Numerical Analysis for Computers,* Reinhold Publishing Corp., New York (1966).

12. O. Veblen and J. Whitehead, *The Foundations of Differential Geometry,* Cambridge University Press, London (1932).

13. H. Manning, *Geometry of Four Dimensions*, The MacMillan Co., New York (1914).

14. A. Vander Lugt, "Signal Detection by Complex Spatial Filtering," *IEEE Trans Inf Th,* **IT-10**, 139 (1964).

15. J. M. Wozencraft and I. M. Jacobs, *Principles of Communication Engineering,* Wiley and Sons, New York (1965).

16. E. Paek and D. Psaltis, "Optical Associative Memory using Fourier Transform Holograms," *Opt. Eng.,* **26**, 428 (1987).

17. D. Gabor, "Associative Holographic Memories," *IBM Jour. Res. Devel.,* **13**, 156 (1969).

18. H. Caulfield, "Linear Combinations of Filters for Character Recognition: A Unified Treatment," *appl. Opt.,* **19**, 3877 (1980).

19. D. Casasent, "Unified Synthetic Discriminant Function: Computational Formulation," *Appl. Opt.*, **23**, 1620 (1984).

20. F. Rossenblatt, *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*, Spartan Books, Washington D.C., (1961).

21. J. Hong, "Optical Computing for Adaptive Signal Processing and Associative Memories," *Ph.D. Dissertation,* California Institute of Technology (1987).

22. J. Yu, "Optical Processing using Photorefractive Crystals," *Ph.D. Dissertation,* California Institute of Technology (1988).

23. T. Cover, "Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition," *IEEE Trans. Elec. Comp.,*

EC-14, 326 (1965).

24. J. Yu, F. Mok, and D. Psaltis, "Capacity of Optical Correlators," *SPIE* **825-22** (1987).
25. R. Duda and P. Hart, *Pattern Classification and Scene Analysis*, John Wiley and Sons, New York (1973).
26. N. Nilsson, *Learning Machines: Foundations of Trainable Pattern-Classifying Systems*, McGraw-Hill, New York (1965).
27. A. Oppenheim and R. Schafer, *Digital Signal Processing*, Prentice-Hall, New Jersey (1975).
28. W. Feller, *An Introduction to Probability Theory and Its Applications*, Wiley and Sons, New York (1970).

IV. Quantization effects on training algorithms

IV.0 Introduction

Suppose we have a finite set Φ of distinct patterns $\{\underline{x}^{(1)}, \underline{x}^{(2)}, \ldots, \underline{x}^{(M)}\}$. Let each pattern in Φ belong to one of two categories. This classification divides Φ into subsets Ω_1 and Ω_2 such that each pattern in Ω_i belongs to category i for $i = 1, 2$.

As we pointed out in the introduction of Chapter III, pattern classification can be achieved by a class of machines known as linear machines. Basically, a linear machine is the realization of a linear discriminant function $g(\underline{x})$. $g(\underline{x})$, on the other hand, is a real-valued linear function of the components of the input pattern $\underline{x}$:

$$g(\underline{x}) = w_1 x_1 + w_2 x_2 + \cdots + w_N x_N + w_{N+1}. \qquad (IV.0.1)$$

(Refer to Figure IV.0.1).

The vector $\underline{w} = (w_1, \ldots, w_N)$ is the weight vector, and the element w_{N+1} is the threshold level of $g(\underline{x})$. When $\underline{x}$ is subjected to a linear machine, the discriminant, or the value $g(\underline{x})$ is computed. The discriminant is then applied to a decision unit which we refer to as a neuron. In general, we will take the neuron to be a hard thresholding device biased at zero, i.e., an output 1 results if the discriminant is more than zero while a output 0 will be obtained otherwise. (Note that w_{N+1} in the above equation actually emulates a variable thresholding level.) Based on this architecture, $\underline{x}$ can be classified to either Ω_1 or Ω_2 depending on the output of the neuron. Without loss of generality, we may assume

$$g(\underline{x}) > 0 \qquad \forall \underline{x} \in \Omega_1$$

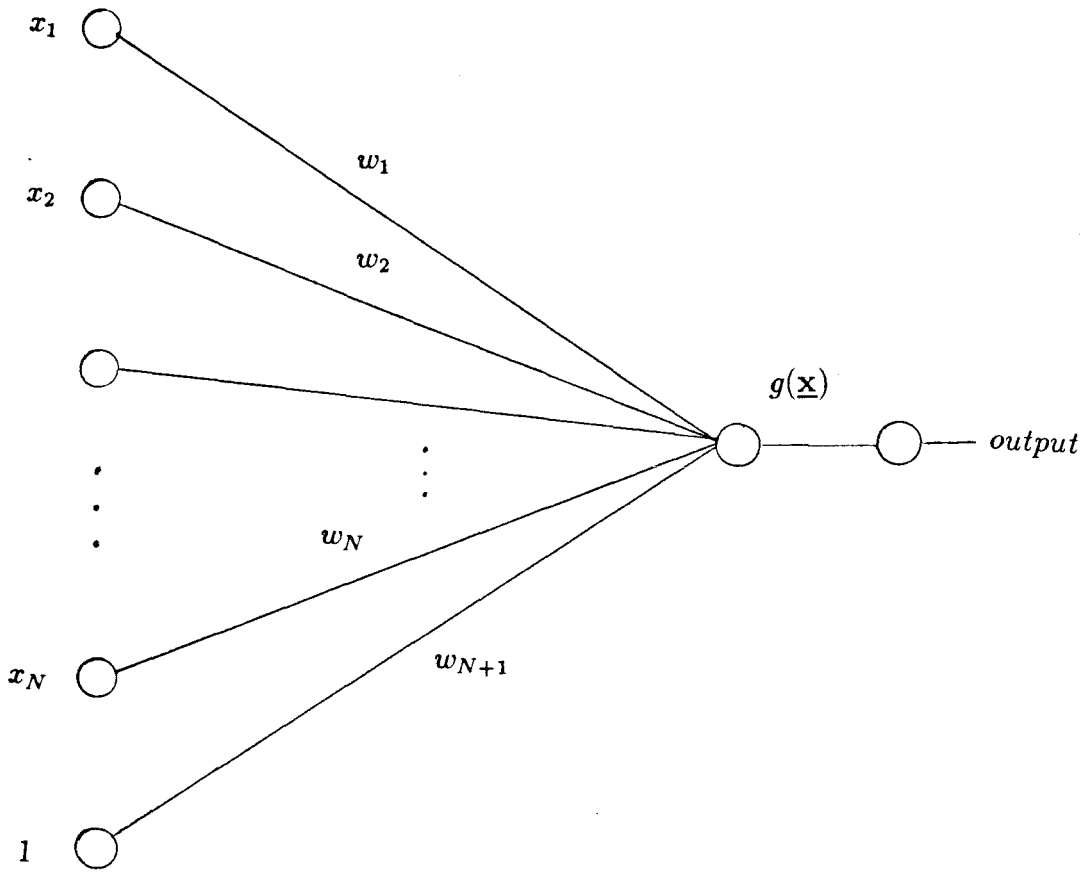


Figure IV.0.1. The schematic diagram of a linear discriminant function.

$$g(\underline{\mathbf{x}}) < 0 \qquad \forall \underline{\mathbf{x}} \in \Omega_2 \tag{IV.0.2}$$

If a linear machine can be found such that each of the patterns in Φ can be placed into the proper categories, we say that Ω_1 and Ω_2 are linearly separable.

In order to correctly classify a set of objects into two categories according to some predetermined assignment, the correct $\underline{\mathbf{w}}$ has to be determined. (The multiple exposure algorithm described in Chapter III is a special way of obtaining $\underline{\mathbf{w}}$.) We call the process of obtaining $\underline{\mathbf{w}}$ "training." (Training is not a term confined to describing only the learning process of $\underline{\mathbf{w}}$ of a linear machine. It applies to more complicated machines, e.g., multi-layer neural networks. But in the remainder of this chapter, we will focus only on single-layer linear machines.)

It is evident that $\underline{\mathbf{w}}$ is a function of the patterns to be classified. To train $\underline{\mathbf{w}}$, a cost (error) function E, which is usually a function of the input patterns, their desired responses, and $\underline{\mathbf{w}}$, is defined. This cost function, of course, has to reflect the goal to be achieved, e.g., classification of all input patterns correctly. For instance, E can be defined as

$$E = \sum_i \left(f(\underline{\mathbf{w}} \cdot \underline{\mathbf{x}}^{(i)}) - t_i\right)^2 \tag{IV.0.3}$$

where

$$f(v) = \begin{cases} 0 & \text{if } v < 0; \\ 1 & \text{if } v \geq 0, \end{cases}$$

is the transfer function of the output neuron, t_i is the desired response for $\underline{\mathbf{x}}^{(i)}$, and the summation is over all input patterns. For the above example, we observe that E is never negative, becoming zero if all the actual reponses, $f(\underline{\mathbf{w}} \cdot \underline{\mathbf{x}}^{(i)})$, coincide with the target responses. The objective of the training process now becomes the process

of minimizing the cost function. In general, input (training) patterns are subjected to the linear machine in groups or in sequence in the training process. The weights w_i are then modified so that the cost is reduced. This process is continued until convergence (minimization) occurs, or the process is instructed to halt by some other criterion.

Whereas the inner-product of an optical trainable linear machine is always implemented by optics (in our case), the training process can be realized in several ways. Currently, the optical trainable linear machines that are under research can be divided into three types of architectures:

1. Optical training. Both the training part and the dot-product part are done by optics. This can be very difficult since complex operations cannot usually be implemented by optics easily.
2. Digital training. The weight vector is trained by a digital computer. The trained weight vector is then written onto a SLM. The recognition , or dot-product, part is achieved by optics.
3. Electro-optic training. This architecture is a hybrid of (1.) and (2.). The training of the weight vector is achieved by utilizing optics and electronics simultaneously. While the optics handle the dot-product part (used for verifying the output response), the electronics is responsible for updating the weight vector.

Aside from the time and extra hardware requirement we pointed out earlier, there are other practical concerns in the implementation of an optical trainable classifier. One of them is imposed by the limited dynamic range of physical devices.

Because of quantization, a training algorithm which works perfectly theoretically may produce unexpected results. Even when the weight vector is trained on a digital computer, it may not work properly when it is downloaded onto a realistic SLM. The purpose of this chapter is to analyze a) the effect caused by limited dynamic range on training, and b) the ability to classify when a realistic SLM is used to record the weight vector trained by algorithms that are proven to work theoretically. Due to the fact that quantization is a highly nonlinear operation, theoretical analysis is extremely difficult. Therefore, qualitative explanations and statistical results, in addition to some examples, will be given to illustrate the effects on different algorithms caused by devices with limited dynamic range.

IV.1 Quantization Effect on Architecture 1.

IV.1.1 Description

Suppose $\underline{\mathbf{x}}$ is a pattern to be classified. To simplify discussion, we augment $\underline{\mathbf{x}}$ by a $(N+1)$-th component (w_{N+1} in eq.IV.0.1.) whose value is always equal to unity. We shall denote this augmented pattern vector by the symbol $\underline{\mathbf{y}}$. The components of $\underline{\mathbf{y}}$ will be given by $y_1, y_2, \ldots, y_{N+1}$; $y_i = x_i$ for $i = 1, \ldots, N$, and $y_{N+1} = +1$. A linear discriminant function of $\underline{\mathbf{x}}$ can now be written in terms of $\underline{\mathbf{y}}$ in the simple form

$$g(\underline{\mathbf{x}}) = \underline{\mathbf{w}} \cdot \underline{\mathbf{y}},$$

where $\underline{\mathbf{w}} = (w_1, \ldots, w_N, w_{N+1})$. We may further simplify the notation by multiply-

ing all vectors in Ω_2 by -1. The classification criterion then becomes

$$g(\underline{\mathbf{x}}) = \underline{\mathbf{w}} \cdot \underline{\mathbf{y}} > 0 \quad \forall \underline{\mathbf{y}} \in \Phi. \tag{IV.1.1}$$

A well known algorithm to train $\underline{\mathbf{w}}$ is the perceptron algorithm. To train $\underline{\mathbf{w}}$, let $\{\underline{\mathbf{y}}^{(i)}\}$ be presented to the LDF in a cyclic sequence. The perceptron training algorithm can be described as follows:

i.) assume one of the vectors $\underline{\mathbf{y}}$ to be the initial $\underline{\mathbf{w}}^{(0)}$.

ii.) suppose $\underline{\mathbf{w}}^{(k)}$ stands for the training weight vector at the k-th iteration,$k = 0, 1, 2...$, then

$$\underline{\mathbf{w}}^{(k+1)} = \underline{\mathbf{w}}^{(k)} + c\underline{\mathbf{y}}^{(k)}, \tag{IV.1.1.2}$$

where

$$c = \begin{cases} 0 & \text{if } \underline{\mathbf{w}}^{(k)} \cdot \underline{\mathbf{y}}^{(k)} > 0; \\ 1 & \text{if } \underline{\mathbf{w}}^{(k)} \cdot \underline{\mathbf{y}}^{(k)} \leq 0, \end{cases}$$

and $\underline{\mathbf{y}}^{(k)}$ is a training pattern in the cyclic sequence.

iii.) repeat ii.) until $\{\underline{\mathbf{y}}_i\}$ are all classified correctly.

If $\{\underline{\mathbf{y}}_i\}$ is linearly separable, then it can be shown that the $\underline{\mathbf{w}}$ trained by the perceptron algorithm will minimize E given by eq.(IV.0.3), assuming $t_i = \pm 1$, to zero. In other words, $\underline{\mathbf{w}}$ so trained will converge to a vector which is a solution of eq.(IV.1.1). If the number of vectors to be trained, M, is not greater than the dimensionality, N, the average convergence time is usually linear with N (observed in our experiment). The dynamics of convergence is shown in Fig.IV.1.1. It can be shown that the solution region (any vector in the solution region will classify $\{\underline{\mathbf{y}}_i\}$ correctly) in weight space is always convex. And the correction, given in ii.),

will always move the training weight vector toward the solution region. Hence, a solution weight vector will always result.

In the proof of the perceptron convergence theorem, it is assumed that the components of $\underline{w}$ can take any real value. This assumption, however, cannot be abided by in any realistic implementation. If the implementation is realized on a digital computer, the quantization effect usually does not pose any serious problem. On the other hand, if the training procedure is implemented by an analog optical computer, the limited dynamic range of the medium used to record the trained $\underline{w}$ loses information about the training patterns. Imagine the recording medium having a dynamic range of K levels. In the N-dimensional weight space, the possible values that $\underline{w}$ may assume are represented by a grid of K^N points. This quantization effect imposes two problems on the training process. The first is that the solution region may not contain any of these allowed points. The second problem is caused by the nonlinear operation associated with the quantization effect. After each iteration, $\underline{w}$ must be one of the K^N points (there are only K^N allowable states in the recording medium). The quantization effect is such that the analog $\underline{w}$ (before being quantized) after the correction will be attracted to the closest discrete point on the grid. (See Fig.IV.1.2).

When the analog $\underline{w}$ is not in the solution region, quantization does not necessarily handicap the training procedure. (As a matter of fact, quantization may sometimes even take the analog $\underline{w}$ into the solution region.) Problems do occur whenever the analog $\underline{w}$ is in the solution region. Now as long as training continues, a quantized solution weight vector may still be obtained, that is, provided that it exists. However, if the correction is not large enough, the same quantized $\underline{w}$ may

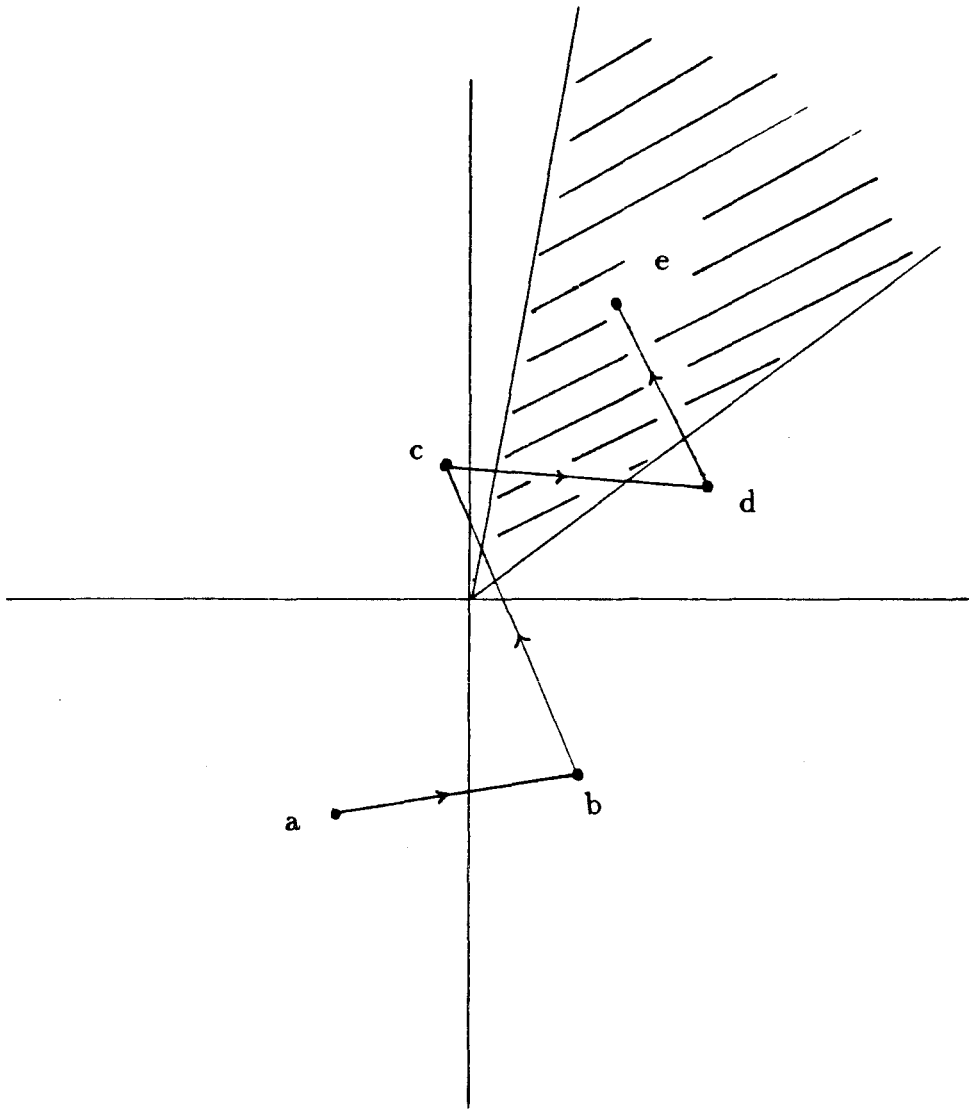


Figure IV.1.1.The dynamics of convergence of a perceptron.

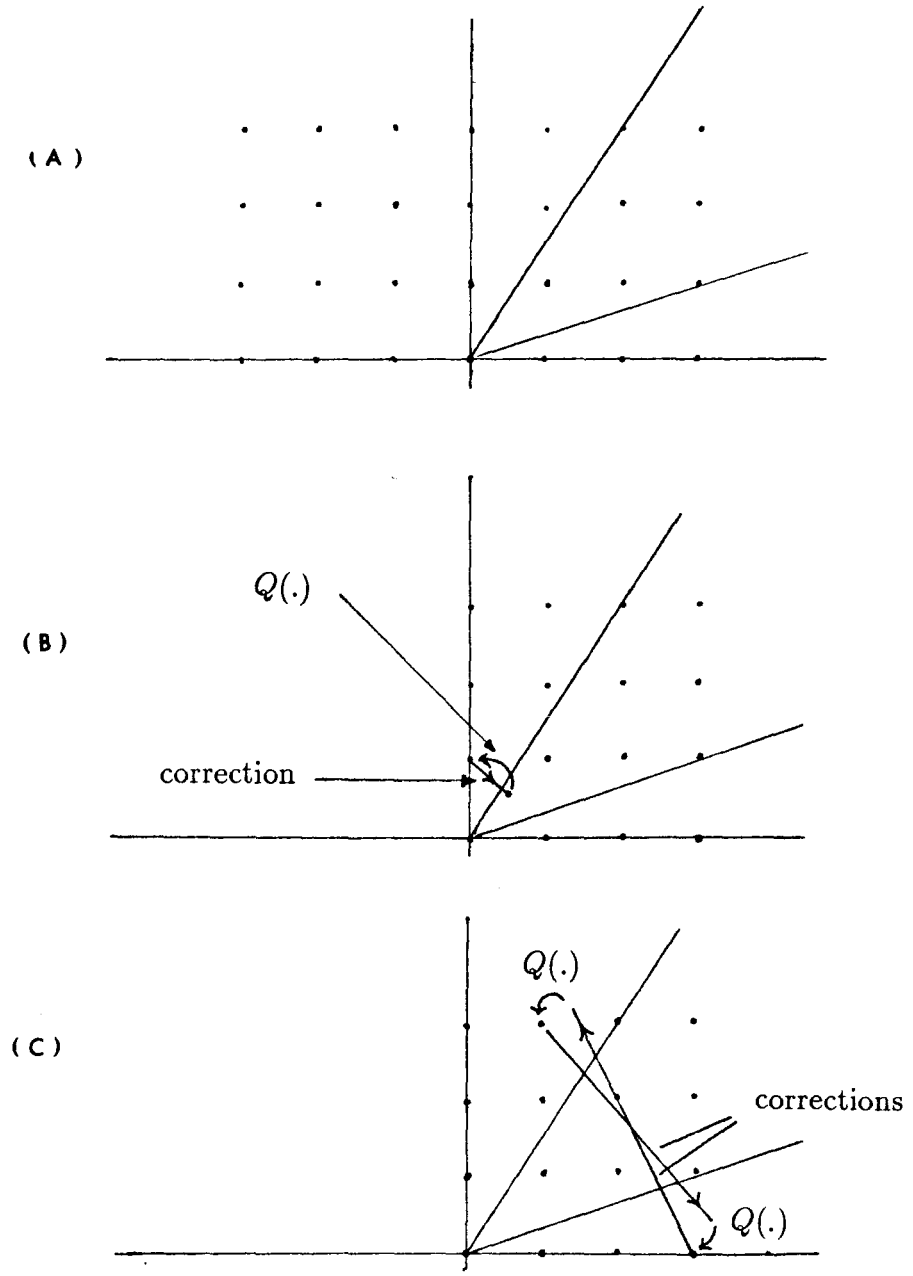


Figure IV.1.2. Quantization effect on training. (a) A typical solution region. (b) Correction not large enough — the same quantum point is resulted every time. (c) Corrections too large — overshooting and oscillation result.

result after each iteration. On the other hand, if the correction is so large that the $\underline{w}$ overshoots, then oscillation, between several quantized points lying outside of the solution region, may happen. In a regular perceptron, correction steps can also be so large that they overshoot the solution region. But the corrections always take the training $\underline{w}$ to the "wider" part (farther away from the origin) of the solution region and hence guarantee that $\underline{w}$ will eventually be in the solution region. In a dynamic range-limited medium, however, quantization can prevent the training $\underline{w}$ from moving toward the "wider" part of the solution region. In general, oscillation is what happens whenever a perceptron with limited dynamic range fails to converge. Since the input, or correction, is also dynamic range-limited, K does not have to be infinite for $\underline{w}$ to be properly trained. As a matter of fact, if the components of the input patterns are binary- and bipolar-valued, then $O(K) = N$ bits per component is sufficient to dichotomize the input patterns of any assignment.

IV.1.2 Examples

Example a. It is difficult to say anything useful about the general behavior of $\underline{w}$ during the training process. Therefore, simulations will be primarily used to illustrate the effects of quantization.

Suppose an optical system is built to implement the perceptron algorithm. Consider patterns with binary and bipolar components. If the recording medium were perfect, then the learning process is described by eq.(IV.1.2). Assume now that the recording medium for $\underline{w}$ has the same dynamic range as the input, i.e., 1

bit. Let

$$\beta[x] = \begin{cases} 1 & \text{if } x \geq 0; \\ -1 & \text{if } x < 0. \end{cases}$$

represent the binarizing effect of the recording medium. The updating rule for $\underline{\mathbf{w}}$ given by eq.(IV.1.1.2) becomes,

$$w_i^{(k+1)} = \beta[w_i^{(k)} \pm y_i^{(k)}], \qquad (IV.1.2.1)$$

where $w_i^{(k)}$ is the *i-th* component of $\underline{\mathbf{w}}^{(k)}$ and $y_i^{(k)}$ is the *i-th* component of $\underline{\mathbf{y}}^{(k)}$.

Digital computation was done to simulate the algorithm described by equation (IV.1.2.1). For the simulation, random binary and bipolar sequences were generated to represent the input patterns. They are arbitrarily assigned to two classes. $\underline{\mathbf{w}}$ is then trained according to eq.(IV.1.2.1), i.e., the training $\underline{\mathbf{w}}$ is binarized after each correction. As pointed out in the previous section, some of the classification may not be realizable by $\underline{\mathbf{w}}$ so trained. The simulation was programmed to stop at 10 times the number of iterations for a normal perceptron to converge (simulation performed as a control). M_s, the number of successful classifications out of a hundred trials, was recorded for each M, the number of vectors to be stored. The experiment was then repeated with the maximum number of iterations allowed set an order of magnitude higher than the previous simulation. M_s was found to be the same as those previously obtained, indicating that M_s is very close if not identical to the true number of trials that converge using this algorithm. M_s was then normalized with respect to the total number of possible dichotomies that can be achieved by a binary weight vector (obtained by testing all 2^N points). We refer to this normalized M_s as P_s, the classification strength. M is also normalized with respect to N. The curve plotting P_s versus M/N for $N = 12$ is shown in Fig.IV.1.2.1.

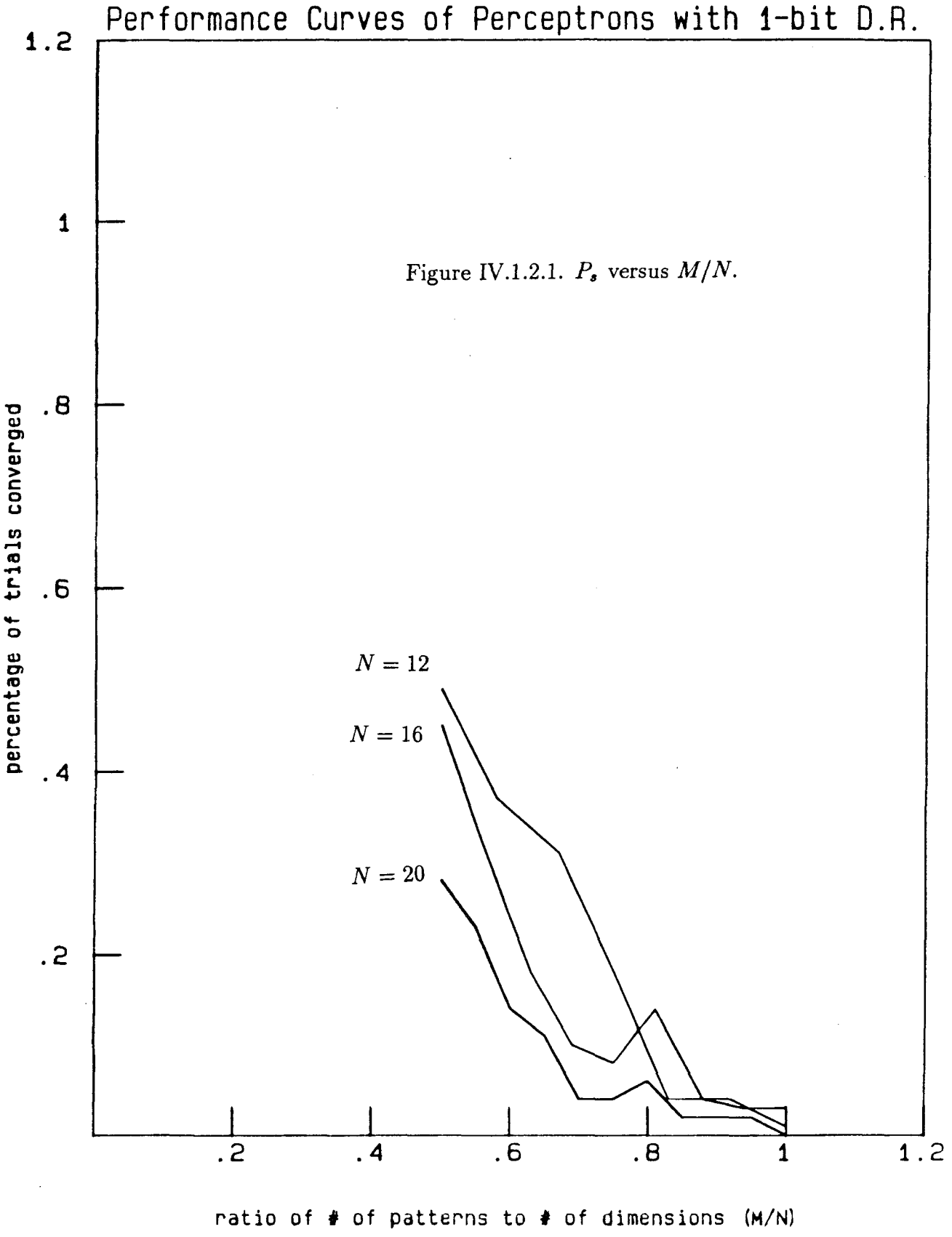


Figure IV.1.2.1. P_s versus M/N.

Simulations were stopped when M/N reached 1, because it can be shown that the capacity of a binary perceptron is upperbounded by N. (There are a total of 2^M solution regions and 2^N binary points in the N-dimensional space. Thus, if $M > N$, the average number of binary points per solution region, $N_s = 2^{(N-M)}$, approaches 0 for large enough N which implies the probability that any solution region will contain a binary point also approaches 0.) From Figure.IV.1.2.1, we observe that P_s decreases as M/N approaches 1. We should realize that the perceptron algorithm has no control of where the training solution vector lands in the solution region as long as it ends up there. (For future reference, we will refer to the training vector before quantization as $\underline{\mathrm{w}}'$.) Whether $\underline{\mathrm{w}}'$ in the solution region will be close to a binary solution point is therefore difficult to predict. As M/N approaches 1, N_s decreases to 1. The probability that $\underline{\mathrm{w}}'$ will be quantized to a binary solution point thus becomes less and less as M increases. Curves of P_s for $N = 16, 20$ are also plotted in the same figure. Note that P_s is also a decreasing function of N. This phenomenon is entirely unexpected. We have no explanation as to why this phenomenon occurs. Now N_s can be written as $2^{N(1-M/N)}$. Thus, for any fixed M/N, N_s increases exponentially with N. In other words, P_s should increase with N, which clearly contradicts what we observed.

Example b. The second case studied in this section will illustrate the effect of limited dynamic range on targeting output responses to specific values. The example chosen is the Widrow-Hoff procedure. Basically, it is a gradient descent type algorithm which solves a system of linear equations. The requirement on $\underline{\mathrm{w}}$ is such that :

$$\underline{\underline{\mathrm{Y}}}^t \underline{\mathrm{w}} = \underline{\mathrm{t}}, \qquad (IV.1.2.2)$$

where

$$\underline{\underline{Y}} = (\underline{y}^{(1)}, \underline{y}^{(2)}, \ldots, \underline{y}^{(M)})$$

and $\underline{t}$ is the target response of $\{\underline{y}^{(m)}\}$. The algorithm can be described as the following:

$$\underline{w}^{(k)} \qquad \text{arbitrary}$$

$$\underline{w}^{(k+1)} = \underline{w}^{(k)} + \rho_k(t_k - \underline{w}^{(k)\,t}\underline{y}^{(k)})\underline{y}^{(k)}, \qquad (IV.1.2.3)$$

where t_k is the target response of $\underline{y}^{(k)}$, $\underline{w}^{(k)}$ is the training vector at the k-th iteration and ρ_k is the acceleration constant which is commonly chosen to be ρ_1/k. The rule given in eq.(IV.1.2.3) can be shown to be a sequence of weight vectors that converges to a limiting vector $\underline{w}$ satisfying

$$\underline{\underline{Y}}^t(\underline{\underline{Y}}\underline{w} - \underline{t}) = \underline{0},$$

which is a necessary condition for minimizing the cost function

$$E = \|\underline{\underline{Y}}\underline{w} - \underline{t}\|^2. \qquad (IV.1.2.4.)$$

Depending on the choice of $\underline{t}$, $\underline{w}$ can be trained to achieve different tasks. One example is the Fisher's linear discriminant, which is optimum in the sense that the average inter-class separation is maximized while the average intra-class separation is minimized.

When the quantization effect of the recording medium is taken into account, the learning rule described by eq.(IV.1.2.3) becomes

$$\underline{w}^{(K+1)} = Q[\underline{w}^{(k)} + \rho_k(b_k - \underline{w}^{(k)\,t}\underline{y}^{(k)})\underline{y}^{(k)}], \qquad (IV.1.2.4)$$

where $Q[.]$ (more than 2 levels) stands for the quantization operator, i.e., the analog $\underline{\mathbf{w}}^{(k+1)}$ is translated to the nearest quantized point. Digital experiments assuming different numbers of grey levels for each component of $\underline{\mathbf{w}}^{(k)}$ were performed. The training patterns used in the simulations were sequences of binary and bipolar random numbers. The sequences were then randomly assigned to two classes with the target responses chosen to be +1 and -1, respectively. The first experiment was done assuming there were only 2 grey levels. As expected, oscillatory states with great hamming distances apart were reached within a few iterations. When these oscillatory states were used as $\underline{\mathbf{w}}$ in the LDF, the actual responses due to the training patterns were remote from their desired values. Another experiment was done assuming 16 grey levels for the weights. This time, more stable results were observed. Though oscillations still occur at a later stage, the unstable states were generally fairly close. When the oscillatory states were chosen to be the weight vectors, the actual responses differ from the target responses on average by 25 percent. As the dynamic range of $\underline{\mathbf{w}}$ was increased to 10 bits, the maximum error in output responses is confined to less than 1 percent.

IV.2 Quantization Effect on Architecture 2.

IV.2.1 Description

We see from the previous section that the effect of limited dynamic range imposes serious problems on the ability of classifying vectors and achieving an accurate target response. We now explore the possibility of executing the training procedure

with a digital computer while implementing the trained LDF optically. Since a digital computer has a much higher dynamic range, the problems that are encountered in optical training should not occur here. Nonetheless, we are not problem-free. Eventually, the trained weight vector has to be written on optical devices with limited dynamic range. There is no guarantee that after $\underline{w}$ is downloaded, all the correct responses will result when $\{\underline{y}^{(i)}\}$ are input to the optical classifier. This is because given $\underline{w}'$, the solution region itself can be shaped (convex only), located, and/or oriented in such a way that all the quantized solution vectors are farther away from $\underline{w}$ than the non-solution quantized points. To illustrate the effects of quantization, statistics on P_s will be presented. Qualitative explanations will also be given.

Example c. Suppose a perceptron is being trained by a digital computer. Here, we assume that the trained weight vector will be written on a 1-bit device. Thus, $\underline{w}$, trained by the digital computer, is generated by the updating rule given by eq.(IV.1.1.1). After it is downloaded onto the 1-bit device, the components of $\underline{w}$ become

$$b_i = \beta[w_i]. \qquad (IV.2.1.1)$$

Digital expermients simulating eq.(IV.2.1.1) were performed with M/N ranging from .5 to 1.0 for N=12,16,20. The average time for convergence before binarization of the components was short, usually a few iterations. In the digital experiment, $\{\underline{f}^{(i)}\}$ were sequences of binary, bipolar random numbers. After $\underline{b} = (\ldots, b_i, \ldots)$ was obtained, the LDF was tested with the training patterns. P_s was recorded and plotted versus M/N (Fig.IV.2.1.1).

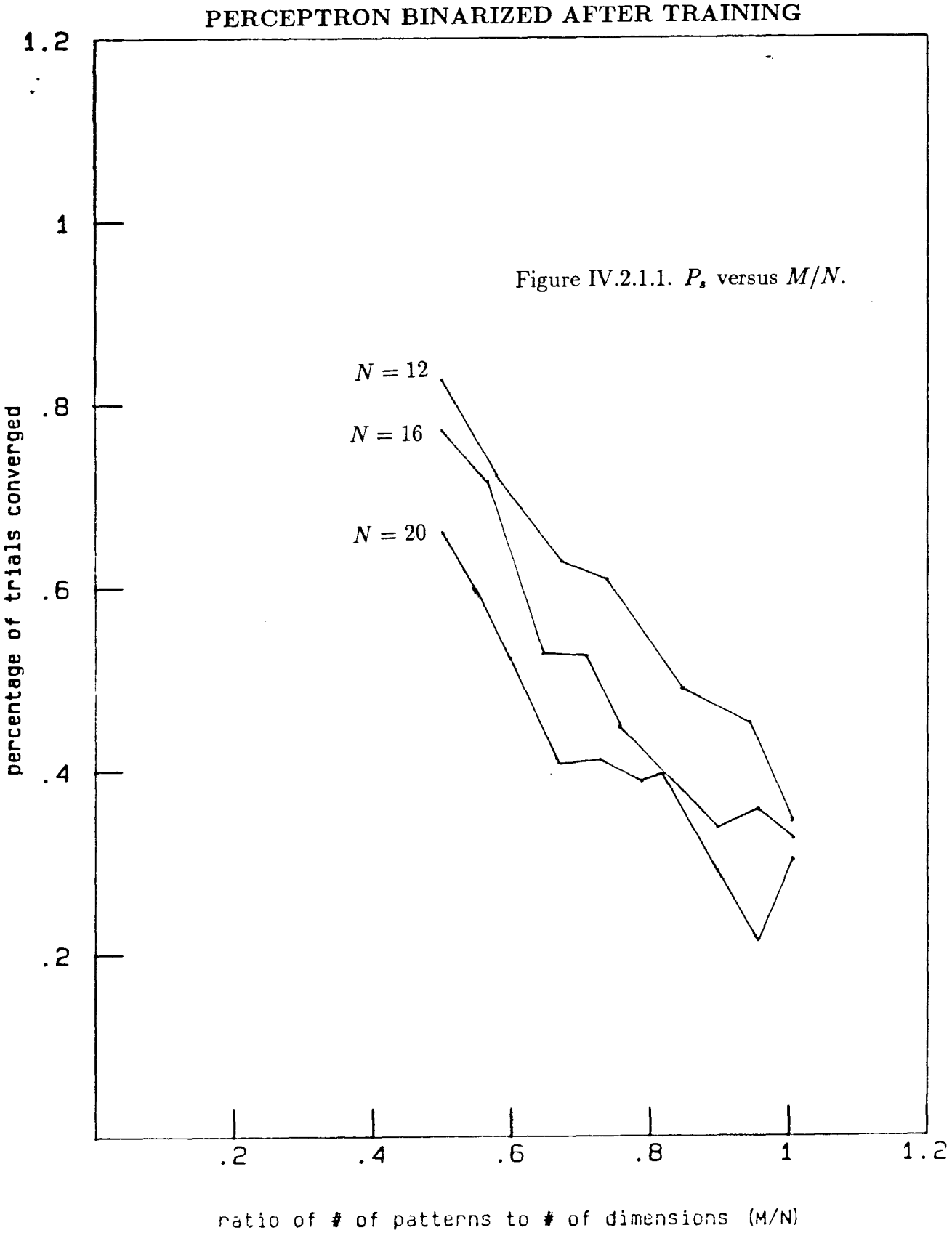


Figure IV.2.1.1. P_s versus M/N.

Observe that P_s rolls off as M/N approaches 1, and P_s decreases as N increases, exhibiting similar behavior as that shown on Fig.IV.1.2.1. Comparing Fig.IV.1.2.1. and Fig.IV.2.1.1, we observe that not only are the P_s curves in the latter case significantly higher than those of the former, they also decrease at a much lower rate as M/N increases.

Example d. The simulation in this example is based on another distorted form of the Widrow-Hoff algorithm. Similar to the algorithm in Example c, $\underline{\mathbf{w}}$ is first trained using a digital computer. The trained weight vector is then written on a dynamic range-limited device. The final form $\underline{\mathbf{w}}$ is then given as

$$\underline{\mathbf{w}} = \mathbf{Q}[\underline{\mathbf{w}}^{(\mathbf{k})}] \qquad (IV.2.1.2)$$

where $\underline{\mathbf{w}}^{(k)}$ is obtained by the Widrow-Hoff algorithm (eq.(IV.1.2.3)). The same statistics as before for the training patterns were used here. The same target responses used in Example b were also used here. After eq.(IV.2.1.2) was implemented, the LDF was tested by the training patterns. The errors of the actual responses compared to the target responses were recorded. For 2 and 16 grey-leveled devices, the average errors obtained were comparable to those in Example b. In other words, the accuracy of the output response is not improved like that of Example c.

IV.2.2 Examples

Example e. Although the training methods described in section IV.2.1 are better than those described in section IV.1.1., they are still unsatisfactory. P_s are still considered to be low, especially as M/N approaches 1. Other algorithms which

may improve P_s are explored here and in later sections. A logical assumption is that :

> if $\underline{\mathbf{w}}$ is close to the "central portion" of the solution region, then the probability P_s that quantizing $\underline{\mathbf{w}}$ will result in a binary solution vector should be high.(IV.2.1)

Thus, it is natural to derive an algorithm which will train $\underline{\mathbf{w}}$ to proceed toward the "central portion." The central portion of a solution region can be interpreted in many ways. Suppose we confine our attention to a hypersphere of radius N centered at the origin (all binary, bipolar points are on the hypersphere) in the N-dimensional space. Any solution region intersects the hypersphere at a $(N-1)$-dimensional hyperspherical surface. One way to define the central portion can then be the region close to the centroid of the $(N-1)$-dimensional hyperspherical surface. One can also consider an energy function and define the central portion to be the region close to the energy minimum. Since energy is a common notion in numerical analysis (numerous algorithms were developed to minimize energy) and there is no obvious reason why one definition of central portion is superior to the others, we will explore an algorithm involving minimization of energy.

Define the energy E of a classifier as

$$E = \sum_{i}^{M} \left(f(\underline{\mathbf{w}} \cdot \underline{\mathbf{y}}^{(\mathbf{i})}) - \mathbf{t}^{(\mathbf{i})}\right)^{\mathbf{2}}, \qquad (IV.2.2.1.)$$

where $f(.)$ is the transfer function of the output neuron and $t^{(i)}$ is the desired response of $\underline{\mathbf{y}}^{(i)}$. (Note the difference between the energy function in eq.(IV.2.2.1.) and eq.(IV.1.2.4.) is the extra transfer function $f(.)$ in eq.(IV.2.2.1.).) Again,

we will use the gradient descent method to minimize the above energy function. Mathematically, this algorithm is given as:

$$\underline{\mathbf{w}}^{(k+1)} = \underline{\mathbf{w}}^{(k)} - c\nabla_w E, \qquad (IV.2.3)$$

where c is the acceleration constant and

$$\nabla_w = \frac{\partial}{w_1}\hat{w}_1 + \ldots + \frac{\partial}{w_N}\hat{w}_N.$$

Since the derivative of $f(.)$ will be involved, the hard-thresholding function

$$f(x) = \begin{cases} 0 & \text{if } x < \text{threshold;} \\ 1 & \text{if } x \geq \text{threshold,} \end{cases}$$

can no longer be used. Instead the soft-thresholding function

$$f(x) = \frac{2}{\pi} tan^{-1}(x) \qquad (IV.2.4)$$

will be employed. Computer experiments were done to simulate this training procedure. Because of the nature of the tan^{-1} function, complete minimization, if possible, will take a very long time. A new parameter ε, hence, is introduced. Convergence is declared whenever $|f(\underline{\mathbf{w}} \cdot \underline{\mathbf{y}}^{(\mathbf{i})}) - \mathbf{t}^{(\mathbf{i})}| < \varepsilon$. ε can then be viewed as a measure of the "closeness" of the trained $\underline{\mathbf{w}}$ to the "central portion" of the solution region. Shown on Figure IV.2.2.1. are P_s versus M/N curves for $N = 12, 16, 20$ and $\varepsilon = 0.1$.

These curves, in general, follow the behaviors with increasing M/N and N that we observed earlier. For comparison, the same curves with $\varepsilon = 0.05$ were also obtained and plotted on the same figure. As we have expected, the curves with a smaller ε have a higher classifying strength. The average number of iterations for convergence versus M/N for each of the same N's and ε's were plotted

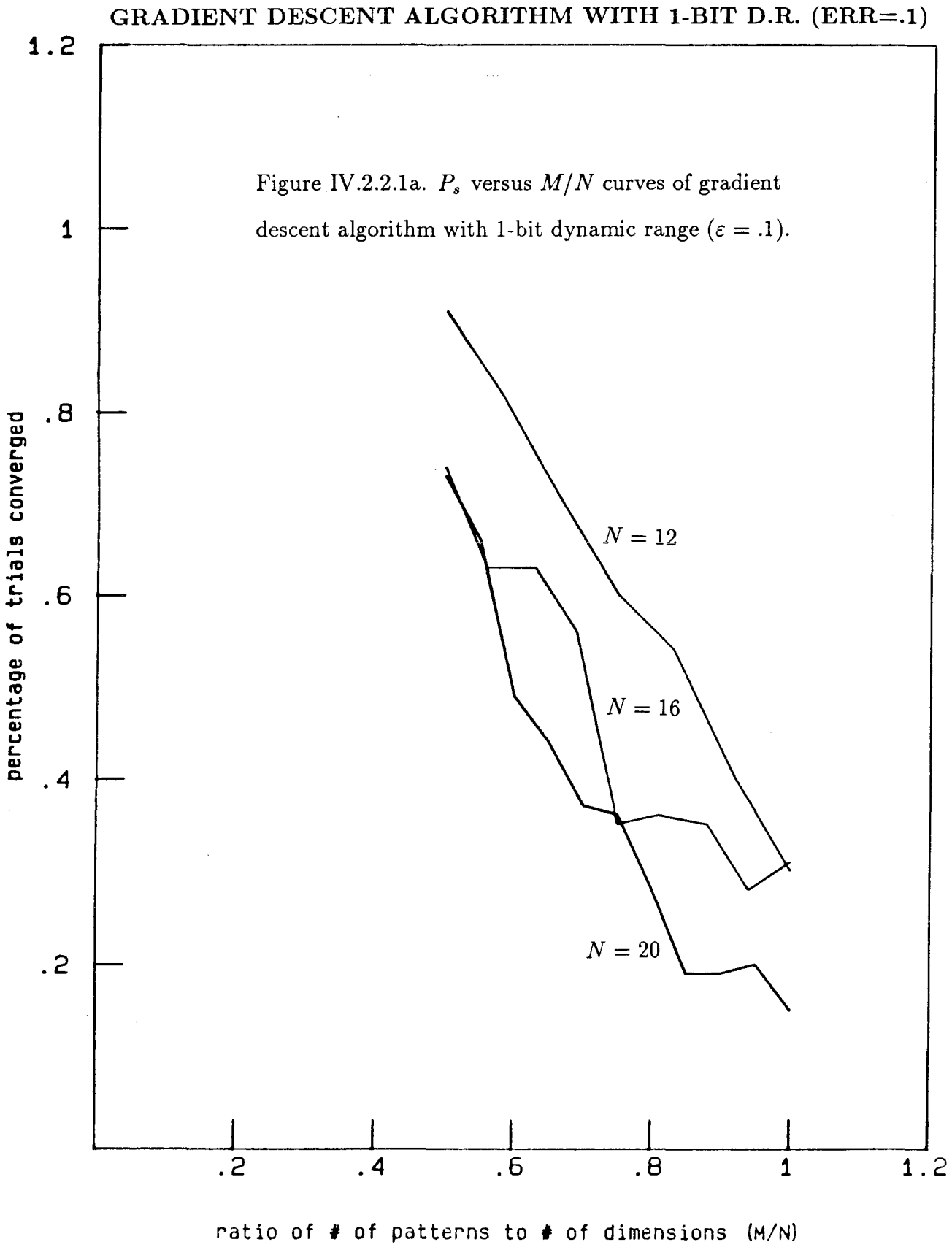


Figure IV.2.2.1a. P_s versus M/N curves of gradient descent algorithm with 1-bit dynamic range ($\varepsilon = .1$).

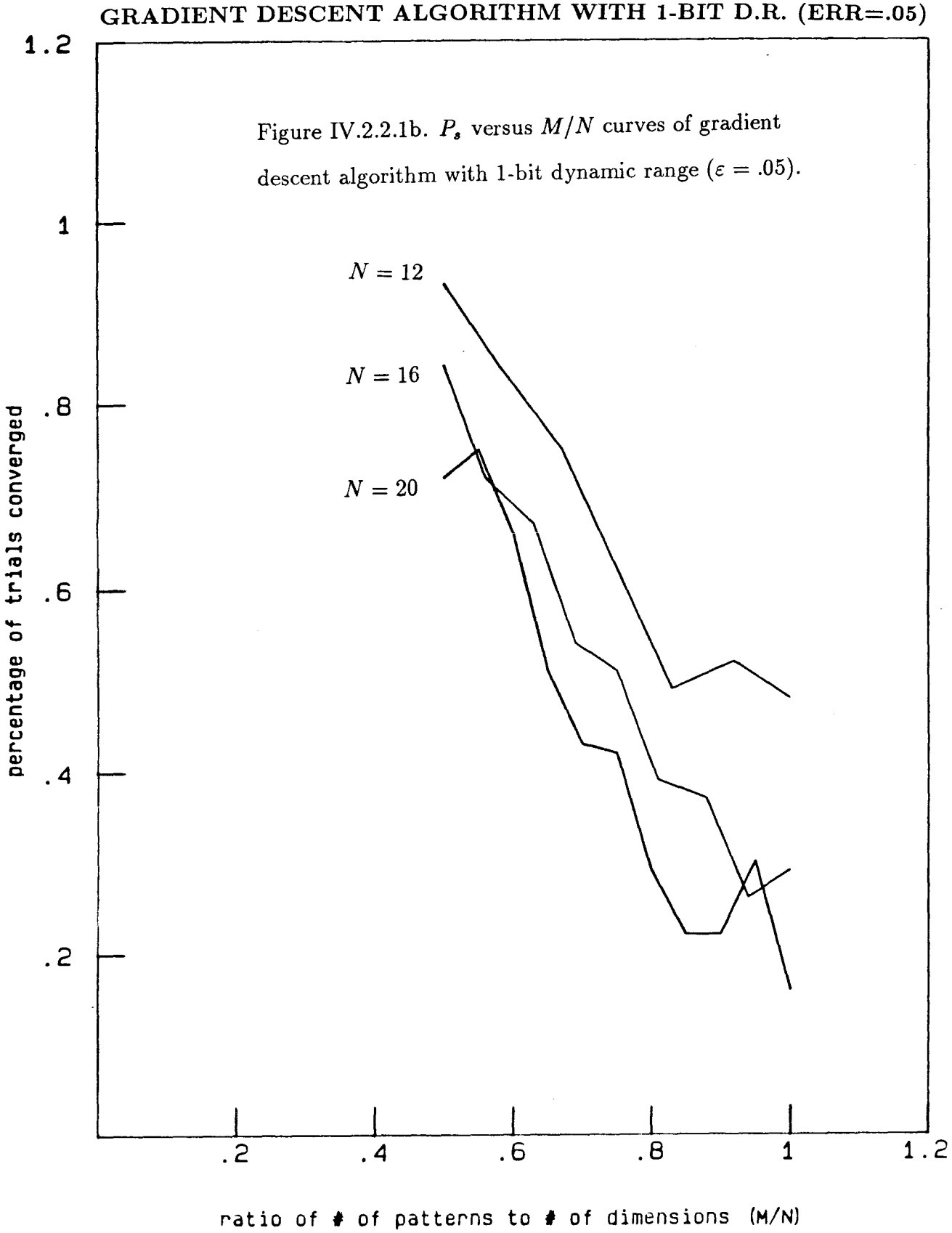


Figure IV.2.2.1b. P_s versus M/N curves of gradient descent algorithm with 1-bit dynamic range ($\epsilon = .05$).

on Figure.IV.2.2.2. Even though the average convergence time in the simulations presented here are orders of magnitude higher, P_s is only slightly higher than that of Example c. Consideration of the average convergence time versus the classifying strength shows that the usage of gradient descent procedure in classification oriented problems is not very effective when the trained $\underline{w}$ is quantized. Last, but not least, is that the results presented here once again illustrate the relatively random relationship between the existence of a real valued solution weight vector and a quantized solution vector.

IV.3 Quantization Effect on Architecture 3.

IV.3.1 Examples

Example f. The training algorithms that we have considered so far either do not have the capability (dynamic range) to accommodate training, or they totally ignore the quantization effect of the medium on which the trained $\underline{w}$ is to be stored. In this example, we consider a procedure which is implemented on a digital computer while the quantization effect is taken into account in the training process. This procedure is modified from the perceptron training algorithm. It can be divided into two parts:

i. the training part. The correction part is identical to that of the perceptron algorithm:

$$\underline{w}^{(k+1)} = \underline{w}^{(k)} + \underline{y}^{(k)} \qquad (IV.3.1.1)$$

ii. the testing part. Correction is only needed if the binarized version of the training weight vector, $\underline{b}$, does not provide the proper response when it is subjected to $\underline{y}^{(k)}$:

$$\underline{b}^{(k)} = \beta[\underline{w}^{(k)}]; \qquad (IV.3.1.2)$$

correction is needed whenever $\underline{b}^{(k)} \cdot \underline{y}^{(k)}$ is negative.

Computer simulations with $N = 12, 16, 20$ were performed according to this algorithm. Statistics of the training patterns were generated as before. The results are presented in Figure IV.3.1.1.

Two sets of graphs are presented. One set corresponds to 10 iterations (i.e., computer program stopped after 10 iterations whether convergence occurred or not) while the other set corresponds to 100 iterations. The general behaviors of P_s with respect to increasing M/N and N is observed here again. Furthermore, P_s also increases with respect to increasing the maximum number of iterations allowed, which is also expected. Compared to the result obtained in Example c, we observe the increase in P_s here is traded off for longer average training time (Figure IV.3.1.2abc.). Specifically, when the allowed training time of (f, 10 iterations) is only slightly longer than the average convergence time of (c), P_s of the former is also only slightly longer than the latter. However, when the allowed training time of (f, 100 iterations) is an order of magnitude longer than the average convergence time of (c), P_s of (f) is a little less than 2 times that of (c). Note that the same kind of trade-off is also visible in Example e. But the trade-off is less severe in Example f. Consideration of both P_s and average training time indicates that training methods used in (c) and (f) are most promising.

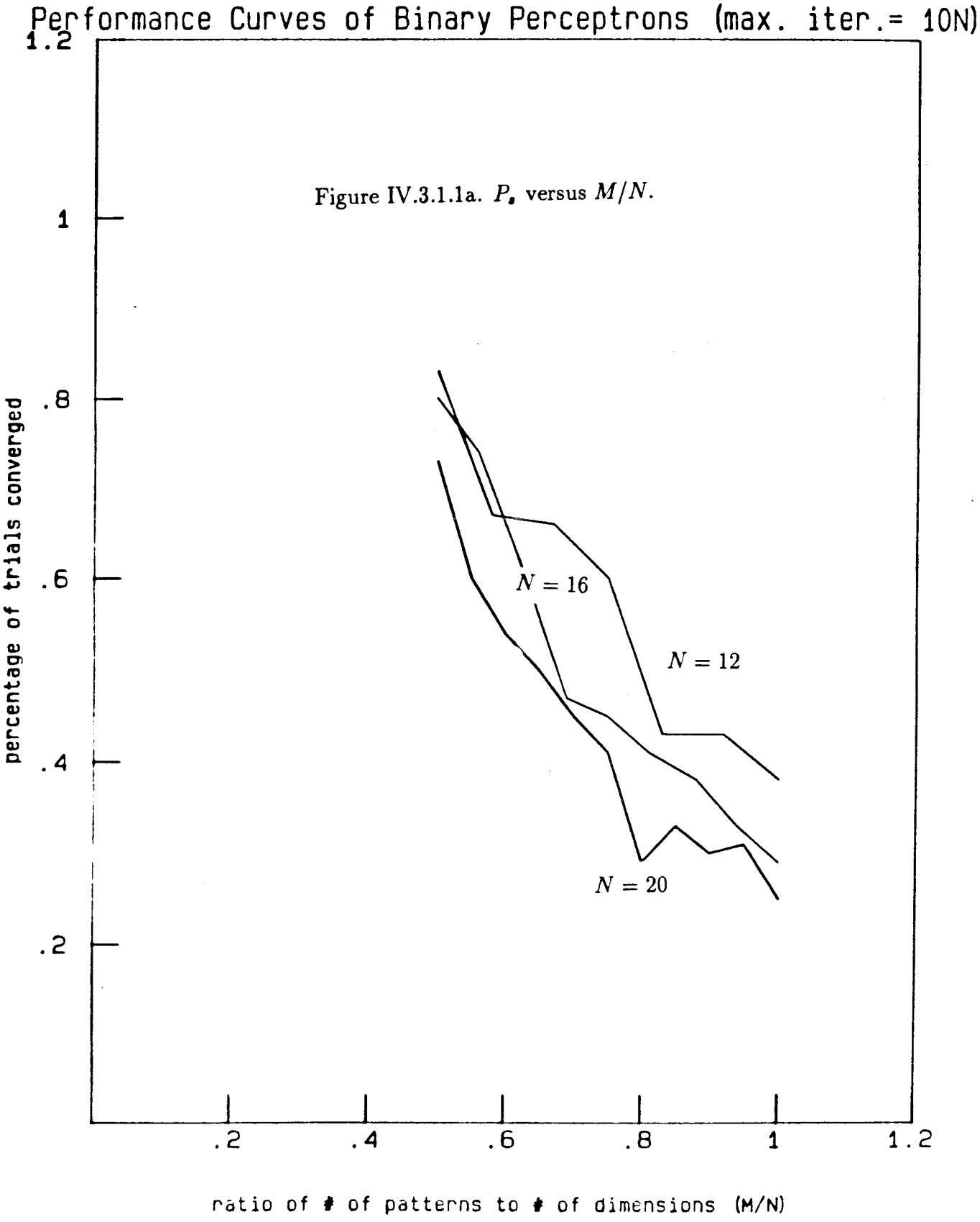


Figure IV.3.1.1a. P_s versus M/N.

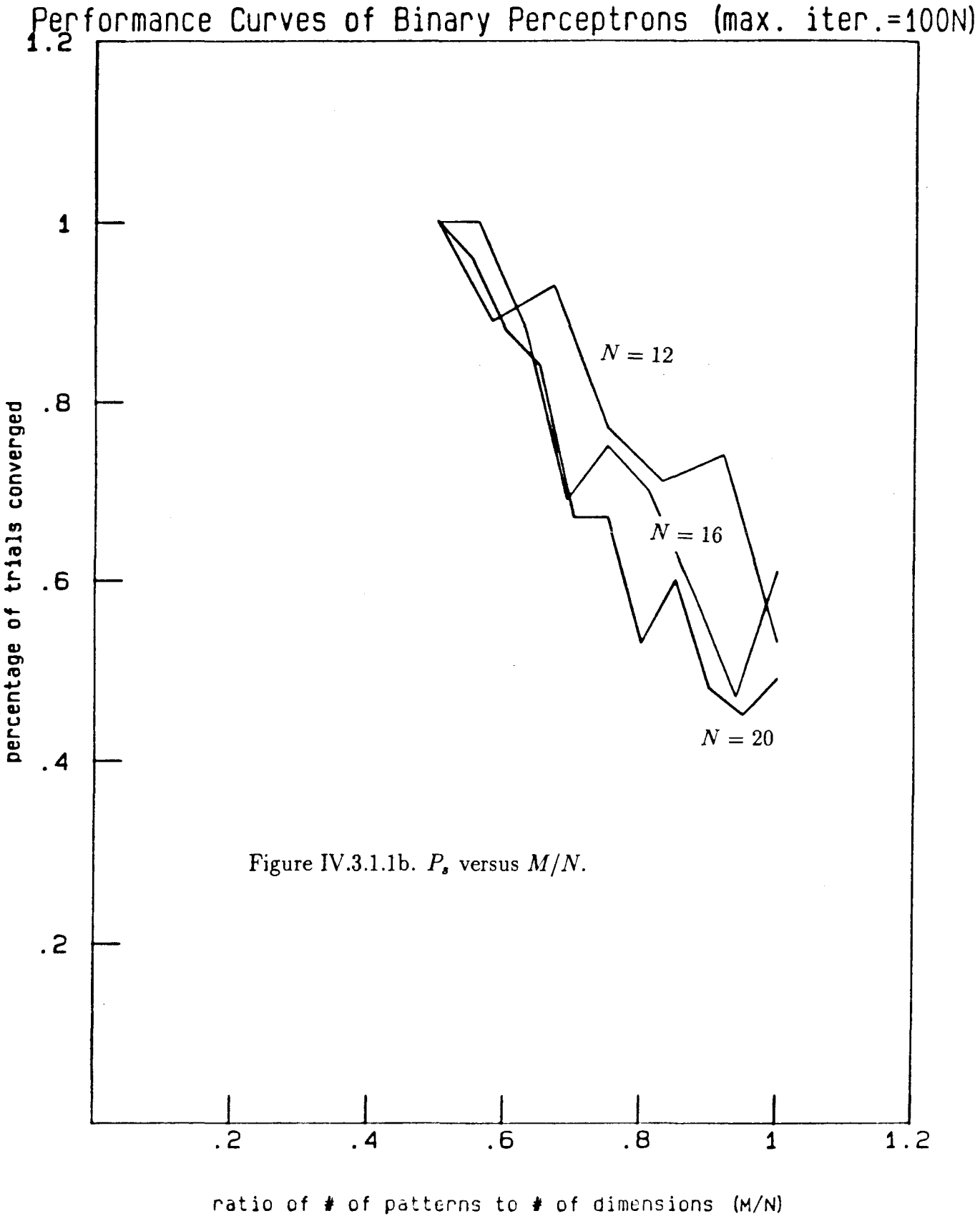
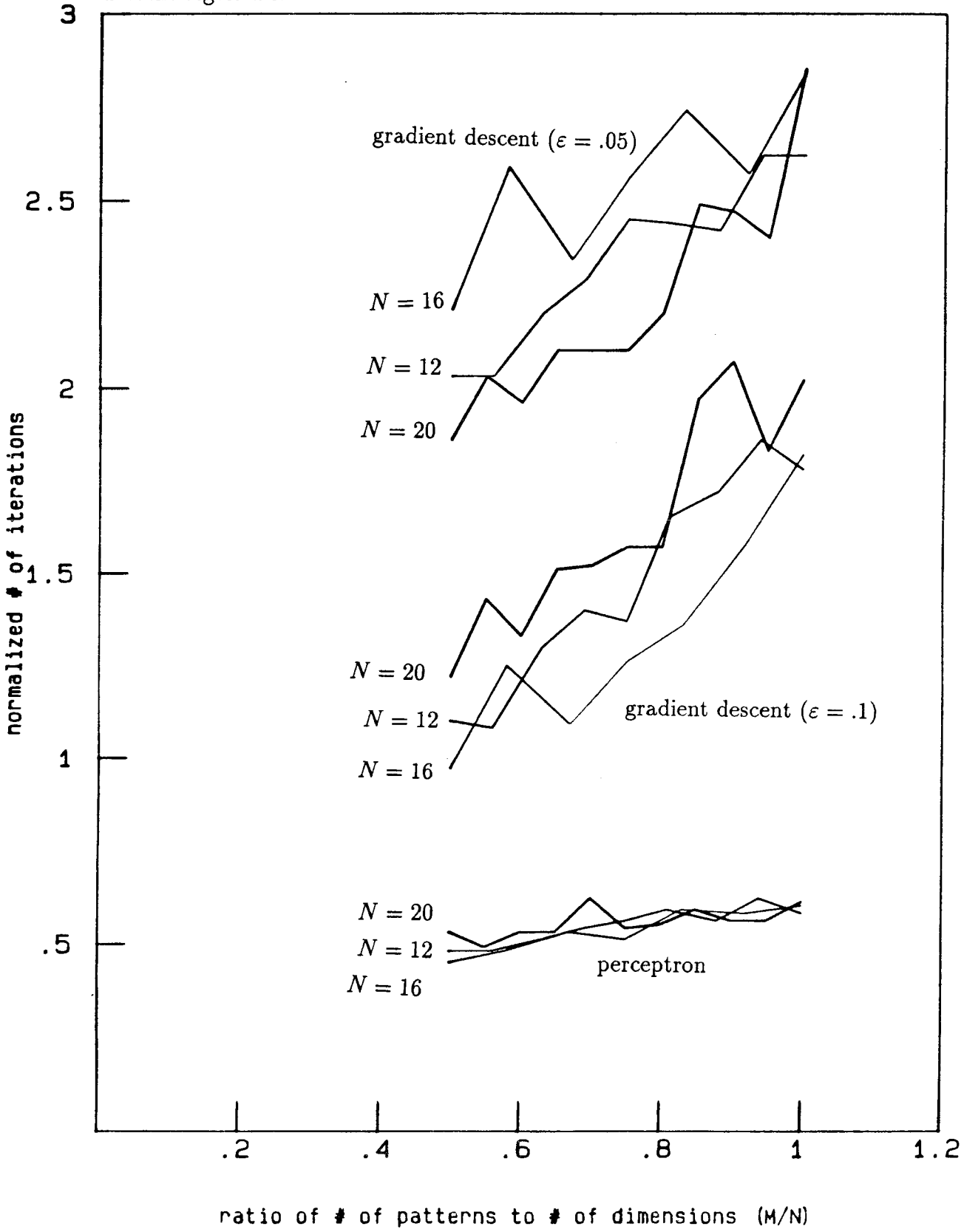


Figure IV.3.1.1b. P_s versus M/N.

Figure IV.3.1.2a. Average number of iterations for convergence of the gradient descent algorithm.



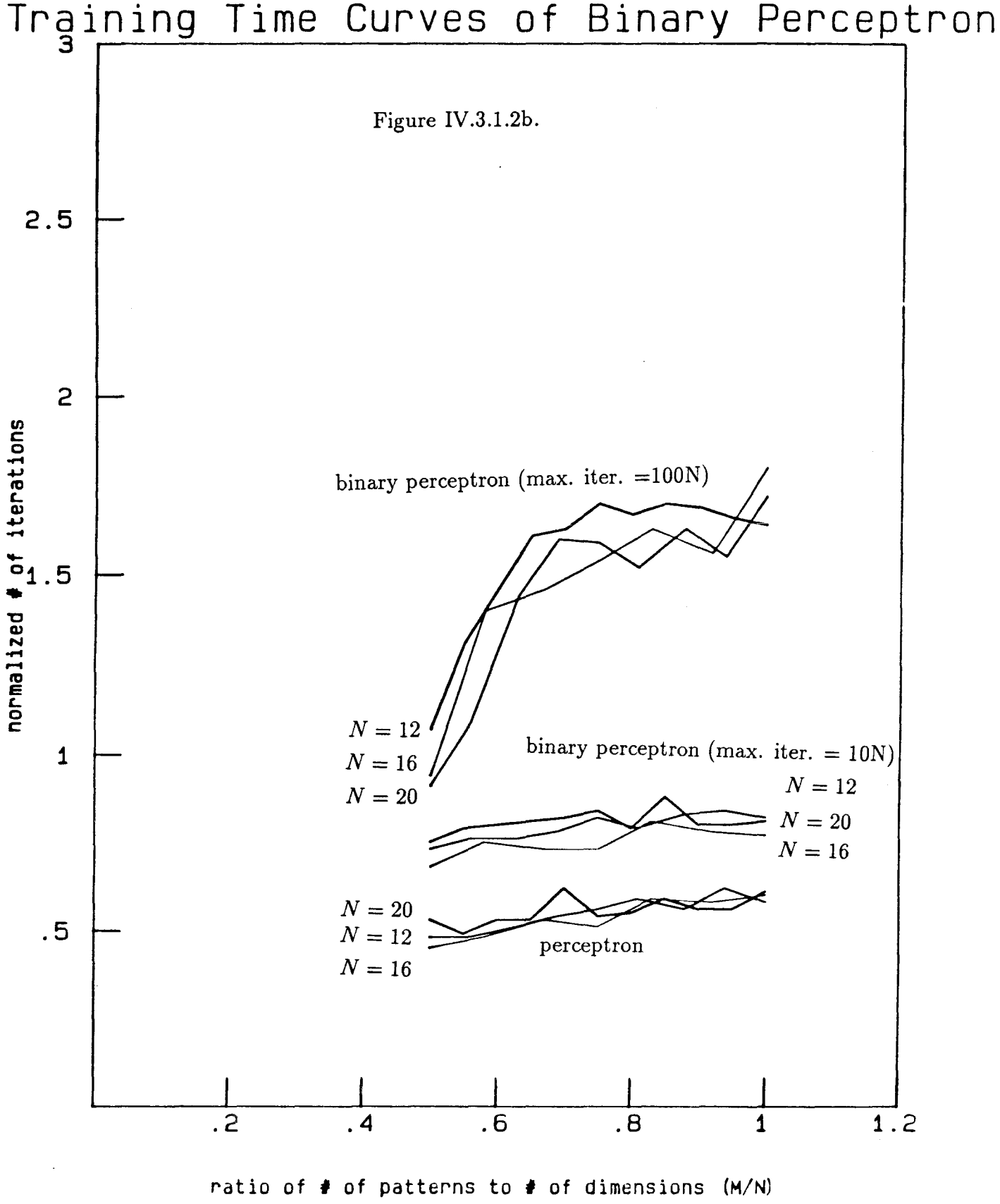


Figure IV.3.1.2b.

Example g. The search for a binary solution vector is difficult. The problem is that it is difficult to design a training algorithm to achieve the task without detailed knowledge of the geometry of the solution region. But having such knowledge is intimately connected to knowing where a binary solution point is. We therefore have a cyclic quest: we cannot find something because we do not know what that thing is. Thus, in order to guarantee that a binary solution vector be found, the best way may be some kind of exhaustive search. On the other hand, we definitely have no interest of searching all the 2^N points. A desirable initial state should therefore be reached before the search begins. A good choice for the initial state is an analog solution vector, since it can be easily obtained by the perceptron algorithm.

Computer simulations were performed to test the usefulness of an exhaustive search. Statistics for the training patterns were identical to what we used throughout this chapter. An analog solution vector was first obtained by use of the perceptron algorithm. This initial vector was then binarized and tested by the training patterns. If misclassification occurred, then the first bit of the initial binary vector was flipped. The resultant binary vector was then tested by the training patterns. If misclassification still occurred, the first bit was flipped back to its original value. The second bit was then flipped. The procedure was then repeated until all bits had been flipped or no misclassification happened. If a binary solution vector still had not been found, the procedure was repeated with two bits flipped and so on. The experiment was done with 1 and 2 bits flipped. In either case, the number of successful trials were recorded. The results are shown in Figure IV.3.1.3.

Observe that the results obtained with 2 bits flipped is better than those with only 1-bit flipped. While the P_s curves still roll off as M/N increases when only 1-

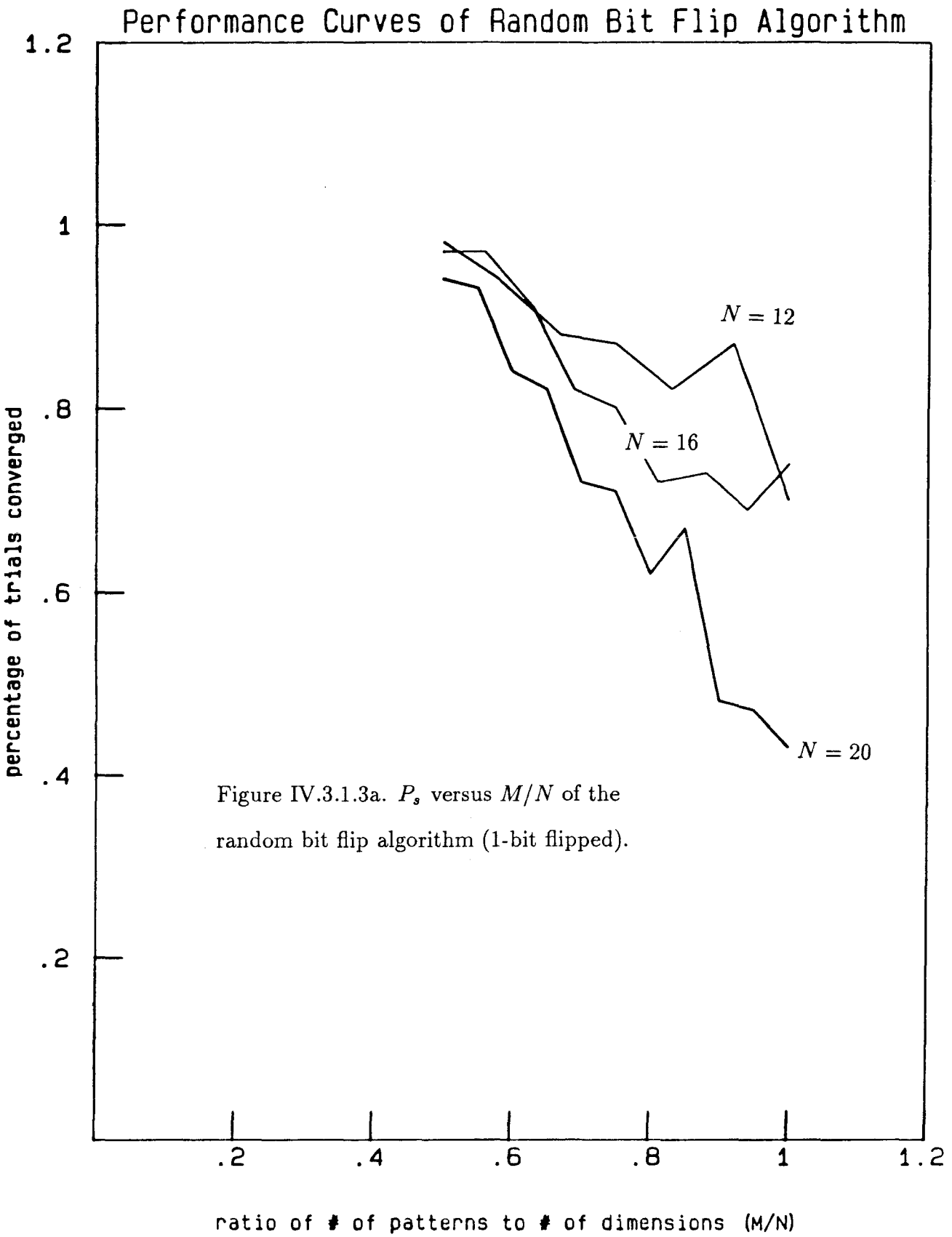


Figure IV.3.1.3a. P_s versus M/N of the random bit flip algorithm (1-bit flipped).

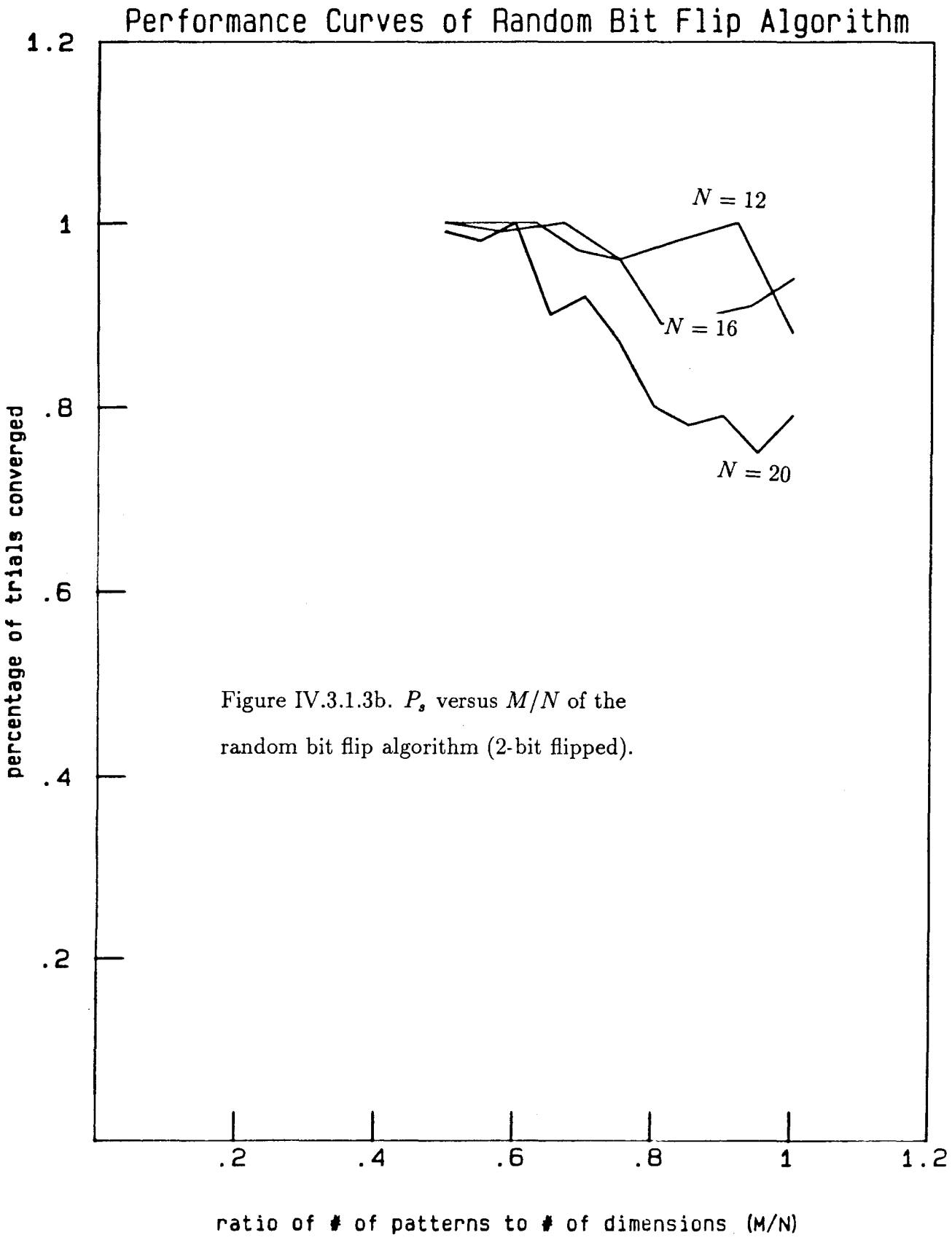


Figure IV.3.1.3b. P_s versus M/N of the random bit flip algorithm (2-bit flipped).

bit was flipped, the same P_s curves remain relatively flat when 2 bits were flipped. This phenomenon is only true for $N = 12$ and 16, which signifies that a binary solution vector on the average is not far away from the initial analog vector (in hamming distance). But as N increases, e.g., 20, so does the average hamming distance between a binary solution vector and the initial analog vector. This means more bits had to be flipped in order to obtain a P_s curve with less gradient. As a whole, the P_s curves obtained in this example are the highest among all that are obtained in this chapter. It is important to note that this algorithm does not place an upper bound other than N on the number of bits that have to be flipped in order to arrive at a binary solution vector. Examples can be constructed to illustrate this point. Refer to Figure. IV.3.1.4.

This is a conformal mapping of a hypersphere of $N = 3$ onto a 2-dimensional plane. Since the binary points are equidistant from the origin, they all fit on the 3-dimensional sphere and are denoted by small circles. The longitude and latitude represent the intersections of the sphere and the planes $x_1 = 0; x_2 = 0; x_3 = 0$. The solution region is depicted by the polygon in the middle of the graph. The edges of the solution region are not straight because they are conformally mapped (compare with shortest distance lines between two points on a world atlas). As can be seen on the graph, the only binary solution vector is at the right-most corner of the polygon. If the initial vector is at the point indicated by the X, then the bits needed to be flipped in order to arrive at the binary solution vector of 3. This means a complete exhaustive search has to be done.

IV.4. Conclusion.

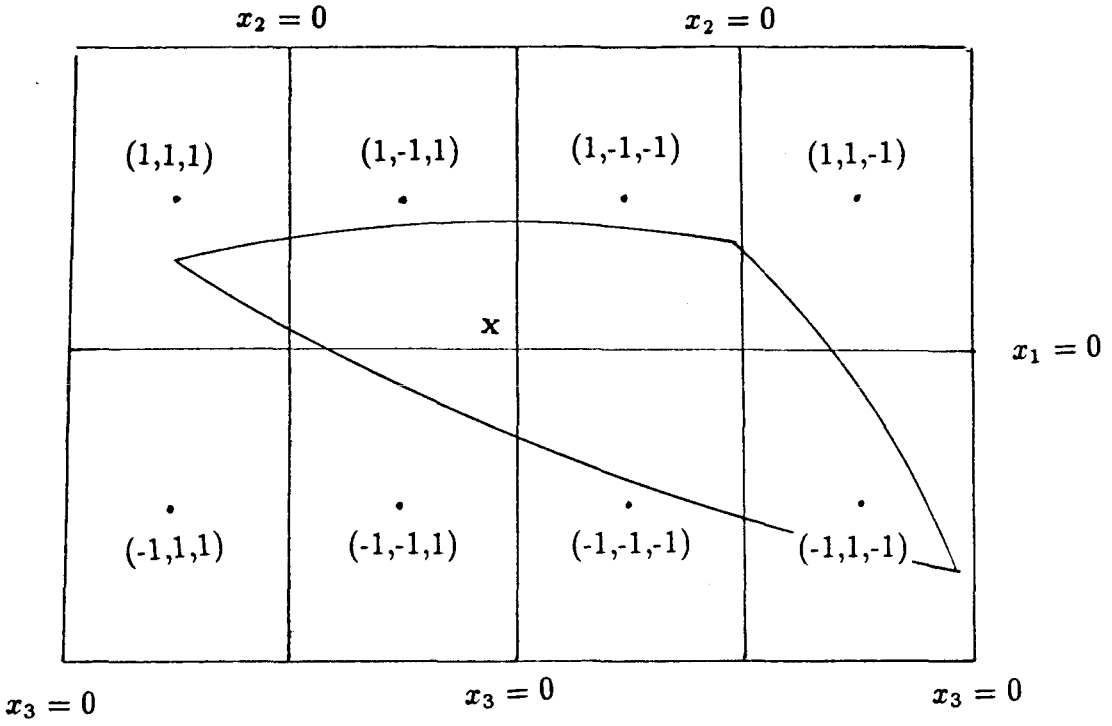


Figure IV.3.1.4. Point x is the initial $\underline{w}$. Since x is in the octant opposite to the one which conatains (-1,+1,-1), the only solution point, all the bits have to be flipped in order to get to (-1,+1,-1).

In this chapter, we analyzed different algorithms subjected to the quantization effect of a realistic device on which the weight vector is to be trained or written. We have come to the conclusion that devices with very limited dynamic range should not be used to implement complicated training algorithms. However, those devices are still suitable for the exhaustive search described in Example g. It is because no information is lost in the updating of $\underline{w}$. As a matter of fact there is no information, aside from the output response of right or wrong, available to the updating rule in this kind of procedure. Observations about the descending nature of P_s as M/N and N increase were also made. Although the training patterns upon which the simulations were run were generated from gaussian random number generators, we believe that the descending property of P_s should remain valid for other input statistics for the algorithms considered. The reason for this is the probabilistic nature of quantizing a vector which is intrinsic to all problems. Finally, it is important to note that the results obtained in this chapter should provide a useful lower bound for the performance of linear machines with higher but still limited dynamic range.

References for Chapter IV

1. S. Gass, *Linear Programming*, Third Edition, McGraw-Hill, New York (1969).
2. M. Minsky and S. Papert, *Perceptrons: An Introduction to Computational Geometry*, MIT Press, Cambridge, Mass. (1969).
3. J. Lee, *Numerical Analysis for Computers,* Reinhold Publishing Corp., New York (1966).
4. O. Veblen and J. Whitehead, *The Foundations of Differential Geometry,* Cambridge University Press, London (1932).
5. H. Manning, *Geometry of Four Dimensions*, The MacMillan Co., New York (1914).
6. D. Casasent, "Unified Synthetic Discriminant Function: Computational Formulation," *Appl. Opt.*, **23**, 1620 (1984).
7. F. Rossenblatt, *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*, Spartan Books, Washington, D.C. (1961).
8. T. Cover, "Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition," *IEEE Trans. Elec. Comp.,* **EC-14**, 326 (1965).
9. B. Widrow and M. Hoff, "Adaptive Switching Circuits," *1960 IRE WESCON Conv. Rev.,* Part 4, 96-104 (August 1960).
10. R. Duda and P. Hart, *Pattern Classification and Scene Analysis,* John Wiley and Sons, New York (1973).
11. N. Nilsson, *Learning Machines: Foundations of Trainable Pattern-Classifying Systems*, McGraw-Hill, New York (1965).
12. A. Oppenheim and R. Schafer, *Digital Signal Processing*, Prentice-Hall, New

Jersey (1975).

13. W. Feller, *An Introduction to Probability Theory and Its Applications*, Wiley and Sons, New York (1970).

14. D. Rumelhart, J. McClelland, and the PDP Research Group, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Vol.1: Foundations*, The MIT Press, Massachusetts (1986).

V. Implementation of Optical Correlators

V.0 Introduction

Recall that the image correlation $g(x,y)$ between two patterns $f(x,y)$ and $h(x,y)$ is defined as

$$g(x,y) = \int\int f(\hat{x}+x, \hat{y}+y)h^*(\hat{x},\hat{y})d\hat{x}d\hat{y}.$$

In the Fourier domain,

$$G(u,v) = F(u,v)H^*(u,v)$$

where

$$G(u,v) = \int\int g(x,y)e^{-j2\pi(ux+vy)}\ dxdy$$

$$F(u,v) = \int\int f(x,y)e^{-j2\pi(ux+vy)}\ dxdy$$

are the Fourier transforms of $f(x,y)$ and $g(x,y)$, and

$$H^*(u,v) = \left[\int\int h(x,y)e^{-j2\pi(ux+vy)}\ dxdy\right]^*$$

is the complex conjugate of the Fourier transform of $h(x,y)$. Due to the Fourier transforming property of a lens, correlation between patterns with high space-bandwidth product can be implemented by optics in a simple way. A schematic diagram of the classic Vander Lugt correlator is shown in Figure I.1. In this architecture, $f(x,y)$ on plane P_1 is Fourier transformed by lens L_1. $F(u,v)$ is then multiplied by $H^*(u,v)$ which is recorded on plane P_2. The optical product is finally inverse Fourier transformed by lens L_2 to produce $g(x,y)$ at the output plane P_3.

The concept of matched filtering was first brought up in the field of communication: suppose a signal, contaminated by additive gaussian white noise, is to be detected by a matched filter. It can be shown, by invoking the Schwarz inequality, that the ratio of signal energy to average noise energy is maximized by the matched filter. In the field of pattern recognition, however, the interpretation of matched filtering is slightly different. Suppose that a Fourier transform hologram of $f(x,y)$ is synthesized and placed at P_2. The amplitude transmittance function of the hologram should contain a term which is proportional to $F^*(u,v)$. Consider the the field distribution transmitted by the hologram when the signal f is presented to P_1. Incident on the hologram is a field distribution proportional to F, and transmitted by the hologram is a field distribution proportional to $\|F\|^2$. This latter quantity is real, which means that the hologram cancels all the phase information of F. Thus the transmitted field distribution is a nonuniform plane wave. L_2 then focuses this plane wave to a bright spot at P_3. When an input signal other than f is present, the phase information will in general not be matched by the hologram and the transmitted field distribution will not be brought to a bright focus by L_2. Thus the presence of the signal f can be detected by measuring the intensity of light at the focal point of L_2.

To realize a pattern classifier by implementing the Vander Lugt correlator, the filter matched to the pattern to be detected has to be available. This can be done by using a photograph of the pattern to record the filter. Unfortunately, this method of implementing the reference filter does not have a great deal of practical value. It is because few realistic problems can be solved with only one stored pattern. While most problems require classes of dissimilar objects to be identified, it is im-

portant that a large number of reference filters be available and be presented at P_2 with convenience. The availability of the reference filters is a definite requirement and cannot be avoided. Since the patterns to be detected must be known prior to detection, the Fourier transforms of the patterns can be computed and recorded. However, recording the matched filters on photographic media is considered undesirable. For it means a mechanical device has to be involved in order for the filters to be changed. This is not only a question of elegance but also one of practicality. The reason is that in order for the correlator to work properly, the origin of the matched filters has to be aligned with the rest of the optical system perfectly. The problem of changing the filters without disrupting the optical system can be solved if a spatial light modulator (SLM) is used. Basically, the function of a SLM is similar to a photographic plate: light transmitting through (or reflecting from) a SLM is modulated according to the information recorded. The major difference, which makes a SLM more preferable, is that the information recorded can be modified by some external means, e.g., electronics control. (We shall call this property programmability.) Therefore, by placing a SLM at P_2, we have an optical system that can remain aligned as we change filters. The filters, of course, are stored in some memory bank interfaced to the SLM which can be accessed with ease. This arrangement provides the additional advantage that complicated filters, other than matched filters, designed by a digital computer can be programmed onto the SLM without much difficulty.

To make the optical correlator really attractive, another SLM should be placed at P_1 to allow real-time update of the input pattern. This SLM should be interfaced with some input device, a video camera for example. In addition, the input SLM

can also be interfaced to some processors, should image preprocessing be desired. Finally, an image detector should be placed at P_3. This image detector should be capable of recording the correlation output in real time. Not only will this detector allow remote monitoring of the output plane, by interfacing the detector with other processors, additional processing of the correlation output can be carried out.

Summing up all the possibilities provided by the input SLM, reference SLM, and the output detector, we have:

i. real-time input;

ii. preprocessing of input patterns;

iii. real-time filters update;

iv. convenience for complicated, designed filters to be programmed;

v. remote monitoring and recording of correlation outputs; and

vi. postprocessing of correlation outputs. We shall call the correlators satisfying these requirements real-time programmable correlators.

Given the input SLM, the reference SLM, and the image detector, a real-time programmable correlator can be built. The extent of success of the correlator, however, depends on the quality of these devices. The following is a list of criteria used to qualify a SLM or a detector:

i. space-bandwidth product, the number of pixels available to represent a pattern. Besides having the ability to represent complicated real life image, pattern recognition, and information theories state that the higher the space-bandwidth product, the higher will be the capability of the filter and the signal-to-noise ratio of the output;

ii. optical efficiency, the amount of useful optical power;

iii. speed, how fast can the information be updated;

iv. dynamic range, the number of grey levels available;

v. the linearity of the transfer function of the device (only applicable if the device is not binary);

vi. optical flatness of the device, important becaue the Vander Lugt correlator is a coherent system. Even if the optical flatness can be improved by other means, a hologram to correct the distortion, for example, it usually means more components are required;

vii. resolution, the size of each pixel. If the pixel size of one SLM is large, then either the pixel size of the other SLM has to be small or extra optical components have to be used to match the sizes of the pixels;

viii. modulation depth, a measure of maximum modulation compared to no modulation. A modulation depth of 1 is perfect whereas a modulation depth of 0 is useless;

ix. uniformity, the respective performances of different pixels under the same excitation;

x. the ability to modulate or process bipolar signals;

xi. cost;

xii. life-time of the recorded signal, the longer the more desirable;

xiii. the size of the device;

xiv. the ease of interfacing with external circuit, i.e., the amount of electronics required;

xv. the change in behavior due to change in environment or the device itself

(it is possible that the heat generated by the device when operating at high speed may change the transfer function of the pixels);

xvi. power requirement;

xvii. the immunity to noise, optical or electronic;

xviii. optical frequency response, determines the useable source (usually visible spectrum is more desirable).

Since the invention of television, the technology of image detectors has advanced to a state that most detectors have proven themselves according to the above criteria. On the other hand, the SLM technology, due to its short history, is still relatively immature. In recent years, several SLMs have been developed. The state of the art of these devices, however, remains a major obstacle limiting the accuracy, processing power, and practicality of optical image correlators.

In this chapter, we investigate the possibility of using some of these devices to implement optical image correlators. The principles of operation of these devices will be explained. Their potential as SLMs will be addressed. Their advantages and limitations will also be commented upon. The optical characteristics of these devices will be analyzed and demonstrated. Experimental results will be presented. Computer analyses and experimental results using these devices, at the input plane and reference plane, to implement correlators will also be presented. A 2-D optical correlation method, in which the 2-D processing space is constructed by one spatial dimension and time, will be introduced. The use of only one spatial dimension, with the Time-and Space-Integrating (TSI) method, allows 1-D spatial light modulators and 1-D time correlators to be incorporated in the synthesis of the 2-D processor.

Specifically, the acousto-optic device will be used as a SLM, and a CCD detector array will be used as an array of optically addressed 1-D correlators. Advantages and limitations of this architecture will be addressed. Experimental results of the optical implementation of this architecture will be presented. In addition, rotation invariant filters will be addressed. Computer and optical experimental results will be presented.

V.1 Magneto-Optic Device as a SLM

V.1.1 Description of the Device

The Litton magneto-optic device (MOD) consists of a two-dimensional array of magneto-optic modulators that are fabricated monolithically on the same non-magnetic substrate. (The one that is used in the following experiment has 128X128 elements.) Each element of the array can be individually addressed electronically through an array of crossed electrodes. The structure of the MOD is shown schematically in Figure V.1.1.1.

When current is applied to a pair of crossed electrodes, a sufficiently high magnetic field is produced at the location where the two electrodes cross so that the magnetization of the pixel at that location is aligned with the applied magnetic field. Only one of the four pixels that surround the position where each pair of electrodes cross is affected because the lowerright corner in each pixel is ion-implanted and this reduces the strength of the applied magnetic field that is required to initiate the magnetization switching. Each pixel of the array can be magnetized in a direction

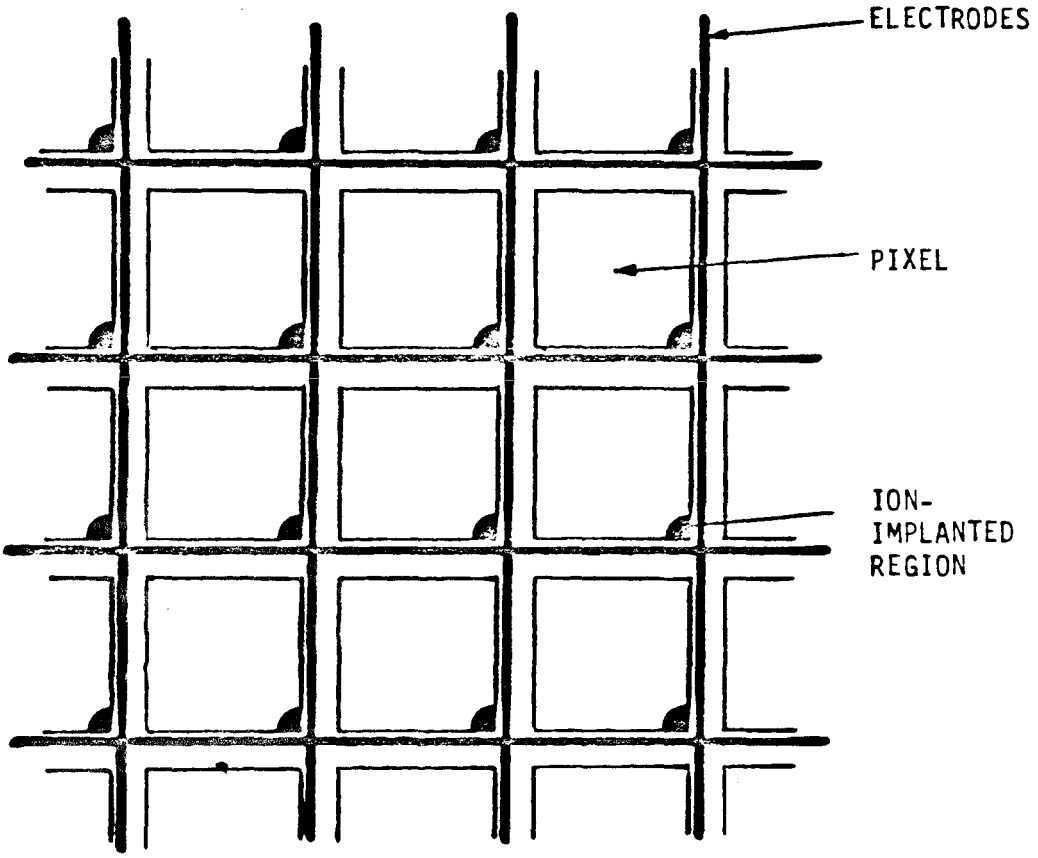


Figure V.1.1.1. The schematic structure of the MOD.

perpendicular to the surface in either polarity. Two pixels of the MOD, magnetized in opposite polarities, are shown in Figure V.1.1.2.

The pixels are illuminated with a plane wave polarized in the y direction. The propagation of light through the magnetized medium results in a rotation of the plane of polarization by an angle θ about the z-axis due to the Faraday effect. The direction of the Faraday rotation is opposite for the two stable magnetization states, as shown in Figure V.1.1.2. The light that is transmitted through the MOD is normally viewed through an analyzer that is oriented to block one of the rotated polarizations. A portion of the incident light is transmitted through pixels that are magnetized in the opposite direction, and thus an image of the binary pattern that is stored in the MOD is observed. This method for observing the stored image provides maximum contrast and is therefore optimum for display applications. In information processing, however, we are interested in the information content. If we examine the x and y components of the two rotated polarizations in Figure V.1.1.2, we realize that the y component (parallel to the incident polarization) is not affected by the magnetization state of the pixel. Thus, the portion of the transmitted light that is polarized in the y direction is unmodulated and acts as a uniform background bias. The direction (or, equivalently, the polarity) of the orthogonal, or x, component is determined by the polarity of the magnetization of the pixel. Therefore, the information that is recorded on the MOD is transferred to the x polarization only. Bipolar modulation of the light amplitude can be obtained if the MOD is used in conjunction with an analyzer oriented to transmit the x polarization. Bipolar modulation of the light amplitude can be obtained if the MOD is used in conjunction with an analyzer oriented to transmit the x polarization. Bipolar modulation is

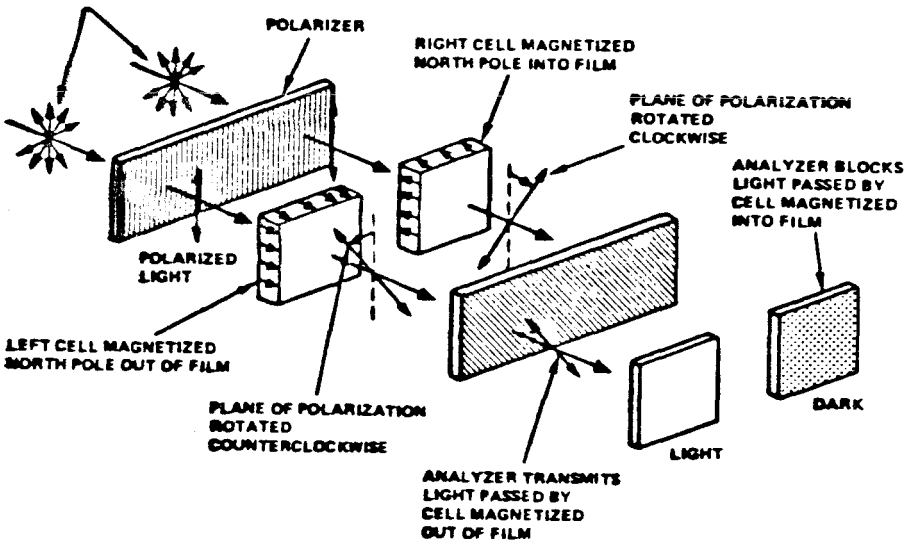


Figure V.1.1.2. Two pixels of the MOD showing the on and off states.

generally desirable in optical processing systems since it eliminates the need for the bias that is typically used to represent bipolar signals.

Listed below are the advantages and limitations of the LIGHT-MOD.

Advantages:

i. cost can be very low if mass produced because its fabrication is based on standard epitaxial fabrication techniques.

ii. random-accessibility of individual pixels.

iii. pixel switching is logic driven, therefore the modulators are relatively immune to electronic noise.

iv. compact (6"X6"X2.5").

v. can be interfaced to micro-computer easily.

vi. optically flat.

v. high monochromatic light contrast (1000:1).

vi. high speed (~300 frames per second for a 128X128 device).

v. parallel addressability of pixels of the same row or column.

vi. recorded information is nonvolatile, so no refreshing is required.

vii. good uniformity (because of the semi-conductor technology).

viii. can modulate bipolar signal.

Limitations:

i. 1 bit-dynamic range.

ii. low optical efficiency (~4 percent, this figure includes blocking by the output analyzer).

iii. space-bandwidth product not high enough (128X128).

iv. pixel size, ($p_f \sim 100\mu m$), not small enough. If a MOD is used as a Fourier plane SLM, then the input pixel size, $p_i \sim 10\mu m$, for a lens with focal length, F=20 cm and the wavelength of the illumation, λ=633 nm. This result is obtained by using the equation

$$p_i = \frac{\lambda F}{128 p_f}.$$

Because SLM with resolution of 100 lines/mm is not available, either lens with bigger F (thus lower numerical aperture), or more lenses have to be used in order to match the pixel size of the input and the reference. Either case means further reduction in optical efficiency and larger optical system.

v. the switching threshold of pixels is very sensitive to the heat the device itself generates while operating. This means "warm up" is needed for the device to reach a steady state or more equipment is required to control the operating temperature of the device.

V.1.2. Optical Experiment

A simple experiment was performed to demonstrate the bipolar modulation property of the MOD. A grating was recorded on a device with 48x48 pixels by magnetizing alternate columns in opposite directions. The Fourier transform and

the image of this grating were optically formed for different settings of the analyzer. The results of the experiment are shown in Figure V.1.2.1.

In Figure V.1.2.1a the analyzer was set parallel to the incident polarization. The diffraction pattern in this case is due to the pixel structure of the MOD; the pattern is symmetric in the two dimensions, indicating that the 1-D grating information is not present. The photographs in Figure V.1.2.1b were recorded by setting the analyzer perpendicular to the incident polarization. The diffraction pattern in this case is that of a bipolar or binary-valued phase grating. Notice that the DC component is absent in the diffraction pattern in Figure V.1.2.1b. There is no contrast in the corresponding image in this case since the intensity of the bipolar modulation is detected. When the analyzer is set so that one of the rotated polarizations is blocked (Figure V.1.2.1c), the contrast is maximized in the image and the Fourier transform becomes that of a conventional amplitude grating. In all the diffraction patterns, the sizes of the diffracted spots are circular, small, and similar. This observation indicates that the device is indeed optically flat and the pixels' characteristics are uniform.

V.1.3. Computer Simulation

Because of the binary nature of the MOD, quantization noise is inevitable. For most images, the quantization noise level increases as the number of bits used to represent each data point decreases. The exceptions are binary and half-tone images. In signal processing, how close a quantized image represents a real image is only of secondary importance. Of primary importance is how a quantized represen-

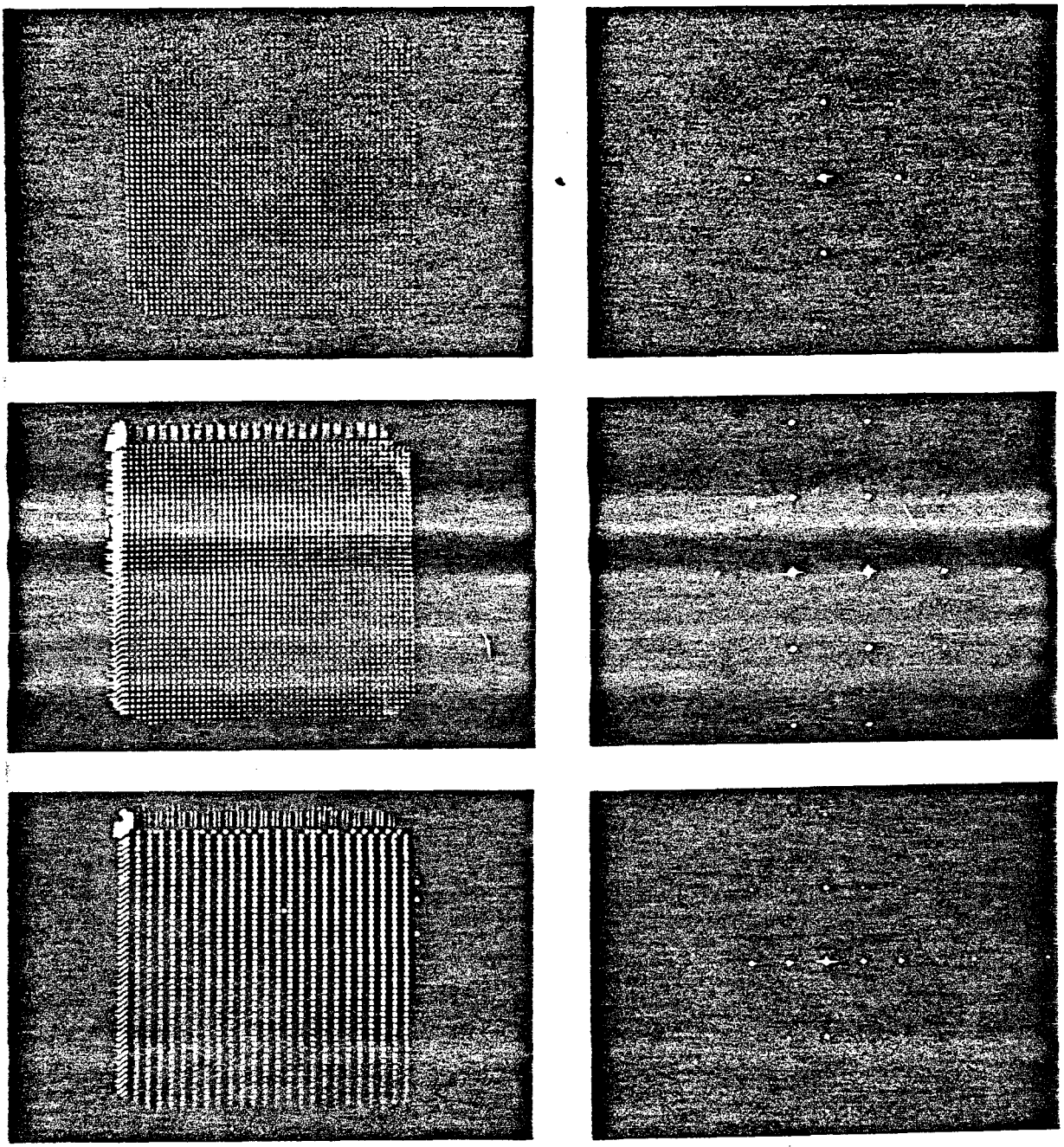


Figure V.1.2.1. The different settings of the analyzer and the corresponding diffraction patterns. a.) The analyzer was set parallel to the incident polarization. b.) The analyzer was set perpendicular to the incident polarization. c.) The analyzer is set so that one of the rotated polarizations is blocked.

tation affects the processed output. For example, one figure of merit in correlation is the signal-to-side-lobe (Chapter II). While it is true that the signal-to-side-lobe ratio also decreases as the number of bit planes in a quantized image decreases, the precise effect of quantization noise remains to be determined. But with only a little a priori knowledge of the image to be quantized, precise determination of the quantization effect is very difficult. To observe the contribution of each bit plane to the peak-to-side-lobe ratio, one of the best methods is perhaps to obtain the correlation of a real-life image and its quantized representation. Digital experiments were performed to illustrate this effect. The real-life image is shown in Figure V.1.3.1.

The infrared image in the box in Figure V.1.3.1. is a bird's-eye view of a power plant. To obtain the quantized representation of the power plant image, the box is divided into a uniform grid of 128 by 128 elements. Each element is then digitized to a 7-bit binary word. The discrete Fourier transform (DFT) of the digitized image is computed and only the sign bit and the 6 most significant bits (a total of 7 bits) of each sample of the DFT are saved. 7 frequency plane filters are then formed by adding different numbers of bit planes of the DFT: the first one is formed by just the sign bit of the DFT; the second one is formed by the sign bit and the most significant bit; and each additional filter is formed by adding one more bit at a time so the 7th filter has all the bit planes of the DFT saved. Correlations are obtained between the 7-bit digitized image and the 7-frequency plane filters. The significant part of the result is shown in Figure V.1.3.2.

Figure V.1.3.2a is the correlation output of the 7 bit image and the 1st filter (formed by only the sign bit). In this case, the peak-to-side-lobe ratio is about 11. Shown in Figure V.1.3.2b is the correlation output between the 7-bit image and

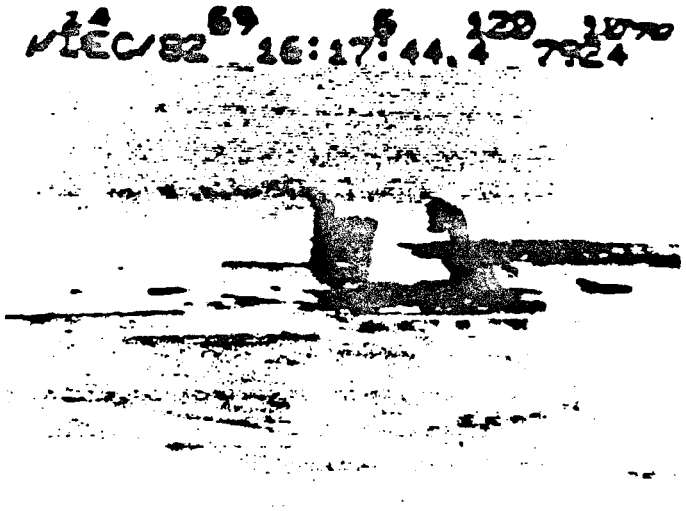


Figure V.1.3.1. The bird's eye view of a power plant.

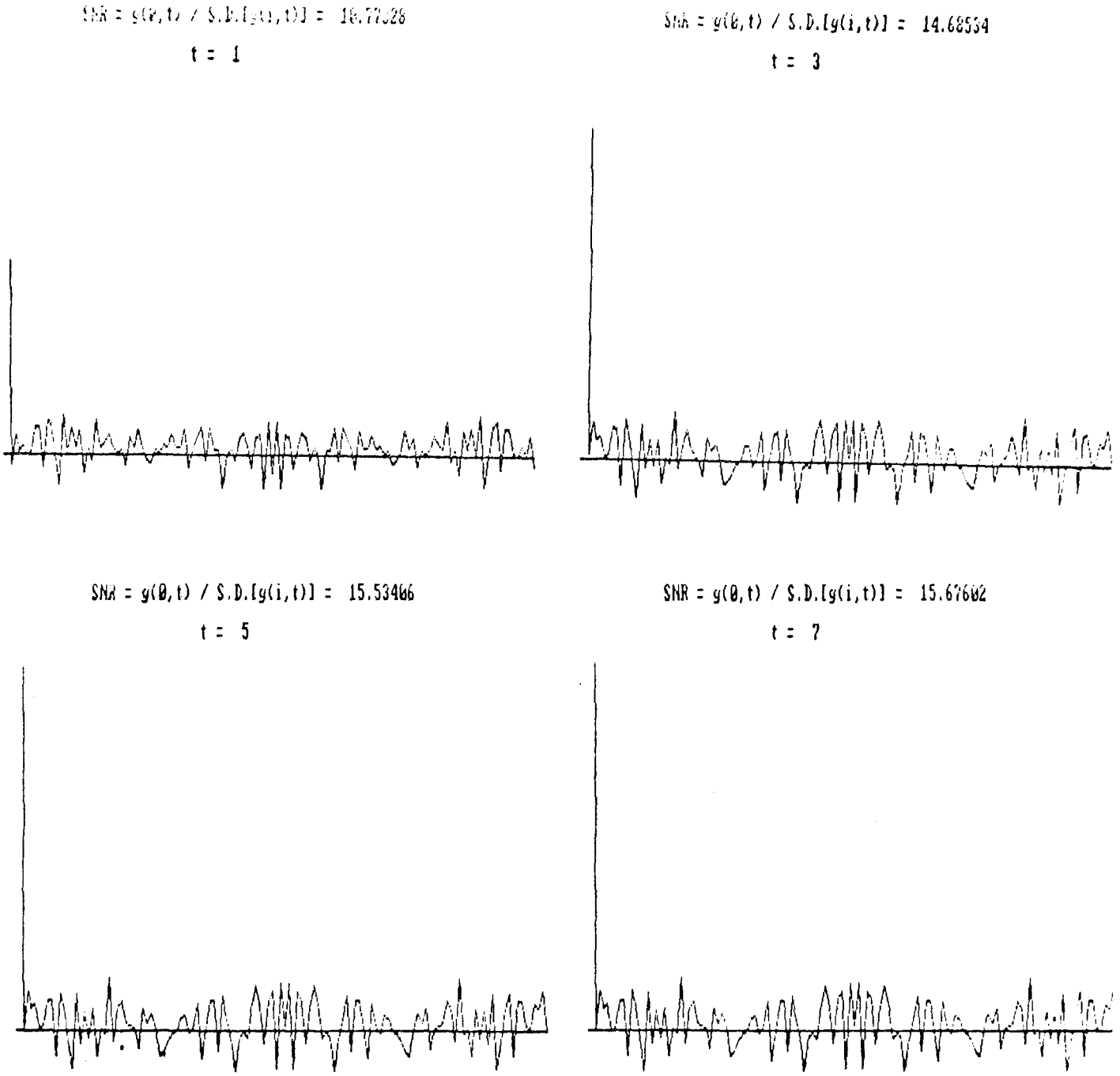


Figure V.1.3.2. The correlation outputs of the power plant and the filters formed by different of bit planes of the DFT of the power plant.

the 2nd filter. The peak to side lobe ratio in this case increases to about 13. The correlation output, when the 7th filter is used, is shown in Figure V.1.3.2d. The peak-to-side-lobe ratio of which case is about 16. Observe that when the number of bits in the frequency plane filter increases to 4, the peak-to-side-lobe ratio begins to saturate. This is a good sign since it means not too many bits are required in order to obtain a less than 5 percent decrease in peak-to-side-lobe ratio. Another promising observation is that the peak-to-side-lobe ratio is only reduced by 1/3 when only the sign bit representation is used. Note that this result agrees amazingly well with the theory obtained in Chapter II, even though the theory is based on images composed of white gaussian random numbers. Digital experiments with other real-life images (a bridge, an oil tank, and an oil field) were also obtained. The results are shown in Figure V.1.3.3. through Figure V.1.3.5. In those cases, the greatest loss in peak-to-side-lobe ratio, when only the sign bit representation is used, is about 1/2 (oil field). The best result is obtained from the simulation done with the bridge. The peak-to-side-lobe ratio in this case only suffers a modest loss of about 10 percent. Nonetheless, in all simulations, the peak-to-side-lobe ratio is acceptable when only the sign bit representation for the frequency plane filter was used.

V.1.4. Optical Experiment

Image correlation using the binary filter was also demonstrated experimentally with an optical system using the MOD. A CGH of the letter "O" was recorded according to eq.(II.1.1.2.) on a 128x128 MOD. A photograph of the hologram is shown in Fig.V.1.4.1a. and its optical reconstruction is shown in Fig.V.1.4.1b.

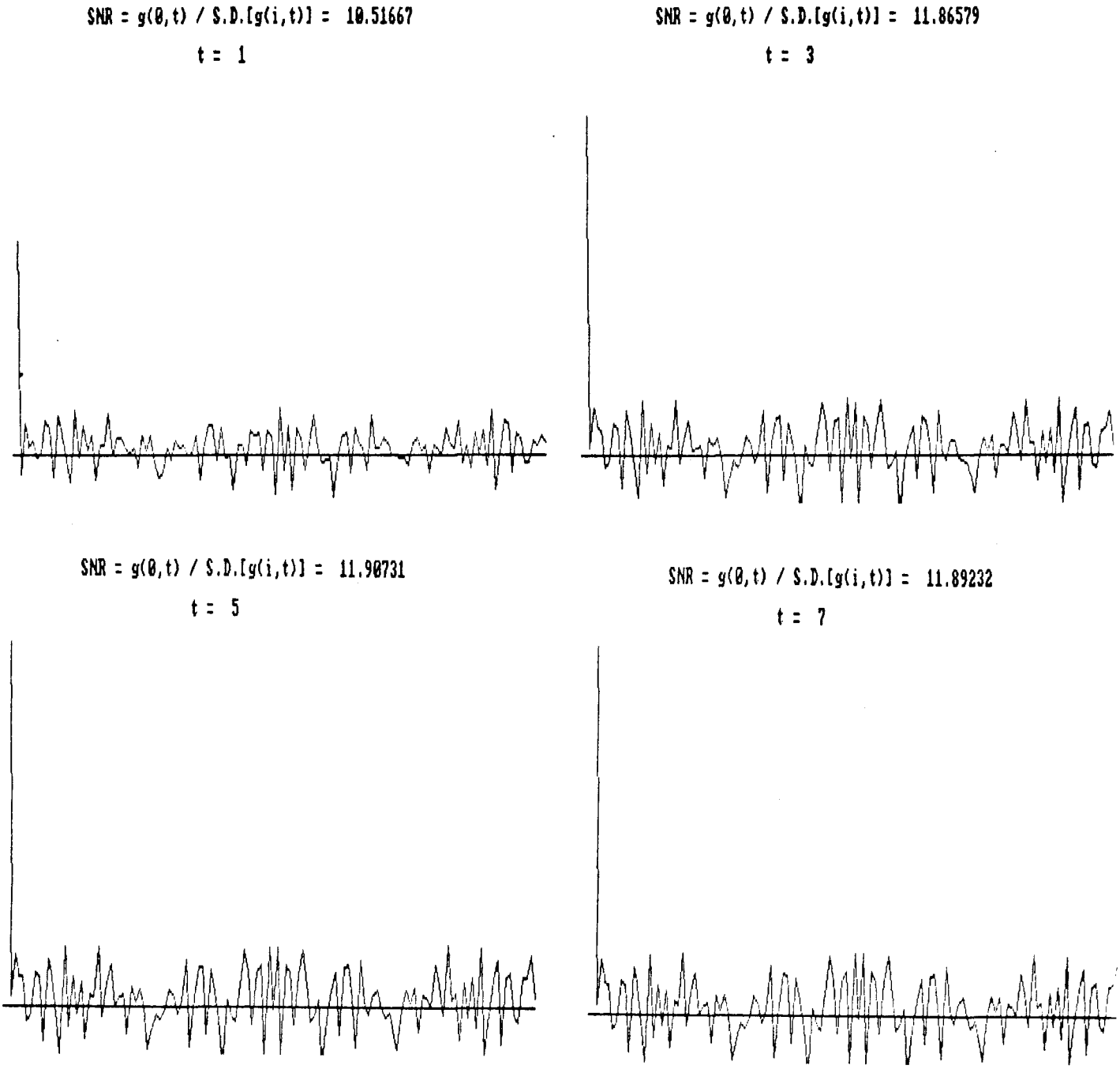


Figure V.1.3.3. Cross-correlations of a bridge and the filters formed by using different number of bit planes of the DFT of the image itself.

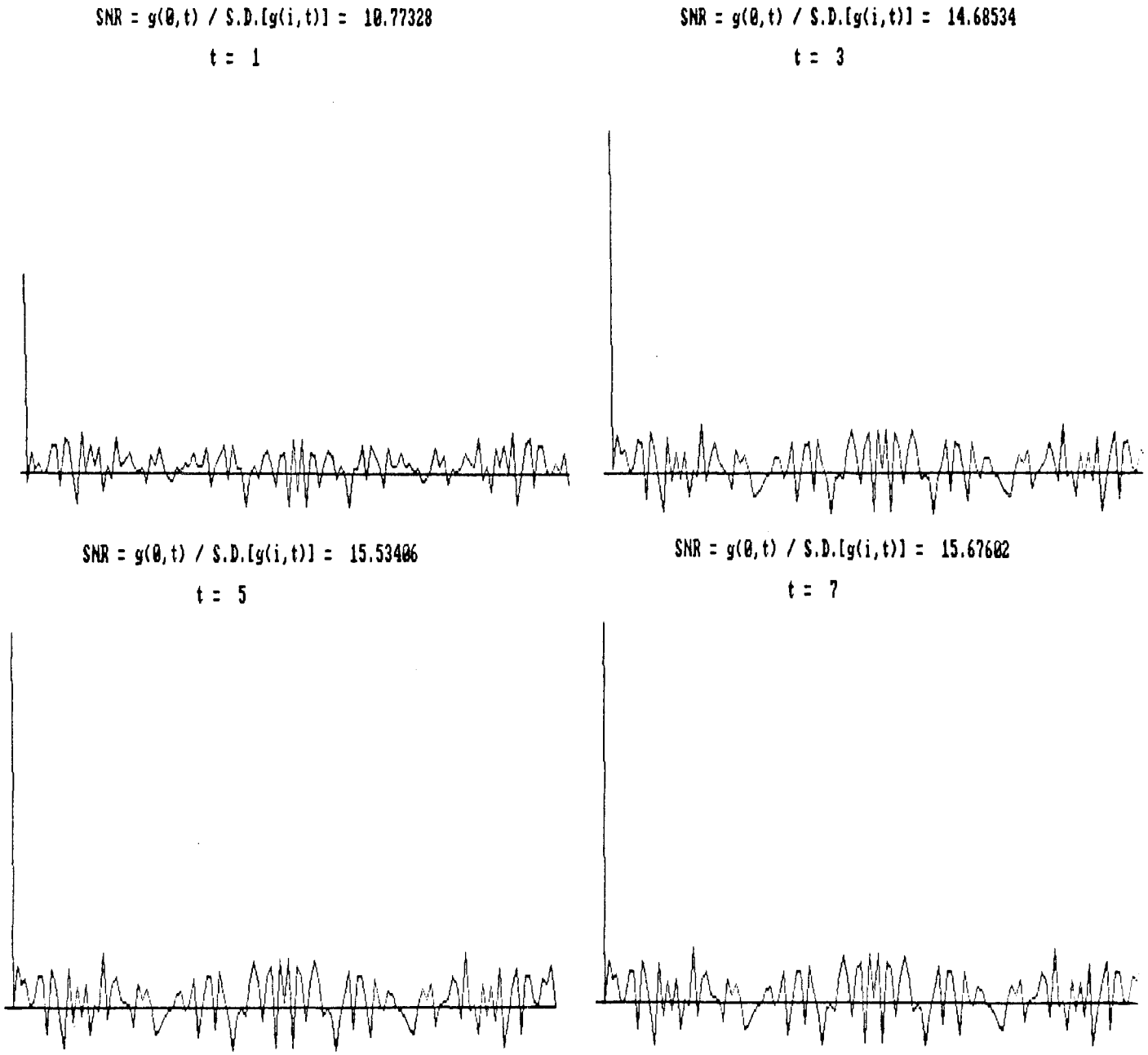


Figure V.1.3.4. Cross-correlations of an oil tank and the filters formed by using different number of bit planes of the DFT of the image itself.

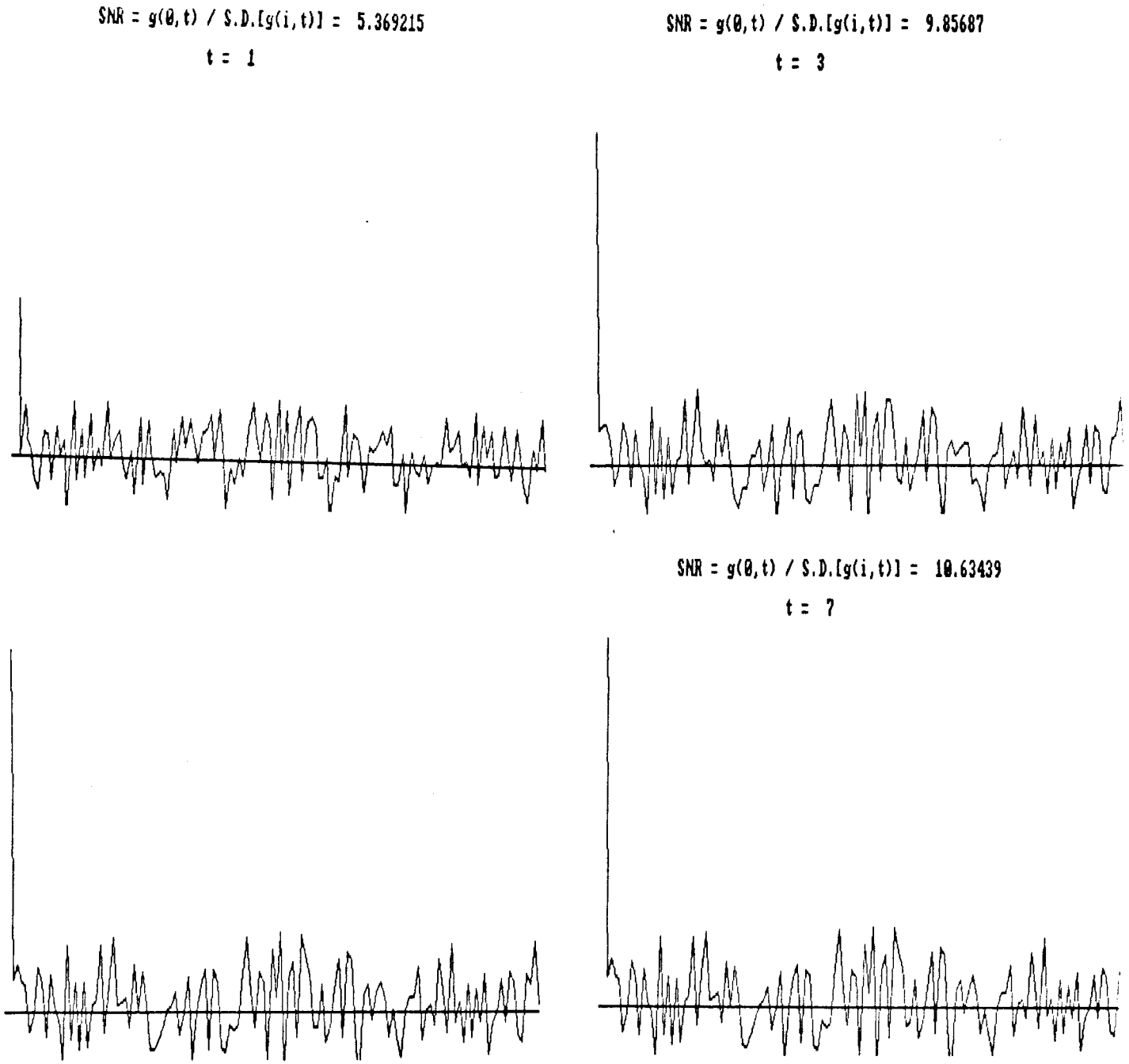
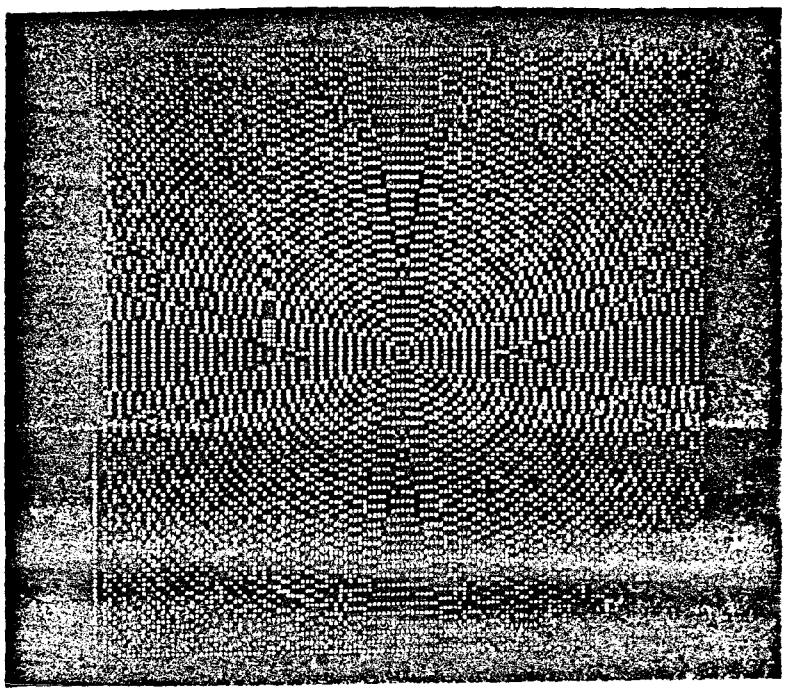


Figure V.1.3.5. Cross-correlations of an oil field and the filters formes by using different number of bit planes of the DFT of the image itself.



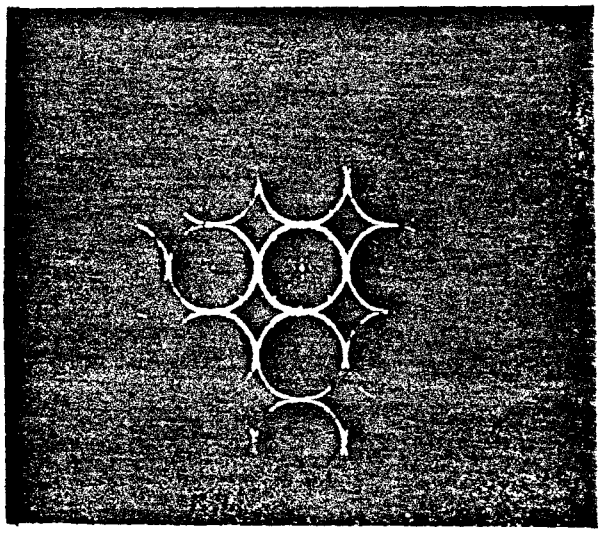
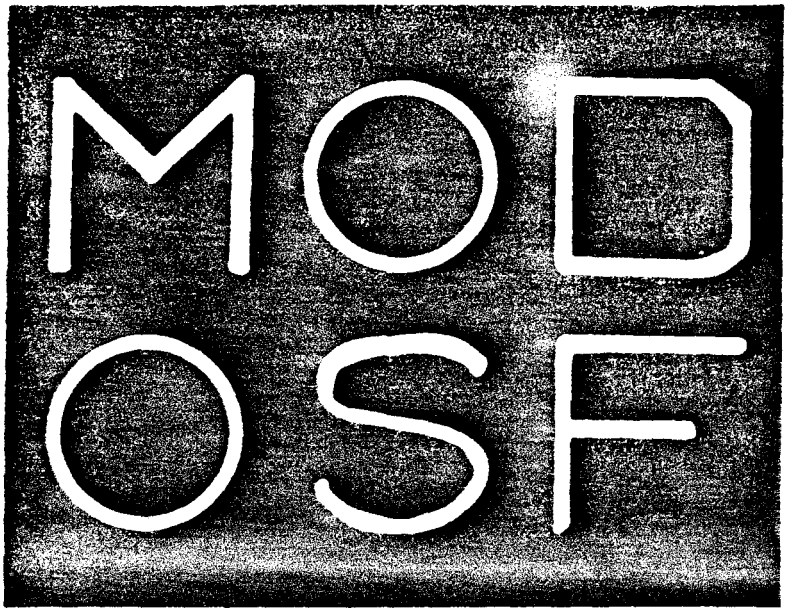


Figure V.1.4.1. A computer-generated hologram of the letter "O" recorded on the MOD, and its reconstruction.

An interesting feature of this hologram is that the primary reconstruction appears on-axis. This happens because the transmittance of this CGH is real and bipolar (not complex), and since bipolar signals can be directly recorded on the MOD, it is not necessary to record the hologram on a high spatial frequency carrier. The reconstruction is of remarkably good quality considering the extreme quantization that is involved in eq.(II.1.1.2.). The success in the reconstruction is due to two factors. One of them is the symmetry of the letter "O," which means that there is no loss in phase information of the Fourier transform of the letter. The other is due to the simplicity of the letter "O." The true 2-D Fourier transform of the letter "O" resembles a 2-D sinusoidal function, which means not much amplitude information is lost through the quantization process. Note that multiple orders are observed because of the sampling of the pixels of the MOD.

To perform an optical correlation, the hologram of Figure V.1.4.1a. was placed at the Fourier plane of a Vander Lugt system, and a transparency of the object shown in Fig.V.1.4.2a. was placed at the input plane.

The transparency used in the experiment was approximately 1cmx1.5cm, and the focal length of the Fourier transform lens was 200mm. The correlation shown in Fig.V.1.4.2b. was obtained using a 5mW HeNe laser. A strong autocorrelation peak is obtained in each occurrence of the letter "O" in the input pattern; weaker spurious peaks due to the sampling in the hologram are also obtained. The value of the correlation function away from the peaks is at a comfortably low level; no attempt was made to compare the background level obtained here with that of the ideal matched filter. A considerable degree of variance was observed, which we attribute partially to the sensitivity of the MOD to the angle of incidence of the



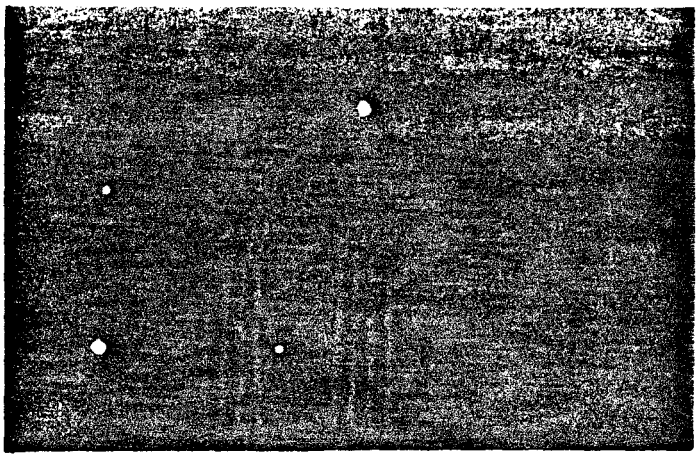


Figure V.1.4.2. (b.) The cross-correlation of (a.) and the CGH of the letter "O".

illuminating light. The space variance limits the size of the input transparency. In our experiment we obtained satisfactory results over a 4 cm^2 area at the input plane using a Fourier transform lens with a 200 mm focal length.

Optical correlation was also performed on the image of the power plant and its binary Fourier plane matched filter. The result is shown in Figure V.1.4.3.

The image of the power plant is shown in Figure V.1.3.1. Its binary Fourier plane matched filter is shown in Figure V.1.4.3a. Since the reconstruction of the filter does not resemble the power plant at all, it is not shown here. The sharp correlation peak is shown in Figure V.1.4.3b.

These experiments support the results of our earlier analysis that a binary spatial filter can provide satisfactory performance. But they also demonstrate that the MOD has sufficient phase uniformity to be used as a holographic optical element.

V.2. A Real-Time Programmable Correlator

The TSI 2-D correlator architecture is presented in this section. Before we commence the description, we will briefly review the fundamentals of operation of the two key components of the proposed architecture: AOD (acousto-optic device) and CCD (charge-coupled device) detector arrays. This will allow us to establish the notational convention we will be using, and will in general facilitate the presentation.

V.2.1. Description of AOD

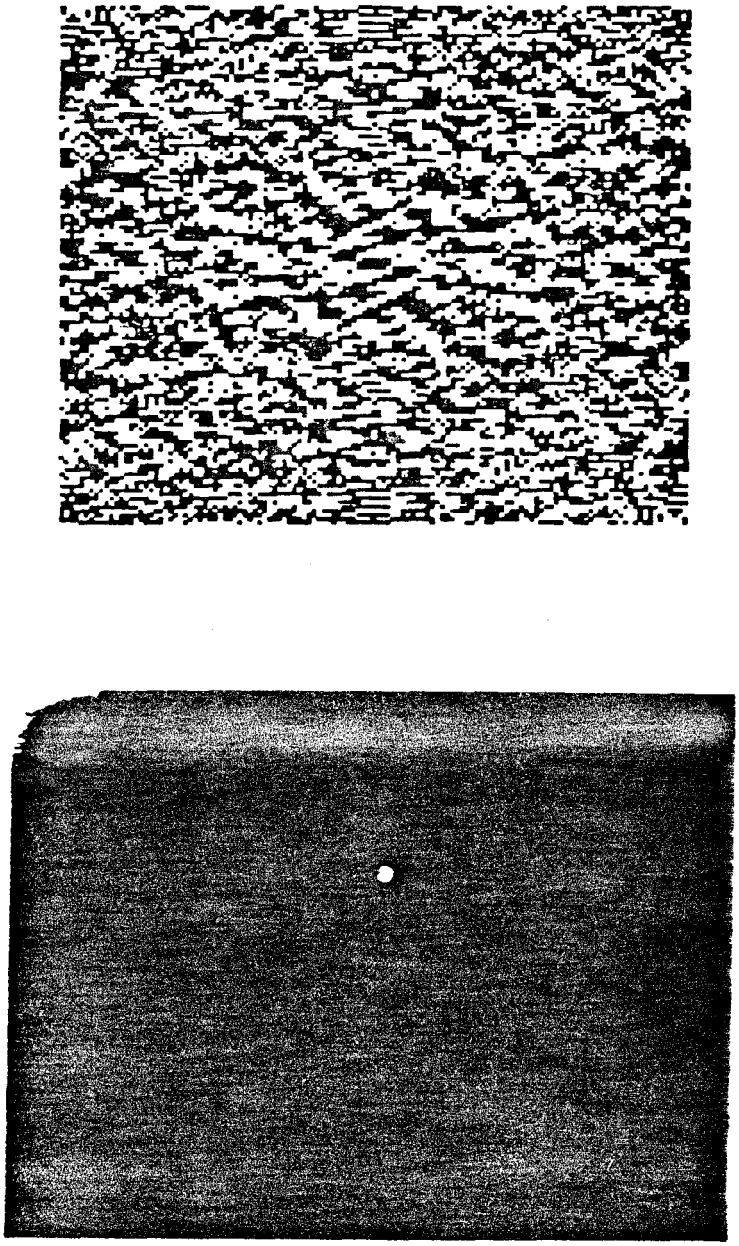


Figure V.1.4.3. The optical correlation of the power plant and its binary Fourier plane matched filter.

A schematic diagram of an acousto-optic deflector operating in the Bragg regime is shown in Figure V.2.1.1.

The voltage applied to the piezo-electric transducer is denoted by $s(t)$. The signal $s(t)$ has the form $s(t) = \alpha(t)cos[w_0 t + \phi(t)]$ where the complex envelope $a(t) = \alpha(t)e^{j\phi(t)}$ is the modulating signal and $w_0/2\pi$ is the center frequency of the AOD for which the Bragg phase matching is satisfied.

$$sin\theta_b = \frac{\lambda w_0}{4\pi v}$$

where λ is the wavelength of light in the acousto-optic crystal and θ_b is the angle of incidence of the optical plane wave. The acoustic wave that is generated at the transducer induces a variation in the index of refraction of the acousto-optic crystal proportional to $s(t - \frac{x}{v})$, where v is the speed of sound in the crystal in the direction of acoustic wave propagation x'. The spatial variation in the index of refraction causes a portion of the incident beam to be diffracted. In the Bragg regime, essentially all the diffracted energy is concentrated in the first order. For relatively weak modulation level (small change in the index of refraction) the amplitude of the diffracted light, $E(x,t)$, is given by

$$\begin{aligned} E(x',t) &\approx c_1 rect(\frac{x' - A/2}{A})\alpha(t - \frac{x'}{v})e^{j\phi(t-\frac{x'}{v})}\, e^{jw_0(t-\frac{x'}{v})}\, e^{j\frac{2\pi sin\theta_b x'}{\lambda}} \\ &= c_1 rect(\frac{x' - A/2}{A})a(t - \frac{x'}{v})e^{jw_0(t-\frac{x'}{v})}\, e^{j\frac{2\pi sin\theta_b x'}{\lambda}} \qquad (V.2.1.1.) \end{aligned}$$

In eq.(V.2.1.1.), c_1 is a constant and A is the aperture of the AOD. Thus, the amplitude of the diffracted light is modulated by the complex signal $a(t-\frac{x'}{v})$ over the spatial window A, is Doppler shifted by the travelling acoustic wave by $w_0/2\pi$, and is deflected around the Bragg angle θ_b. For the rest of our discussion we will not carry

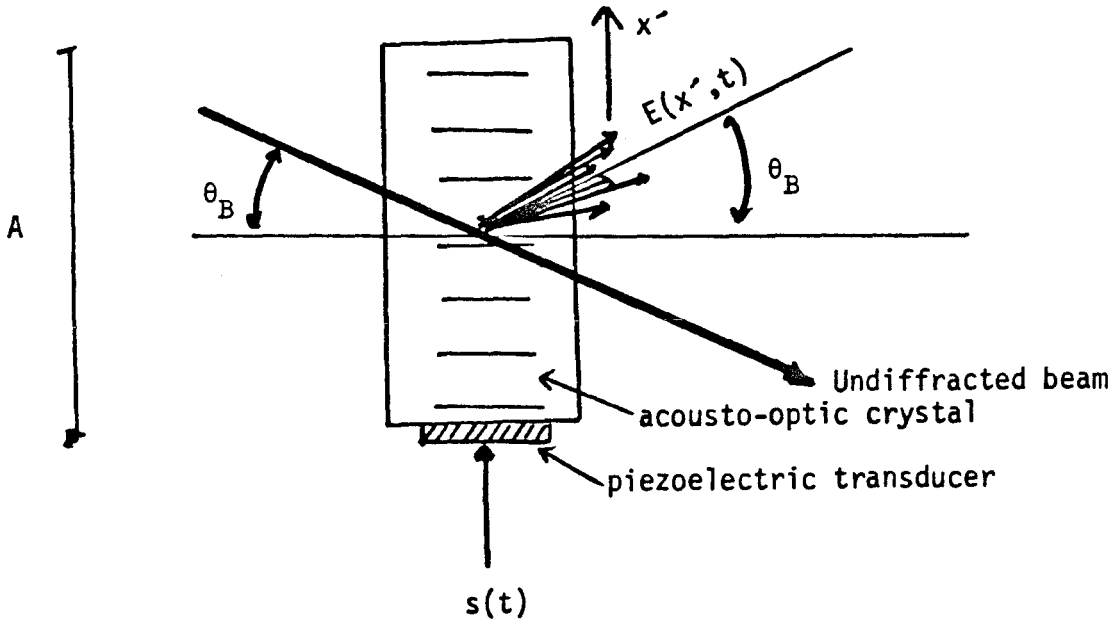


Figure V.2.1.1. Acousto-Optic Deflector.

the deflection term $e^{j\frac{2\pi sin\theta_b x'}{\lambda}}$ in the equations since it does not affect the operation of the system we will describe. In eq.(V.2.1.1.) the origin of the x' coorrdinate is at the piezoelelectric transducer, and x' increases away from the transducer. In our analysis it is more convenient to use the coorrdinate transformation $x = -x' + A$. In the x coordinate system, the origin is at the far end of the acousto-optical and x increases toward the transducer. Eq(V.2.1.1.) can be written as follows:

$$E(x,t) = c_1 rect(\frac{x - A/2}{A}) a(t + \frac{x}{v} - \frac{A}{v} e^{jw_0 t}). \qquad (V.2.1.2.)$$

We will use the above equation in the rest of our discussion.

V.2.2. Description of CCD

A CCD array is formed by a linear cascade of many pixels. Two-dimensional arrays are formed by growing side-by-side on the same chip a large number of linear arrays. Each pixel consists of three adjacent Metal Oxide Semiconductor (MOS) junctions. A positive voltage V_1 is applied to the first stage of each pixel during exposure. The positive voltage repels the majority carrier (holes in this case) and creates a depletion region below the oxide layer. The structure is illuminated through the transparent electrode and gate structure. Incident photons create electron-hole pairs in the semiconductor. The photogenerated positive carriers are repelled into the bulk of the semiconductor while the negative electrons in the vicinity of each pixel accumulate in the depletion region of the first stage of each pixel. The accumulated charge is a measure of the intensity of the illumination at each pixel location. After exposure, the voltage V_1 is lowered and simultaneously a positive voltage V_2

is applied to the middle stage at each pixel. With V_1 lowered, the charge in the first stage starts spreading, due to self-repulsion and diffusion, and drifts to the middle electrode where the positive voltage V_2 holds the charge together. When the CCD is operated as an image sensor, after a single exposure, each charge is shifted in the same manner through the entire array, until it reaches the output stage where it is measured. In our system, the CCD array will be exposed periodically. After each exposure the charge that is accumulated in the first stage of each pixel is shifted by only 3 stages of 1 pixel. Let the intensity of the illuminating light at the location y during the nth exposure be denoted by $I(n, y)$. The charge generated at the pixel located at y during the nth exposure is proportional to $I(n, y)$. After N exposures ($N \geq n$), the charge $I(n, y)$ shifts by N-n pixels or by a distance $(N - n)\triangle y$ in the y direction. $\triangle y$ is the pixel separation. Thus the charge that is accumulated in the CCD as a function of pixel position y after N exposures, is given by

$$Q(y) = \sum_{n=1}^{N} I[n, y + (n - N)\triangle y], \qquad (V.2.2.1.)$$

for $y = 0$ to $M\triangle y$, where M is the number of pixels in the CCD and $I(n, y) = 0$ for all $y \leq 0$. The maximum number of exposures N for which the addition in eq (V.2.2.1.) can be performed is equal to M.

V.2.3. Description of Operation

The proposed system architecture is shown in Figure V.2.3.1.

We will demonstrate that this system computes the cross-correlation $g(\xi, \eta)$ of

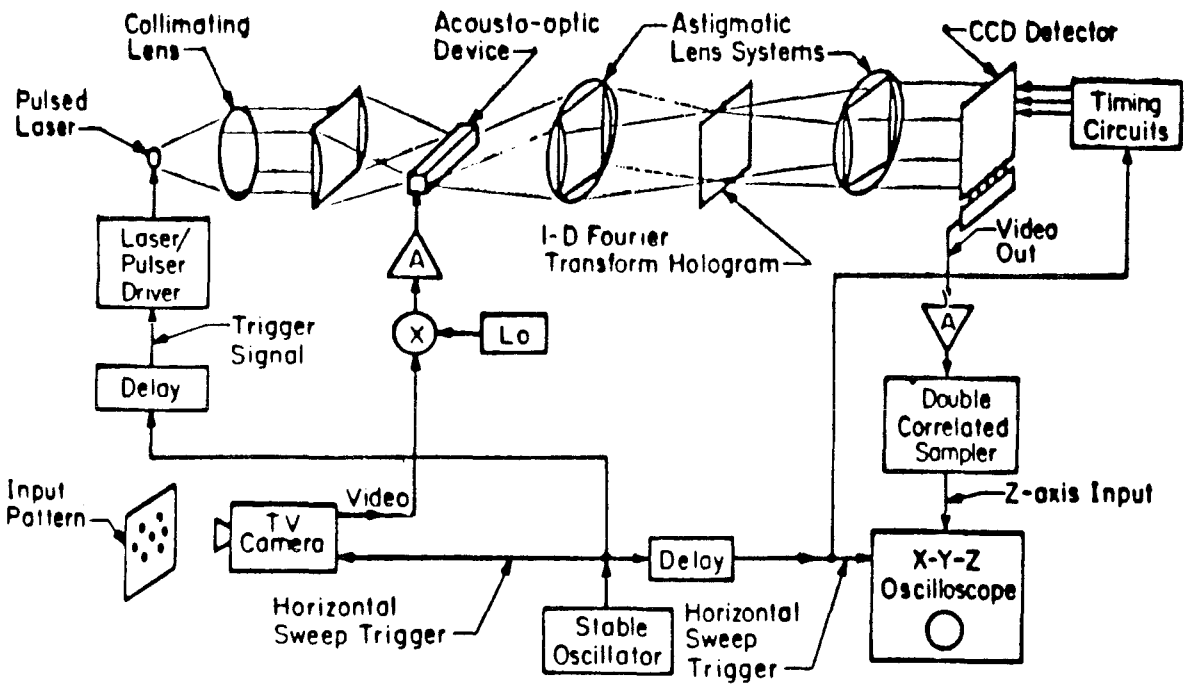


Figure V.2.3.1. The system architecture of the real-time, programmable, correlator.

two complex functions $f(x,y)$ and $h(x,y)$:

$$g(\xi,\eta) = \int\int f(x,y)h(x+\xi,y+\eta)dxdy.$$

The operation of the correlator in Figure V.2.3.1. can be summarized as follows: The optical portion of the system is a multichannel 1-D correlator that correlates one row of the function $f(x,y)$ with all rows of the function $h(x,y)$. The AOD is used to enter all rows of $f(x,y)$ sequentially in the optical system. The 1-D correlations are integrated appropriately by operating the CCD in the shift-and-add mode described above, to form the full 2-D correlation. In most image recognition applications, the input scene and the filter functions are real. For this reason and in order to present the principles of operation of the proposed system more clearly, we will restrict our analysis initially to real functions f and h. For the sake of completeness, we will describe the operation of the system with complex signals f and h at the end of this section.

The 1-D Fourier transform of a transparency with amplitude transmittance $h(x,y)$ is formed with an astigmatic lens system which transforms in the x-direction and images along y. A hologram of the resulting light amplitude distribution is formed by recording its interference with a plane wave reference on photographic film or any other suitable device. The amplitude transmittance of the developed hologram is proportional to

$$\begin{aligned}\left|Bexp(-j2\pi sin\theta_h F f_x) + H(f_x,y)\right|^2 &= B^2 + \left|H(f_x,y)\right|^2 \\ &+ BH^*(f_x,y)exp(-j2\pi sin\theta_h F f_x) \\ &+ BH(f_x,y)exp(j2\pi sin\theta_h F f_x),\end{aligned}$$

where

$$H(f_x, y) = \int h(x, y) exp(-j2\pi f_x x) dx, \qquad (V.2.3.1.)$$

θ_h is the angle of incidence on the hologram of the reference beam, B is its amplitude, F is the focal length of the lens in the x-direction, and f_x is the spatial frequency variable which is linearly related to the spatial variable x' in the plane of the hologram ($f_x = \frac{x'}{\lambda F}$). In eq.(V.2.3.1.), only the term containing H^* contributes to the correlation and the remaining terms are blocked out in the optical system. Thus the effective complex transmittance of the hologram can be written as

$$t_h(f_x, y) = BH^*(f_x, y) exp(-j2\pi sin\theta_h F f_x). \qquad (V.2.3.2.)$$

This hologram is placed in plane P_2 of Figure V.2.3.1. The image $f(x, y)$ is scanned in a raster format to produce a temporal electronic signal $r(t)$. The raster signal $r(t)$ is related to the 2-D function $f(x, y)$ by the following equation.

$$r(t) = f[(t - (n-1)T)v_s, n\delta y];$$
$$n = 1, \ldots, N, \qquad (V.2.3.3.)$$

where v_s is the scanning velocity of the device (such as a TV camera) that produces the raster signal, T is the duration of each raster line ($v_s T$ is the size of $f(x, y)$ in the x-direction), and δy is the resolution of $f(x, y)$ in the y-direction ($N\delta y$ is the size of $f(x, y)$ in the y-direction). In eq.(V.2.3.3.) we assume that $f(x, y) = 0$ for $x > v_s T$ and $x < 0$. $r(t)$ is heterodyned to the center frequency w_0 and applied to the AOD in plane P_1 of Figure V.2.3.1. The modulation $t_a(x, t)$ introduced by the AOD on the amplitude of the diffracted light can be found by substituting $a(t)$ by

$r(t)$ in equation (V.2.3.2).

$$t_a(x,t) = c_1 rect(\frac{x-a/2}{A})r(t+\frac{x}{v}-\frac{A}{v})e^{jw_0 t};$$
$$= c_1 rect\frac{x-A/2}{A} f[(t+\frac{x}{v}-\frac{A}{v}-nT)v_x, n\delta y]e^{jw_0 t} \qquad (V.2.3.4.)$$

We set $A = v_s T$, i.e., the aperture of the AOD can accommodate exactly one raster line of $f(x,y)/4$. For convenience, we also set $v_x = v$. At time instances $t = nT$ the modulation of the AOD is given by

$$T_a(x,nT) = c_1 f(x,n\delta y)exp(jw_0 nT)$$
$$= c_1 rect(\frac{x-A/2}{vT})f[(nT+\frac{x}{v}-\frac{A}{v}-nT+T)v, n\delta y]exp(jw_0 nT).(V.2.3.5.)$$

The *rect* function can be dropped in eq.(V.2.3.5.) since $f(x,y)$ was defined to be nonzero for $0 < x < vT$. Thus, at times $t = nT$ a single line of the function $f(x,y)$ modulated spatially the light diffracted by the AOD. A pulsed light source is used in the system to illuminate the AOD only at the instances $t = nT$. The temporal modulation of the source can be written as

$$t_s(t) = rect(\frac{t-nT}{\tau}) \qquad n = 1,\ldots,N \qquad (V.2.3.6.)$$

where τ is the duration of each light pulse, and the pulse shape has been approximated by a rectangular function. The light diffracted by the AOD is modulated by the product of eqs.(V.2.3.4.) and (V.2.3.6.). We denote this modulation function by $t_{sa}(x,t)$. For $\frac{1}{\tau}$ larger than the bandwidth of $r(t)$, t_{sa} can be approximated by

$$t_{sa}(x,a) = t_s(t)t_a(x,t)$$
$$= exp(jw_0 t)rect(\frac{t-nT}{\tau})f[(t+\frac{x}{v}-nT)v, n\delta y]$$
$$\approx f(x,n\delta y)exp(jw_0 t)rect(\frac{t-nT}{\tau}). \qquad (V.2.3.7.)$$

Lens L_1 in Figure V.2.3.1. is used to collimate the above light distribution in the vertical direction so that it illuminates the hologram in plane P_2 uniformly in the y-direction. Lens L_2 takes the Fourier transform in the x-direction. Thus the amplitude of the light entering plane P_2 is modulated by

$$t'_{sa}(f_x,t,n) = \int t_{sa}(x,t)exp(-j2\pi f_x x)dx$$
$$= F(f_x,n\delta y)exp(jw_0t)rect(\frac{t-nT}{\tau}) \qquad (V.2.3.8.)$$

The light immediately after plane P_2 is modulated by the product of eqs.(V.2.3.8.) and (VI.2.3.2.). The astigmatic lens system L_3 images plane P_2 onto the output plane P_3 in the y-direction, while it performs the Fourier transform in the horizontal (x) direction. The amplitude of the light at plane P_3 is given by

$$t_d(\hat{x},y,t,n) = \int t'_{sa}(f_x,t,n)t_h(f(x,y)exp(-j2\pi f_x\hat{x}'df_x$$
$$= c_2exp(jw_0t)rect[t-nT]$$
$$\cdot \int F(f_x,n\delta y)H^*(f_x,y)exp[-j2\pi(sin\theta_hF+\hat{x}')f_x]df_x$$
$$= c_2exp(jw_0t)rect(\frac{t-nT}{\tau})\int f(x,n\delta y)h(x+\hat{x},y)dx \qquad (V.2.3.9.)$$

where $\hat{x} = \hat{x}' + sin\theta_h F$ is the horizontal spatial coordinate in P_3 and c_2 is a constant. The convolution theorem was used to obtain the last form of eq.(V.2.3.9.). The output light distribution is detected interferometrically by the CCD detector array at the output plane in order to obtain a signal proportional to eq.(V.2.3.9.). A reference beam derived from the same source is heterodyned to the center frequency w_0 of the AOD and is made incident on the detector at an angle θ_d. The amplitude of the reference beam is described by

$$t_r(\hat{x},t) = Aexp(j2\pi sin\theta_d\hat{x}/\lambda)exp(jw_0t)rect(\frac{t-nT}{\tau}). \qquad (V.2.3.10.)$$

The signal $I(\hat{x}, y, n)$ that is detected by the CCD is proportional to the time integrated intensity of the sum of eqs(V.2.3.9.) and (V.2.3.10.):

$$\begin{aligned} I(\hat{x}, y, n) &= c_3 \int |t_d + t_r|^2 dt \\ &= \int_{nT}^{nT+\tau} |Aexp(j2\pi f_0 \hat{x} + \int f(x, n\delta y) h(x + \hat{x}, y) dx|^2 dt \\ &= \tau c_3 \Big[A^2 + |\int f(x, n\delta y) h(x + \hat{x}, y) dx|^2 \\ &\quad + 2A[\int f(x, n\delta y) h(x + \hat{x}, y) dx] cos(2\pi f_0 \hat{x}) \Big], \end{aligned} \qquad (V.2.3.11.)$$

where $f_0 = sin\theta_d/\lambda$. The third term in the above equation is the 1-D correlation of the $n\text{-}th$ row of $f(x, y)$ with all the rows of $h(x, y)$. The correlation forms on the spatial carrier f_0. Since both f and h are real functions, only the amplitude of the fringe pattern is modulated. In the y-direction, the CCD is operated in the shift-and-add mode described earlier. The charge that is accumulated in the CCD after N light pulses can be found by substituting eq.(V.2.3.11.) in eq.(V.2.2.1.)

$$\begin{aligned} Q(\hat{x}, y) &= \sum_{n=1}^{N} I[\hat{x}, n, y + (n - N)\delta y] \\ &= c_4 [NA^2 + \sum_{n=1}^{N} |\int f(x, n\delta y) h(x + \hat{x}, y + (n - N)\delta y|^2 \\ &\quad + 2A[\sum_{n=1}^{N} \int f(x, n\delta y) \\ &\quad \cdot h(x + \hat{x}, y + n\delta y - N\delta y) dx] cos(2\pi f_0 \hat{x})] . \end{aligned} \qquad (V.2.3.12.)$$

The third term in eq.(V.2.3.11.) forms on a spatial carrier of frequency f_0. By setting f_0 equal or larger to the bandwidth of $f(x, y)$ this term can be separated from the other two baseband terms by electronic filtering after the signal from the

CCD is converted to a video signal. The envelope of the carrier in eq.(V.2.3.12.) is recognized to be the 2-D correlation with the integration over the continuous variable y replaced by the summation over the discrete variable $n\delta y$. The correlation pattern is shifted by the constant $N\delta y$ in the y-direction. For a CCD detector with N horizontal rows, this means that each slice of the correlation pattern is formed sequentially at the last row of the device. A fast horizontal CCD transfers each line of the 2-D correlation to the output stage of the device where it can be displayed or further processed electronically.

V.2.4. Adaptation of the Architecture

In the proposed system, it is important to select the amplitude A of the reference beam in an optimum manner. Ideally we would like to choose A such that the peak value of the correlation function in eq.(V.2.3.12.) is equal to NA, in order to utilize fully the dynamic range of the detector. At the same time, we must choose A large enough so that the interference pattern formed during each light pulse is recorded linearly. Depending on the type of images being processed, the peak value of the signal recorded during each pulse can vary significantly from pulse to pulse. If this is the case, we will be forced to choose a value for A which can be significantly higher than the optimum. As a result, the bias would be greater than the peak value of the correlation in eq.(V.2.3.12.) with a corresponding loss in the dynamic range in which the signal can be detected. In these instances, it is preferable to use a reference beam whose amplitude depends on the energy of each line of the input function, and thus varies temporally and/or spatially. The fringe

pattern $cos[2\pi f_0 \hat{x}]$ in eq.(V.2.3.11.) must remain stationary from pulse to pulse, i.e., it must not depend on n. Any jitter in the location of the fringes will cause the signal term in eq.(V.2.3.12.) to decrease when the summation over n is performed. Two factors can adversely affect the stability of the fringe pattern: mechanical instability of the optical system and instability in electronic timing circuits. The bias build-up and the requirement for stability in the system are consequences of the use of interferometric detection in the system. We are considering several methods by which the additional complexity that is introduced by interferometric detection can be avoided. For instance, let us suppose that interferometric detection was not used at all. In this case the signal accumulated at the CCD is given by

$$Q(\hat{x},y) = \sum_{n=1}^{N} \left| \int f(x,n\delta y) h(x+\hat{x}, y+n\delta y) dx \right|^2. \qquad (V.2.4.1.)$$

The difference between the above function and the correlation contained in equation (V.2.3.12.) is that in eq.(V.2.4.1.) the 1-D correlation is squared before the summation over n is performed. It can be shown using the Schwarz inequality that the function in eq.(V.2.4.1.) is maximized when $f = h$ and the peak value occurs at $\hat{x} = 0$ and $y = 0$. In other words, this function exhibits some of the properties that make the correlation function useful for pattern recognition applications.

The most obvious way to introduce a programmable reference to the TDI correlator is by recording the 1-D Fourier transform hologram on a SLM rather than on photographic film. One good candidate for the real-time SLM is the Litton LIGHT-MOD. Its use as a spatial filter in optical correlators has been investigated in previous sections. The algorithm used to generate a 2-D suboptimum binary and bipolar filter has been verified both theoretically and experimentally. However,

in the TDI case, we need a slightly different algorithm for the generation of the reference filter. Since only one dimension of the correlation is done using the space integrating capability of optics in the TDI operation, only a 1-D Fourier transform of the image to be matched is required. The modified algorithm can be described as follows:

$$H(u,y) = \beta\Big[Re[F(u,y)]\Big]$$

where

$$F(u,y) = \int f(x,y) e^{j\frac{2\pi}{\lambda F}ux}\, dx.$$

$f(x,y)$ is the pattern to be matched. In other words, the reference filter is obtained by taking the sign information of the real component of the Fourier transform of $f(x,y)$ in only the x direction. This modification in generating a binary and bipolar reference filter actually benefits the peak-to-side-lobe ratio of the correlation. It is because only a 1-D Fourier transform is required and, thus, only half of the information associated with the antisymmetric component of the pattern is discarded in the quantization process. The relative amount of information loss can perhaps be better illustrated by considering the following special case: suppose $f(x,y)$ can be written as $f_x(x)f_y(y)$. The reference filter is hence given as

$$H(u,v) = \beta\Big[Re[F_x(u)]\Big]\beta\Big[f_y(y)\Big] \qquad (V.2.4.2.)$$

where

$$F_x(u) = \int f_x(x) e^{j\frac{2\pi}{\lambda F}ux}\, dx.$$

It is clear from eq.(V.2.1.1.) that only the information associated with the imaginary component of $F_x(u)$ is totally discarded. As a comparison, all the information

associated with the antisymmetric component of $f(x, y)$ is destroyed in a 2-D binary and bipolar reference filter.

V.2.5. Optical Experiment

We will begin this section by briefly describing the operation of the electronic timing circuit. Before a correlation commences, the reference filter is written onto the MOD. Since the pattern to be recognized is input via a raster scan TV camera, a TDI cycle is triggered by the frame (vertical) sync of the video signal. The video signal is multiplexed by a carrier frequency before applying to the AOD. As only one video line (the separnnations of video lines are denoted by the horizontal sync of the video signal; there are 512 video lines in a US commercial TV) is entirely within the window of the AOD, the pulsed light source is triggered. The triggering is achieved by properly delaying the horizontal sync of the same video line. The pulsing of the light source practically "freezes" the information of the video line. The constraint on the pulse width will be discussed later. The 1-D correlation achieved by the space integrating technique is recorded on the CCD detector. As soon as the pulsed light level is dropped back to zero, the detected signal is ready to be shifted downward by one line. (The shifting operation of the CCD is explained in section..) The shifting is triggered from the same horizontal sync properly delayed. The only requirement is that the shifting operation has to finish before the next light pulse is triggered. When the shifting is complete, the readout register of the CCD detector contains one video line of the 2-D correlation output to be obtained. The rest of the lines of the 2-D correlation output are obtained in the identical manner. At the end of one

input frame, a complete frame of the entire TDI correlation is acquired. Because its format is compatible with a standard video signal, the output signal line of the CCD detector can be connected to a video monitor, which allows the real time TDI correlation to be observed.

We now describe the experimental setup. The reason for the choice of the optical components and the width of the light pulse will be given later. In the experiment, the light source is a 1W argon laser with the line 514 nm selected. To provide the light pulse, a glass AOD with a center frequency of 40 MHz is used. The raw beam from the argon laser is focused at the center of the glass crystal. A light pulse diffracting at the Bragg angle was obtained when a 15 kHz sin wave modulated by a 100 ns square pulse is applied to the transducer of the glass AOD. An approximate of 50 percent diffraction efficiency was obtained. The pulsewidth was chosen to be 100ns simply because it has to be no longer than 2 times the inverse of the bandwidth of the video signal so that the motion of the signal in the light diffracted by the AOD can be neglected. The video bandwidth in this experiment was less than 5MHz. The video signal from the TV camera was heterodyned to the center frequency of the input AOD (50MHz), amplified, and applied to the AOD. The acousto-optic device in this experiment was a TeO_2 (crystal technology #4050S) device with 35 MHz bandwidth and 70 microsecond delay. This was more than adequate to accommodate one standard video line (63 microseconds and 5 MHz). After approximately 52.7 microseconds from the start of the horizontal clock, the signal in the AOD is an acoustic replica of the video line from the input image. At that instant the light pulse is fired to produce a readout of the signal in the AOD.

The light diffracted by the AOD was Fourier transformed in the horizontal direction and imaged vertically to illuminate the binary and bipolar one-dimensional Fourier transform hologram written on the MOD. The optical system chosen to handle this job has an effective focal length of 3.2 m in the horizontal direction and a magnification of 1 in the vertical direction. The vertical magnification is 1 simply because the height of the AOD window and the MOD are both 1cm. With the effective focal length of the Fourier transform lens system of 3.2 m, the effective aperture size of the AOD can be obtained by using the following equation:

$$A = \frac{\lambda F}{r}$$

where r is the resolution of the MOD which is 64 line pairs per mm. Substituting $\lambda = 514nm$, $F = 3.2m$ and $r = 64lp/mm$ into the above equation, we have $A = 36\mu s$ which is about 50 percent of the total useable window of the AOD. In this case the pixel size of the input signal on the AOD is $.28\mu s$, about .4 percent of the total aperture. This is the reason that such an effective long focal length is required. For any pixel size less than this determined value, the optical power diffracted would not be sufficient. Recall that the optical efficiency of the MOD is only 4 percent. The effective focal length and magnification requirement on the lens system was fulfilled by using three spherical lenses with focal lengths of 40cm, 5cm, and 40cm, and a concave cylindrical lens with focal length of 20cm, respectively .

The light transmitted through the MOD is inverse Fourier transformed in the horizontal direction and imaged in the vertical direction onto the CCD. The CCD detector used in the experiment was a SONY XC-37 whose driving electronics were modified to allow us to scroll the charge on the CCD continuously during each frame.

This CCD camera has 384 pixels in the horizontal direction and 491 pixels in the scrolling direction. To ensure that the resolution of the correlation output does not exceed that of the CCD camera. A Fourier transform lens of 20cm (obtained by using eq..) focal length was used. The imaging on the vertical direction is achieved by an additional 7.6 cm focal length cylindrical lens. The scrolling action of the CCD completes the 2-D correlation as described earlier, and the full 2-D correlation is produced at 30 frames per second and displayed on a monitor.

The experimental result, demonstrating the TDI mode 2-D correlation using the experimental setup described in the previous paragraph, was obtained. The result is shown in Figure V.2.5.2. The input pattern that was imaged onto the TV camera was the letter X shown in Figure V.2.5.2a. The one-dimensional Fourier transform hologram of the letter X, computed and then recorded on the MOD, is shown in Figure V.2.5.2b. The autocorrelation of the letter X that was produced in real-time as a video signal by the CCD and displayed on a monitor, is shown in Figure V.2.5.2c.

V.3. Binary Rotation Invariant Filters

The correlation methods that we have examined so far only work when the orientation of the input object is known. In many applications, however, the object's orientation, in addition to its location, is usually not specified. One method to detect a rotated object is through the use of Mellin transform. This method basically makes use of a cartesian to polar coordinate transformation and thus trades

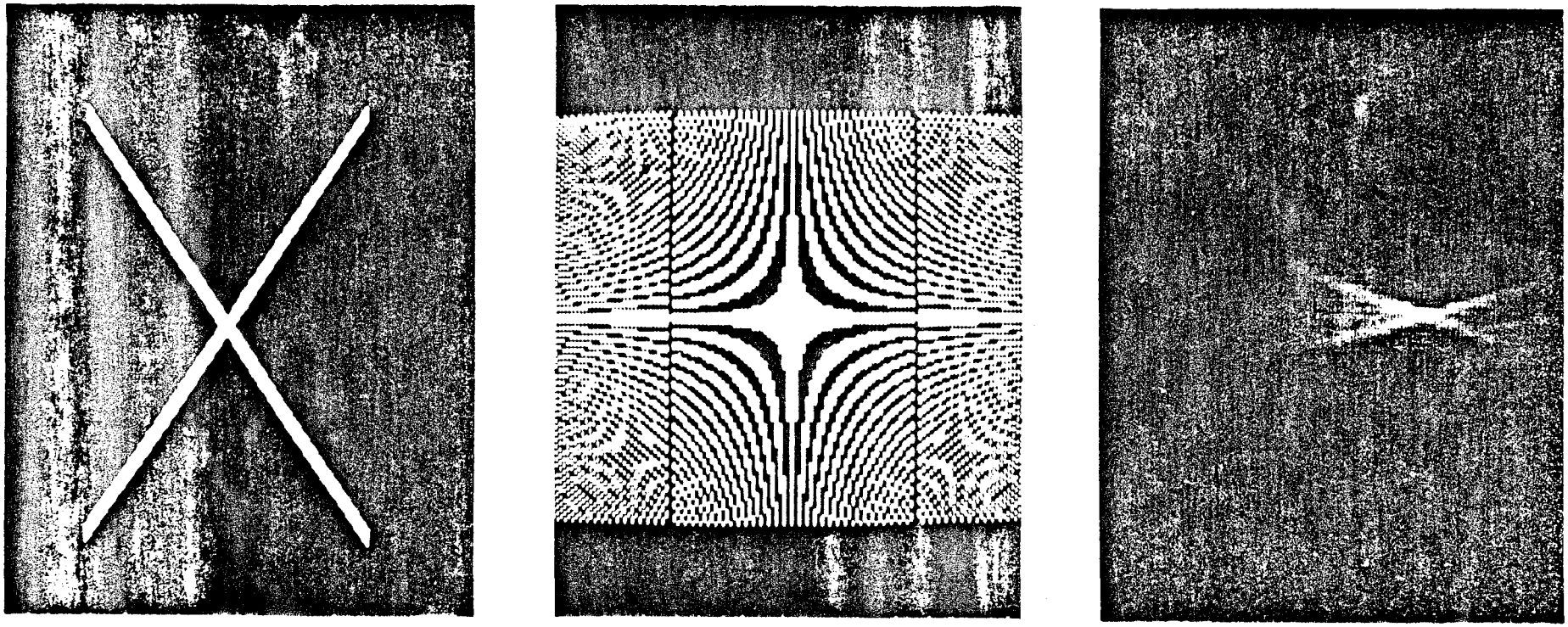


Figure V.2.5.2. (c.) The cross-correlation of (a.) and (b.).

shift-invariant for rotation invariant correlation. Another technique which allows both location- and orientation-insensitive correlation involves circular harmonics decomposition. In this section, we will investigate the possibility of using a binary version of this rotation and translation invariant technique for correlation. We will begin by reviewing the fundamentals of circular harmonic expansion.

V.3.1. Basic Theory

Mathematically, any pattern in polar coordinate $f(r,\theta)$ can be decomposed into its circular harmonic components:

$$f(r,\theta) = \sum_{m=-\infty}^{\infty} f_m(r) exp(jm\theta)$$

where

$$f_m(r) = \frac{1}{2\pi}\int_0^{2\pi} f(r,\theta) exp(-jm\theta) d\theta. \qquad (V.3.1.1.)$$

Note that the coefficient $f_m(r)$ of each circular harmonic component $exp(jm\theta)$ is not a function of θ. If $f(r,\theta)$ is rotated by an angle α, then

$$f(r,\theta+\alpha) = \sum_{m=-\infty}^{\infty} f_m(r) exp(jm\theta) exp(jm\alpha) \qquad (V.3.1.2.)$$

The circular harmonic components are all orthogonal, for $n \neq m$:

$$\int_0^{2\pi} f_m(r) exp(jm\theta) f_n(r) exp(jn\theta) d\theta = 0, \qquad m \neq n.$$

The effect induced on the correlation output when one of the inputs is rotated by angle of α can be illustrated by considering the following: Let $f_1(x,y)$ and

$f_\alpha(x,y)$ denote $f(r,\theta)$ and $f(r,\theta+\alpha)$ in cartesian coordinates. The cross-correlation between f_1 and f_α is given as

$$R_\alpha(x,y) = \int\int f_1(\xi+x,\eta+y) f_\alpha(\xi,\eta) d\xi d\eta. \qquad (V.3.1.3.)$$

When $\alpha = 0$, the autocorrelation peak $R_0(0,0)$ appears at $x = 0, y = 0$. For an arbitrary value of α, however,

$$R_\alpha(0,0) = \int\int f_1(\xi,\eta) f_\alpha(\xi,\eta) d\xi d\eta. \qquad (V.3.1.4.)$$

In general, $R_\alpha(0,0)$ is drastically reduced for $\alpha \neq 0$ We will refer to $R_\alpha(0,0)$ as the center correlation value.

The influence of the rotation angle α on the center correlation value can be shown by rewriting eq.(V.3.1.4.) in polar coordinates. Let $C(\alpha)$ denote $R_\alpha(0,0)$ in polar coordinates. Then

$$C(\alpha) = \int_0^\infty r dr \int_0^{2\pi} f(r,\theta+\alpha) f(r,\theta) d\theta. \qquad (V.3.1.5.)$$

Substituting the circular harmonic expansion of $f(r,\theta)$ given by eq.(V.3.1.1.) into eq.(V.3.1.5.) yields

$$C(\alpha) = \int_0^\infty r \Big[\sum_{m=-\infty}^{\infty} f_m(r) \int_0^{2\pi} f(r,\theta+\alpha) exp(-jm\theta) d\theta \Big] dr.$$

In view of eq.(V.3.1.2.), we can write the above equation as

$$C_(\alpha) = 2\pi \sum_{m=-\infty}^{\infty} exp(jm\alpha) \int_0^\infty r |f_m(r)|^2 dr. \qquad (V.3.1.6.)$$

Eq.(V.3.1.6.) expresses the center correlation value in the form of a summation of the contribution from all the circular harmonic components. Such a summation does not, in general, remain constant when α varies.

After realizing that the cause of variation of the center correlation is the contributions to the correlation from all of the circular harmonic components, it is natural to suggest that only a single circular harmonic is used as the reference filter. Suppose $h(r,\theta) = f_m(r)exp(jm\theta)$ is the reference filter in a correlator. The center correlation between $f(r,\theta+\alpha)$ and $h(r,\theta)$ can be expressed as

$$C_m(\alpha) = Ae^{(jm\alpha)},$$

where

$$A_m = 2\pi \int_0^\infty r|f_m(r)|^2 dr \qquad (V.3.1.7.)$$

The output intensity

$$|C_m(\alpha)|^2 = A^2$$

is constant independent of the orientation of f, which indicates that circular harmonics may be suitable for rotation invariant pattern detection.

Translation invariance can also be incorporated in a correlator using a circular harmonic component as the reference filter. This additional invariance can be achieved by simply utilizing the Fourier transform of the circular harmonic selected as the reference pattern. It can be shown that the Fourier transform of the $m\text{-}th$ order circular harmonic is given as

$$F_m(\rho,\Theta) = (-1)^m F_m(\rho)e^{jm\Theta}, \qquad (V.3.1.8a.)$$

where (ρ,Θ) are polar coordinates in Fourier space,

$$F_m(\rho) = \int_0^\infty f_m(r)J_m(2\pi r\rho)rdr \qquad (V.3.1.8b.)$$

and $J_m(.)$ is the m-th order Bessel function.

The circular harmonics components of a real image, in general, are gray valued complex functions. From eq.(V.3.1.8.), their Fourier transforms are also gray-valued and complex. Thus, the algorithm for generating the rotation and translation invariant filter has to be modified if a binary and bipolar SLM is used to record the reference. The procedure used for generating the suboptimum matched filter described in Chapter II can be employed here as well. Suppose the m-th order circular harmonic of $f(r,\theta)$ is to be matched. Adapting eq.(II.1.2.1.) to suit the present task, the Fourier transform reference filter $H(\rho,\Theta)$ in polar coordinates is given as

$$H(\rho,\Theta) = \beta\Big[Re[(-1)^m F_m(\rho)e^{jm\Theta}]\Big] \qquad (V.3.1.9.)$$

From here on, the filter given by the above equation will be addressed as the m-th order binary Fourier transform circular harmonic filter or abbreviated as the m-th BFTCHF.

Recall that if the input image to be correlated is a symmetric function, i.e.,

$$f(x,y) = f(-x,-y)$$

in cartesian coordinates or in polar coordinates

$$f(r,\theta) = f(r,\theta+\pi),$$

then there is no loss in information associated with the antisymmetric component of f when the binary suboptimum Fourier transform filter is formed. It is because the Fourier transform of a symmetric function has no imaginary component. As for the

circular harmonic filter, except for the 0-th order, the other circular harmonic components are complex. It is therefore natural to assume that some phase information associated with a circular harmonic component is lost when the filter described by eq.(V.3.1.9.) is implemented. While it is true that the entire correlation output will be different if the phase information of the imaginary component can be retained, the center correlation value, fortunately, is not affected. Consider the coefficient of the m-th order circular harmonic. From equation (V.3.1.1.),

$$f_m(r) = \frac{1}{2\pi}\int_0^{2\pi} f(r,\theta)cos(m\theta)d\theta. + \frac{j}{2\pi}\int_0^{2\pi} f(r,\theta)sin(m\theta)d\theta.$$

But since f is symmetric, the second term on the RHS of the above equation is zero. In other words, $f_m(r)$ is real. In view of eq.(V.3.1.8b.), $F_m(\rho)$ is also real. Applying this result to eq.(V.3.1.9.), we obtain

$$H(\rho,\Theta) = \beta\Big[(-1)^m F_m(\rho)cos(m\Theta)\Big]. \qquad (V.3.1.10.)$$

Now the Fourier transform of the circular harmonic decomposition of $f(r,\theta)$ is given as

$$f(r,\theta) = \sum_{\substack{n=-\infty \\ n\ \text{even}}}^{\infty} (-1)^n F_n(\rho)e^{jn\Theta},$$

where $F_n(\rho)$ is given by eq.(V.3.1.8b.) and n is even because $f(r,\theta)$ is symmetric. When $f(r,\theta)$ is applied to the filter described by eq.(V.3.1.10.), the correlation output $C_m(r,\theta)$ can be expressed as

$$\begin{aligned} C_m(r,\theta) &= FT^{-1}\Big[F(\rho,\Theta)H(\rho,\Theta)\Big] \\ &= \sum_{\substack{n=-\infty \\ n\ \text{even}}}^{\infty} \int_0^{\infty}\Big[\int_0^{2\pi} e^{jn\Theta}\,\beta[cos(m\Theta)]\,e^{j\rho r cos(\Theta-\theta)}\,d\Theta\Big] \\ &\quad (-1)^n F_n(\rho)\beta[(-1)^m F_m(\rho)]\,r dr \qquad (V.3.1.11.) \end{aligned}$$

Thus the center correlation value is given as

$$C_m(0,\theta) = \sum_{\substack{n=-\infty \\ n \text{ even}}}^{\infty} \int_0^{\infty} \Big[\int_0^{2\pi} e^{jn\Theta}\, \beta[cos(m\Theta)]\, d\Theta \Big] (-1)^n F_n(\rho) \beta[(-1)^m F_m(\rho)]\, r dr \qquad (V.3.1.12.)$$

But, because $\beta[cos(m\Theta)]$, $cos(n\Theta)$ and $sin(n\Theta)$ are orthogonal, i.e.,

$$\int_0^{2\pi} cos(n\Theta)\beta[cos(m\Theta)]\, d\Theta = 0 \qquad n \neq m;$$
$$\int_0^{2\pi} sin(n\Theta)\beta[cos(m\Theta)]\, d\Theta = 0 \qquad \forall n, m,$$

eq.(V.3.1.12.) can be reduced to

$$C_m(0,\theta) = \int_0^{\infty} \Big[\int_0^{2\pi} |cos(m\Theta)| d\Theta \Big] |F_m(\rho)| r dr. \qquad (V.3.1.13.)$$

Now, the integral enclosed by the brackets in the above equation is identical to

$$\int_0^{2\pi} |sin(m\Theta)| d\Theta$$

Therefore, eq.(V.3.1.13.) can also be written as

$$C_m(0,\theta) = \int_0^{\infty} \Big[\int_0^{2\pi} |sin(m\Theta)| d\Theta \Big] |F_m(\rho)| r dr \qquad (V.3.1.14.)$$

which is exactly the center correlation value that would result if the reference filter is obtained by applying the beta function, $\beta(.)$, to the imaginary component of the Fourier transform of the *m-th* order circular harmonics of f. In view of this identity, we may conclude that practically no information is lost as far as the center correlation value is concerned. When f is not symmetric, all the information associated with its antisymmetric component will be discarded in the process of generating the

BFTCHF. But just as in the procedure of obtaining the binary matched filter, all the phase information accompanying the symmetric component of f is conserved when the BFTCHF is produced.

V.3.2. Computer Simulation

To observe the performance of BFTCHFs in recognizing patterns, digital experiments were carried out. The pattern to be identified in the experiment is the character "A." To generate the reference filters, a computer image of the character "A" was first synthesized. Let $f(l\triangle r, k\triangle\theta)$, $l = 0,\ldots,119$, $k = 0,\ldots,15$, denote the 2-D pattern of "A" in polar coordinates. f is then decomposed into its circular harmonic components. The following discrete modification of eq.(V.3.1.1.) was used to compute the circular harmonic component coefficients:

$$f_m(l\triangle r) = \sum_{k=0}^{119} f(l\triangle r, k\triangle\theta)e^{jmk\triangle\theta}\ \triangle\theta \qquad (V.3.2.1.)$$

where $\triangle\theta = \frac{2\pi}{120}$ and $\triangle r = 1$. Using eq.(V.3.2.1.), the 0th circular harmonic components were obtained. The circular harmonic was then transformed from polar coordinates to cartesian coordinates resulting in an array of 32X32 data points. This array was further expanded into another array of 128X128 data points by patching zeros for the extra data points. The DFT of this circular harmonic in cartesian coordinates was computed. Reference filter was generated according to eq.(V.3.1.9.)(Fig.V.3.2.1a.)

To test the effectiveness of this reference filter, two test patterns were used. The first test pattern contains 4 "A"s in different orientations (Fig.V.3.2.1b.). The

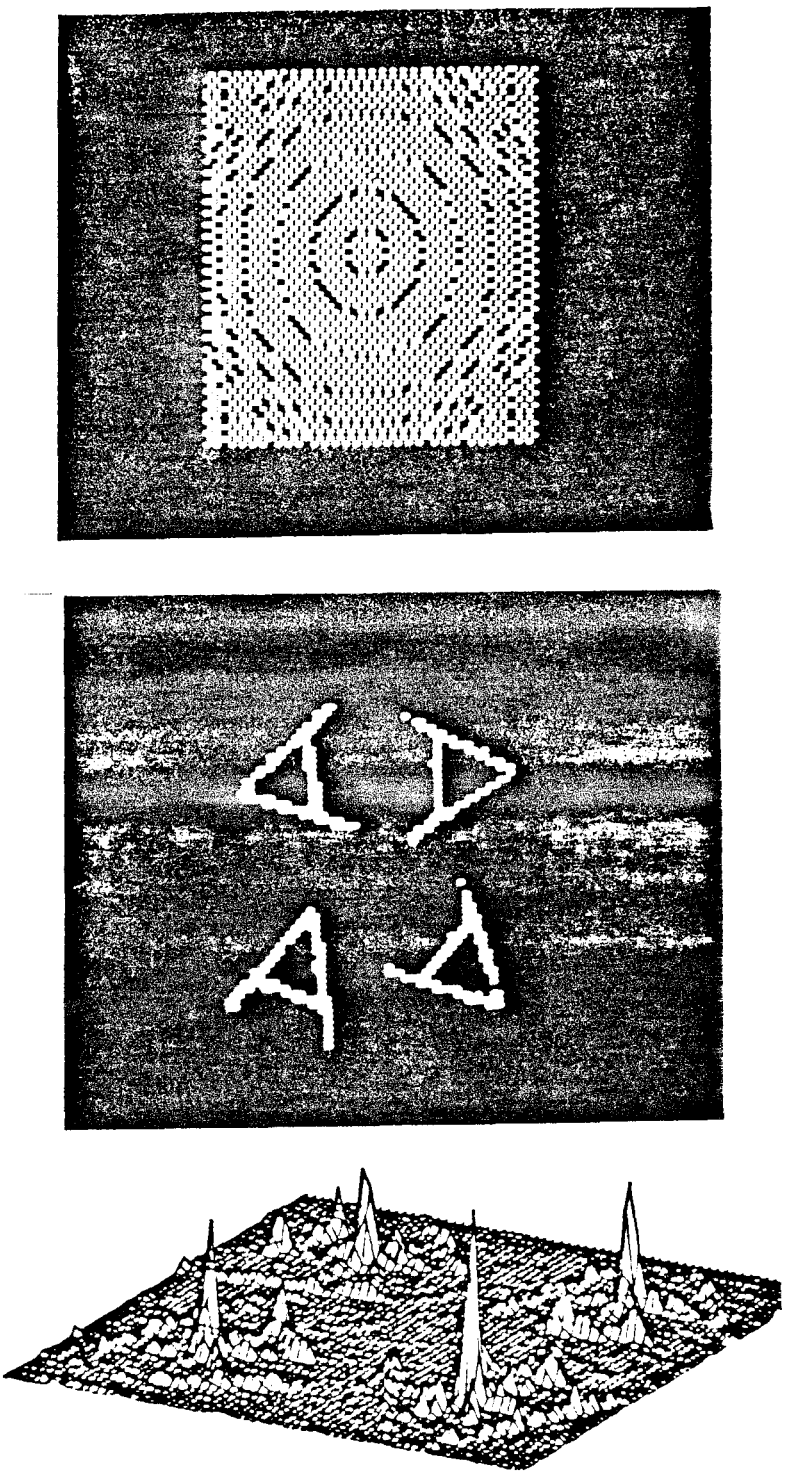
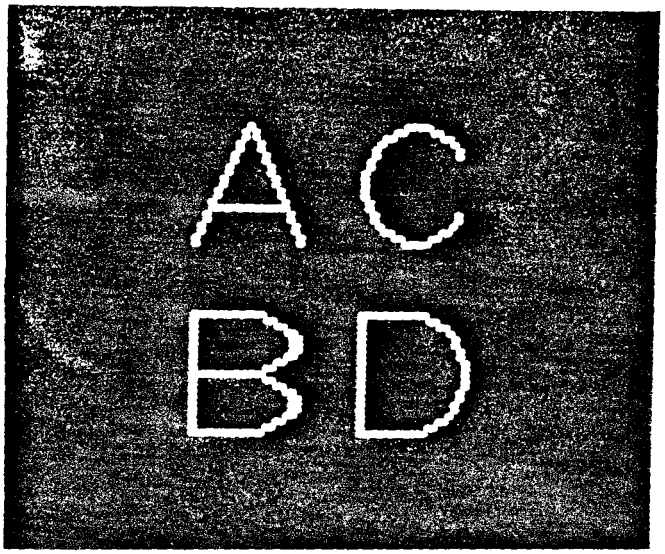


Figure V.3.2.1a,b,c.



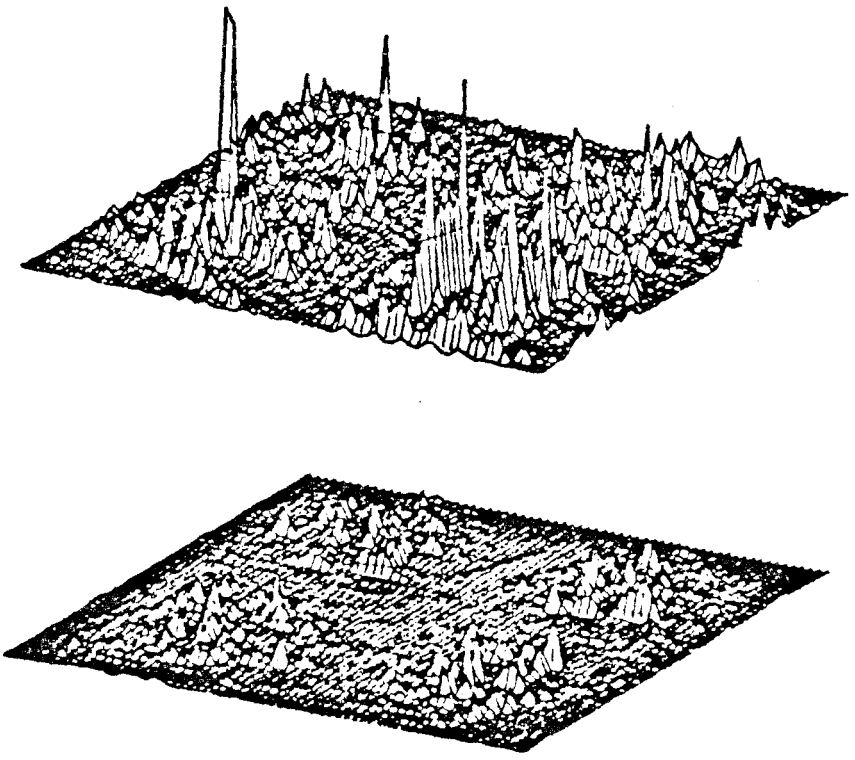


Figure V.3.2.1d,e,f.

second test pattern contains the characters "A," "B," "C" and "D" (Fig.V.3.2.1d.). These two test patterns were correlated with the reference filter (Fig.V.3.2.1a.), respectively. The results were obtained. Shown in Figure V.3.2.1c. is the isometric plot of the correlation output when the input is the first test pattern. Four peaks corresponding to the locations of the differently oriented "A"s are visible. This simulation result demonstrates that the 0-th BFTCHF possesses the rotation and translation invariant recognition capability. The correlation output when the second test pattern is applied to the reference filter is shown in Fig.V.3.2.1e. Besides the highest peak at the left-most corner which is due to the character "A," peaks of smaller magnitudes due to the characters "B" and "C" are also visible. The comparable magnitudes of the peaks hardly qualify the 0-th BFTCHF as an inter-class discriminator. Experiments were repeated with BFTCHF obtained from the 1st and 2nd order circular harmonics. The results were not promising. No peak of outstanding magnitude was visible. Shown in Figure V.3.2.1f is the correlation between test pattern 1 and the 1-st order BFTCHF.

To explain the difference in performance between the 0-th BFTCHF and the BFTCHFs of the other orders, consider the following fact. Suppose $h(r)$ is a 2-D circularly symmetric pattern which is only a function of r. It can be shown that

$$\int_0^{2\pi} \int_0^{\infty} |f(r,\theta) - h(r)|^2 r dr d\theta$$

is a minimum if and only if

$$h(r) = f_0(r)$$

where $f_0(r)$ is the 0-th order circular harmonic of $f(r,\theta)$. In the "closeness" sense of $f_0(r)$ is to $f(r,\theta)$, it is no surprise that 0-th BFTCHF outperforms the other

BFTCHFs. Moreover, unlike the non-zeroth orders, the Fourier transform of the zeroth order circular harmonic is always symmetric and real regardless of the symmetry of $f(r,\theta)$. Thus, when the real part of the Fourier transforms of the circular harmonics are quantized to +1 or -1 to form the reference filters, the 0-th BFTCHF is always the one with the minimum loss in information.

V.3.3. Optical Implementation

An optical correlation between the letter "A" and its 0-th BFTCHF was performed. A Vander Lugt correlator was used. The 0-th BFTCHF was obtained through the use of a computer following the procedure outlined in the previous section. The only deviation was that the initial size of the computer image is 128X128 instead of 32X32. After obtaining the reference filter, it was programmed onto the Litton LIGHT MOD located at the Fourier plane. A negative transparency of the letter "A" of the right size was then placed at the input plane. The correlation output was imaged on a SONY CCD camera and was observed on a TV monitor. The result is shown in Figure V.3.3.1. The bright spot in the middle of the figure is the correlation peak. The lines are the off-center correlation structure. To verify the rotation invariant property of the reference filter, the input image was rotated. It was observed that the lines of the correlation rotated in the same direction and at the same speed as the input image. It was also observed that the rotations of the lines were about the correlation peak which remained stationary.

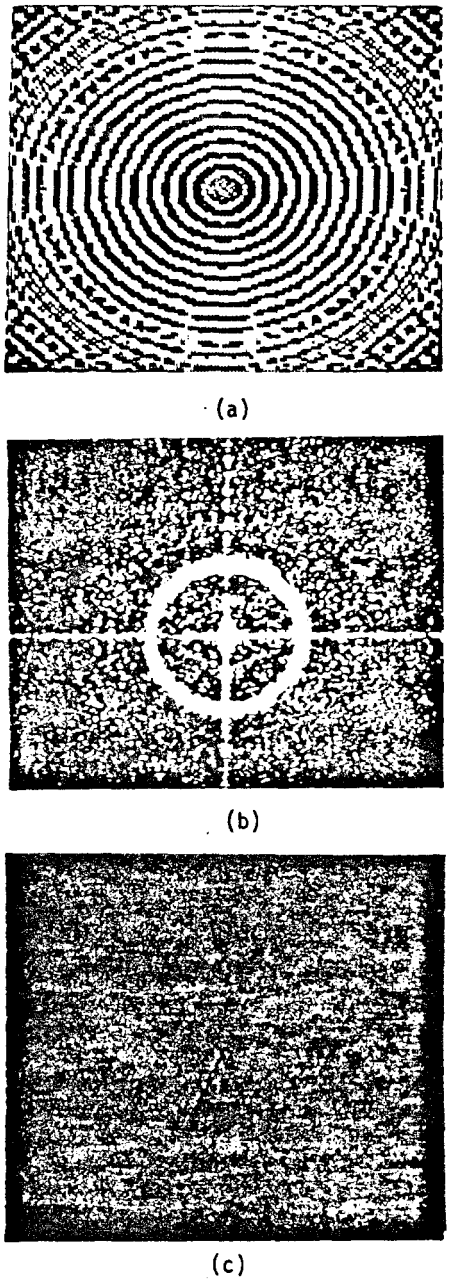


Figure V.3.3.1. The cross-correlation of the letter "A" and its 0-*th* order BFTCHF.

References for Chapter V

1. R. Sprague and C. Koliopoulos, "Time Integrating Acoustooptic Correlator," *Appl. Opt.*, **15**, 89 (1976).
2. W. Rhodes, "Acousto-Optic Signal Processing: Convolution and Correlation," *Proc. IEEE*, **Vol. 69**, 65 (1981).
3. D. Psaltis, "Incoherent Electro-Optic Image Correlator," *Opt. Engr.*, **23**, 12 (1984).
4. D. Psaltis, "New Optical Transforms for Pattern Recognition," *Proc. IEEE*, **65**, 77 (1977).
5. J. Duthie and J. Upatnieks, "Compact Real-Time Coherent Optical Correlators," *Opt. Engr.*, **23**, 7 (1984).
6. B. Guenther, C. Chritensen, and J. Upatnieks, "Coherent Optical Processing: Another Approach," *IEEE J. of Q.E.*, **QE-15**, 1348 (1979).
7. D. Psaltis, "Two-Dimensional Optical Processing Using One-dimensional Input Devices," *Proc. IEEE*, **72**, 962 (1984).
8. D. Psaltis, "Optical Image Correlation using Acoustooptic and Charge-Coupled Devices," *Appl. Opt.*, **21**, 491 (1982).
9. D. Psaltis, E. Paek, and S. Vankatesh, "Optical Image Correlation With a Binary Spatial Light Modulator," *Opt. Eng.*, **23**, 698 (1984).
10. A. Korpel, "Acousto-Optics — A Review of Fundamentals," *Proc. IEEE*, **69**, 48 (1981).
11. I. Chang,"1. Acousto-optic Devices and Applications," *IEEE Trans. Son. Ult.*, **SU-23**, 2 (1976).
12. K. Wagner and D. Psaltis, "A Space Integrating Acousto-Optic Matrix-Matrix

Multiplier," *Opt. Comm.,* **52**, 173 (1984).

13. F. Mok, J. Diep, H. Liu, D. Psaltis, "Real-Time Computer-Generated Hologram By Means of Liquid-Crystal Television Spatial Light Modulator," *Opt. Letters*, **11**, 748 (1986).

14. J. Hall, "Arrays and Charge-Coupled Devices," Ch.8, in "Applied Optics and Optical Engineering," Vol. VIII, R. Shannon and J. Wyant Eds., Academic Press, New York (1980).

15. C. Hester, and D. Casasent,"Multi-variant Technique for Multi-Class pattern Recognition," *Appl. Opt.* , **19**, 1758 (1980).

16. J. Goodman, "Introduction to Fourier Optics," McGraw-Hill, New York (1968).

17. D. Casasent, "Coherent Optical Pattern Recognition," *Proc. IEEE*, **67**, 813 (1979).

18. G. Turin, "An Introduction to Matched Filters," *IRE Trans. Inform. Theory*, **IT-6**, 311(1960).

19. D. Psaltis, F. Mok, E. Paek,"On The Use Of The Litton Magneto-Optic Device in Optical Processors," *Proc. SPIE*, **6**, 465 (1984).

20. W. Bleha, L. Lipton, E. Wiener-Arnear, J. Grinberg, P. Reif, D. Casasent, H. Brown, and B. Markevitch, "Application of the Liquid Crystal Light Valve to Real-Time Optical Data Processing," *Opt. Eng.* , **17**, 371 (1978).

21. B. Horowitz and F. Corbett,"The PROM — Theory and Applications for the Pockels Readout Optical Modulator," *Opt. Eng.* , **17**, 353 (1978).

22. C. Warde, A. Weiss, A. Fisher, and J. Thachara, "Optical Information Processing Characteristics of the Microchannel Spatial Light Modulator," *Appl. Opt.* , **20**, 2066 (1981).

23. W. Colburn and E. Tompkins, "Improved Thermoplastic Photoconductor Devices for Holographic Recording," *Appl. Opt.* , **13**, 2934 (1974).

24. Y. Hsu and H. Arsenault, "Optical Pattern Recognition Using Circular Harmonic Expansion," *Appl. Opt.* , **22**, 4016 (1982).

25. H. Arsenault and Y. Hsu, "Rotation-Invariant Discrimination between almost Similar Objects," *Appl. Opt.* , **22**, 130 (1983).

26. Y. Yang, K. Chalasinska-Macukow, and H. Arsenault, "Digital and Optical Analysis of the Optimum Circular Symmetrical Filter in a Character Recognition System," *Optica Acta*, **30**, 189 (1983).

27. Y. Neng and H. Arsenault, "Pattern Discrimination by Multiple Circular Harmonic Components," *Appl. Opt.* , **23**, 841 (1984).

28. E. Paek and S. Lee, "Discrimination Enhancement in Optical Pattern Recognition by using a Modified Matched Filter," *Can. J. Phys.* , **57**, 1335 (1979).

29. A. Lohmann and H. Werlich, "Incoherent Matched Filtering with Fourier Holograms," *Appl. Opt.* , **10** 670 (1971).

30. A. Vander Lugt, "Signal Detection by Complex Spatial Filtering", *IEEE Trans Inf Th,* **IT-10**, 139 (1964).

31. A. W. Lohmann and D. P. Paris, "Computer Generated Spatial Filters for Coherent Optical Data Processing," *Appl. Opt.* , **7**, 651 (1968).