92-32
11975

N95- 32430

The Composite Classification Problem in Optical Information Processing

Final Report
NASA/ASEE Summer Faculty Fellowship Program—1994
Johnson Space Center

Prepared By:	Eric B. Hall, Ph.D.
Academic Rank:	Assistant Professor
College and Department:	Southern Methodist University Dept. of Electrical Engineering Dallas, Texas 75275
NASA/JSC	
Directorate:	Engineering
Division:	Tracking and Communications
Branch:	Tracking Techniques
JSC Colleague:	Richard D. Juday
Date Submitted:	August 5, 1994
Contract Number:	NGT-44-005-803

ABSTRACT

Optical pattern recognition allows objects to be recognized from their images and permits their positional parameters to be estimated accurately in real time. The guiding principle behind optical pattern recognition is that a lens focusing a beam of coherent light modulated with an image produces the two-dimensional Fourier transform of that image. When the resulting output is further transformed by the matched filter corresponding to the original image, one obtains the autocorrelation function of the original image, which has a peak at the origin. Such a device is called an optical correlator and may be used to recognize and locate the image for which it is designed. (From a practical perspective, an approximation to the matched filter must be used since the spatial light modulator (SLM) on which the filter is implemented usually does not allow one to *independently* control both the magnitude and phase of the filter.) Generally, one is not just concerned with recognizing a single image, but is instead interested in recognizing a variety of rotated and scaled views of a particular image. In order to recognize these different views using an optical correlator, one may select a subset of these views (whose elements are called training images) and then use a composite filter that is designed to produce a correlation peak for each training image. Presumably, these peaks should be sharp and easily distinguishable from the surrounding correlation plane values. In this report we consider two areas of research regarding composite optical correlators. First, we consider the question of how best to choose the training images that are used to design the composite filter. With regard to quantity, the number of training images should be large enough to adequately represent all possible views of the targeted object yet small enough to ensure that the resolution of the filter is not exhausted. As for the images themselves, they should be distinct enough to avoid numerical difficulties yet similar enough to avoid gaps in which certain views of the target will be unrecognized. One method that we introduce to study this problem is called probing and involves the creation of artificial imagery. The second problem we consider involves the classification of the composite filter's correlation plane data. In particular, we would like to determine not only whether or not we are viewing a training image, but, in the former case, we would like to determine *which* training image is being viewed. This second problem is investigated using traditional M-ary hypothesis testing techniques.

A Review of Quadratic Classifiers Consider a random vector X taking values in $\mathbb{R}^d$ with a multivariate[1] Gaussian density function

$$f(x) = \frac{1}{(2\pi)^{d/2}\sqrt{|\Sigma|}} \exp\left(-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)\right) \tag{1}$$

where $\mu = \mathrm{E}[X] = [\mu_i]_{i=1}^d$ and $\Sigma = \mathrm{E}[(X-\mu)(X-\mu)^\top] = [\sigma_{ij}]_{i,j=1}^d$. Note that this distribution is completely characterized by $d + \frac{1}{2}d(d+1)$ parameters. We sometimes denote this distribution by $\mathrm{N}(\mu,\, \Sigma)$ where the covariance matrix Σ is a symmetric, nonnegative definite matrix. In writing expression (1), we have assumed that Σ is in fact *positive* definite, and hence invertible. If, instead, Σ is singular then $\alpha^\top X = 0$ a.s. for some nonzero vector α from $\mathbb{R}^d$. In such a case, we say that the distribution of X is degenerate. If $\mathrm{E}[|\alpha^\top X|^2] > 0$ for each nonzero $\alpha \in \mathbb{R}^d$ then the covariance matrix Σ is positive definite. The nonnegative square root of the value $(x-\mu)^\top\Sigma^{-1}(x-\mu)$ in the exponent of (1) is called the Mahalanobis distance from x to the mean μ. Note that a set of points with equal Mahalanobis distance to the mean forms a hyperellipsoid in $\mathbb{R}^d$.[2]

Our interest lies with the *a posterior* density function

$$p(t_i|x) = \frac{p(x|t_i)P(t_i)}{p(x)}$$

where t_i denotes an image with an *a priori* probability of $P(t_i)$. After taking the natural log of the *a posterior* density function and neglecting the terms that do not change with i,[3] we obtain a discriminant function of the form $g_i(x) = \ln p(x|t_i) + \ln P(t_i)$. Assume now that $p(x|t_i)$ is a multivariate Gaussian density function with mean vector μ_i and covariance matrix Σ_i. In this case, it follows that

$$g_i(x) = -\frac{1}{2}(x-\mu_i)^\top\Sigma_i^{-1}(x-\mu_i) - \frac{d}{2}\ln(2\pi) - \frac{1}{2}\ln|\Sigma_i| + \ln P(t_i). \tag{2}$$

If $\Sigma = \sigma^2\mathrm{I}$, then equation (2) reduces to

$$g_i(x) = \frac{-\left\|x-\mu_i\right\|^2}{2\sigma^2} + \ln P(t_i) \tag{3}$$

[1]The components of X are said to be jointly or mutually Gaussian. Whereas a sum of Gaussian random variables need not be Gaussian, and uncorrelated Gaussian random variables need not be independent, these useful properties do hold when the random variables are jointly Gaussian.

[2]For more information concerning the topics in this section, see [3] or [4].

[3]Throughout this discussion the g_i's are equal up to additive, constant functions of i and/or multiplicative, positive, constant functions of i.

where $\|x-\mu_i\|^2 \equiv (x-\mu_i)^\top(x-\mu_i)$. Note that if the *a priori* probabilities are equal then this test assigns x to the category that has the nearest mean, where distance is determined using the previous norm. This type of classifier is called a *minimum distance classifier.* However, if we expand g_i a step further (retaining all but the $x^\top x$ term), equation (3) reduces to

$$g_i(x) = \alpha_i^\top x + \beta_i \tag{4}$$

where $\alpha_i = \mu_i/\sigma^2$ and $\beta_i = -\mu_i^\top \mu_i/2\sigma^2 + \ln P(t_i)$. A test of this form is called a *correlator detector* or *linear detector.* As indicated by equation (4), the decision boundaries induced by equation (3) are hyperplanes.

Next, consider the case in which Σ_i is equal to some constant matrix Σ for all i. In this case, equation (2) reduces to

$$g_i(x) = -\frac{1}{2}(x-\mu_i)^\top \Sigma^{-1}(x-\mu_i) + \ln P(t_i).$$

If the *a priori* probabilities are equal then this test assigns x to the distribution whose mean has the smallest Mahalanobis distance to x. Expanding further, however, we again obtain a test function of the form

$$g_i(x) = \alpha_i^\top x + \beta_i$$

where this time $\alpha_i = \Sigma^{-1}\mu_i$ and $\beta_i = -\frac{1}{2}\mu_i^\top \Sigma^{-1}\mu_i + \ln P(t_i)$. Note that again we have obtained a linear classifier and that our decision boundaries are hyperplanes.

In the general case, equation (2) may be written as

$$g_i(x) = x^\top A_i x + \alpha_i^\top x + \beta_i$$

where $A_i = -\frac{1}{2}\Sigma_i^{-1}$, $\alpha_i = \Sigma_i^{-1}\mu_i$, and $\beta_i = -\frac{1}{2}\mu_i^\top \Sigma_i^{-1}\mu_i - \frac{1}{2}\ln|\Sigma_i| + \ln P(t_i)$. Note that this test function is quadratic and that the corresponding decision boundaries are hyperquadrics.

The Classification Problem & Bayesian Inference Consider N_0 classes of objects denoted by $\omega_0, \ldots, \omega_{N_0}$. Assume that object class ω_k contains N_k training images denoted by $T_{1k}, T_{2k}, \ldots, T_{N_k k}$. Let $N_T = \sum_{k=1}^{N_0} N_k$ denote the total number of training images. The standard classification problem seeks a partition of the "signal space" into N_T classification regions denoted by $R_1, \ldots, R_{N_T}$. The *composite classification problem,* on the other hand, seeks a partition of the "signal space" into N_0 regions denoted by $C_1, \ldots, C_{N_0}$; that is, each region corresponds to a different object class where each object class can contain many different training images.

Let N_F denote the number of composite filters that are formed from the training images, and let N_M denote the number of "features" that are required by the chosen correlation metric. Then, the dimension n of our signal space is given by $N_F N_M$. (In this context, a "signal" corresponds to the output of our composite filter. Our goal is to classify these outputs into their corresponding object classes. The goal of the standard classification problem would be to map each possible output to the training image that produced it. Note that, for the moment, we are assuming that only training images are available as inputs to our system.) Our goals, once we develop our test, will be to choose the training image groups and composite filters so that (1) the distributions of the test statistics can be estimated accurately with as few samples as possible, and, (2) the distributions of the test statistics do not significantly overlap. (The first goal has pragmatic motivations, and the second goal reflects the standard desire that our test be as close to singular as possible.)

Let X denote our signal; that is, let X be a random vector taking values in $\mathbb{R}^{N_M N_F}$ that represents the N_M outputs of each of the N_F filters. Let $p_{T_{ik}}(x)$ be a probability density function for X when the training image T_{ik} is used as the input. Let Π_{ik} denote the *a priori* probability that the input to our system is training image T_{ik}. We will use these *a priori* probabilities to develop a Bayesian test. One assumption at this point is, of course, that these probabilities both exist and are known. Further, we will assume that the sum of the Π_{ik}'s over all possible training images is unity. That is, as mentioned above, we will continue to assume that only training images are input into our system. Based upon these assumptions we have the following expression for a probability density function for the output of our system:

$$p_T(x) = \sum_{k=1}^{N_0} \sum_{i=1}^{N_k} \Pi_{ik} p_{T_{ik}}(x).$$

Note that this density is only exact if our inputs are exclusively training images. In a more general setting in which our inputs need not be training images we would hope that this density would be a close approximation of the true density of the output. (An interesting problem would be to investigate the behavior of this approximation as $N_T \to \infty$.)

The Standard Classifier For a standard classifier, we let $N_0 = N_T$; that is, our object classes are singleton sets each containing a single training image. In this case, we will denote T_{ik} by T_k and Π_{ik} by Π_k. According to the usual Bayes formula, we have

$$p(T_k|X = x) = \frac{p_{T_k(x)}\Pi_k}{p_T(x)}$$

where $p(T_k|X = x)$ denotes the conditional probability that T_k was input given that we observed x. Our Bayesian hypothesis test then is to assign x to T_k if and only if $p(T_k|X = x) \geq p(T_i|X = x)$ for all $i = 1, \ldots, N_T$. (That is, we choose the training image corresponding to a maximum *a posterior* distribution.)

The Composite Classifier For a composite classifier, N_0 is less than (usually *much* less than) N_T. Let $p_{\omega_k}(x)$ be a probability density function for X when the input is a training image from the object class ω_i. Let Λ_k be the *a priori* probability that the input will be from class ω_i. (Yet again, we are assuming that the input will always be a training image.) Note that

$$\Lambda_k = \sum_{j=1}^{N_k} \Pi_{jk}.$$

Also, note that

$$p_{\omega_k}(x) = \frac{1}{\Lambda_k} \sum_{j=1}^{N_k} p_{T_{jk}}(x)\Pi_{jk},$$

and further, using Bayes' formula, note that

$$p(\omega_k|X = x) = \frac{p_{\omega_k}(x)\Lambda_k}{p_T(x)} = \frac{1}{p_T(x)} \sum_{j=1}^{N_k} p_{T_{jk}}(x)\Pi_{jk} \tag{5}$$

where $p(\omega_k|X = x)$ is the conditional probability that the input is from class ω_k given that x is observed. Our test in this case assigns x to object class ω_k if and only if $p(\omega_k|X = x) \geq p(\omega_j|X = x)$ for all $j = 1, \ldots, N_0$. That is, we assign x to class ω_k if and only if

$$\frac{p(\omega_k|X = x)}{p(\omega_j|X = x)} = \frac{\sum_{i=1}^{N_k} p_{T_{ik}}(x)\Pi_{ik}}{\sum_{i=1}^{N_j} p_{T_{ij}}(x)\Pi_{ij}} \geq 1$$

for all $j = 1, \ldots, N_0$.

We will now assume that $p_{T_{jk}}(x)$ is an n-variate Gaussian density function with mean vector m_{jk} and covariance matrix C_{jk} where we recall that $n = N_M N_F$. Note that, in this case, $p_T(x)$ and $p_{\omega_k}(x)$ are *mixtures* of multivariate Gaussian densities, which, of course, can be far from Gaussian. Several problems present themselves at this point. First, the rather rueful distribution of X does not bode well for analytic solutions.[4] Second, the parameters m_{jk} and C_{jk} are rarely known and hence often must be estimated. These estimates may then substituted into the test given above. Unfortunately, when this substitution is done, our test is generally no longer Bayesian, and hence, need no longer satisfy any desired optimality property such as minimum probability of error.[5] Although the test statistic is difficult to work with, one possible

[4]To be precise, it is only our *approximation* of the distribution of X that is a mixture of Gaussian distributions unless we assume that our inputs will only be training images. Of course, there is no reason to expect that the true distribution of X is any less rueful than our approximation of that distribution.

[5]The likelihood ratio corresponding to such a test is sometimes said to be a *generalized likelihood ratio*, particularly when the unknown parameters are replaced with maximum likelihood estimates.

simplification involves coordinate transformations that simplify the calculation of the Mahalanobis distances in the exponents.

TRAINING IMAGES

In the previous section, we assumed that the input to our system was always a training image, that the output of our system given that the input was a training image was Gaussian, and that the output of our system given that the input was a training image from a particular object class was equal in distribution to a mixture of Gaussian distributions.

There are two types of non-training images. A *first category non-training image* is an unknown view of a known object. Ideally, this type of image will be close enough to an appropriate training image so that their distributions will have significant overlap. A *second category non-training image* is an image of an object that is not intended to be recognized. Ideally, this type of image should be far away from the training images so that its distribution will not have significant overlap with that of any training image.

Of course, as we approach the ideal situation, we are generally going to need an ever increasing supply of training images. While a larger number of training images would be helpful when the input is a non-training image, it would also increase the complexity of our system and it would decrease the performance when the input actually is a training image. What we need is: (1) a method of determining how many training images we need to meet some desired goal, and (2) a method of obtaining appropriate additional training images when such images are required. In the next section, we will consider both of these problems.

Probing In [2] the term *probing* is introduced and used to describe the creation of artificial images to improve and analyze pattern recognition algorithms. A deterministic pattern recognition algorithm is simply a function mapping the set of all possible images to some decision set. Ideally, one would choose this function by considering each possible image in turn and determining for each the appropriate decision that should be made if that image appears as the input to our system. Unfortunately, however, such a design procedure is generally not tractable due to the enormous number of possible input scenes. (For example, there are over $10^{39,000}$ different 128 by 128 pixel input images with 256 gray scales.) It is at this point that probing becomes useful since it allows one to intelligently sample this enormous image space in order to select images for which a pattern recognition scheme can be designed.

We have identified three different methods of probing that appear promising with regard to optical information processing. First, probing can be used to form a "map" of the image space. As an example, consider two images I_1 and I_2 from the image space consisting of all $N \times N$ pixel images composed of M gray scale levels. (Note that such an image can be viewed as a point in the set $\{0,1,\ldots,M-1\}^{N^2}$.) For each value of $p \in [0,1]$, let V_p be a random object taking values in $\{0,1,\ldots,M-1\}^{N^2}$ such that each pixel value in V_p is equal with probability p to the corresponding pixel value in I_2 and is equal with probability $1-p$ to the corresponding pixel value in I_1.[6] (Note that $V_0 = I_1$ and $V_1 = I_2$. Note also that if a particular pixel in I_1 and I_2 agree then they also agree with the corresponding pixel in V_p. Thus, the pixel variance in V_p is only positive for pixels at which I_1 and I_2 disagree.) As p increases from 0 to 1,[7] our decision algorithm (when applied to V_p) will on average no longer recognize I_1 at some point p_1 and will begin instead to recognize I_2 at some point p_2. These average values of p allow us to determine when images are "adjacent" and to recognize the possible existence of a "hole" between I_1 and I_2. A map of this sort allows us to distinguish between two procedures that perform the same when the inputs are always training images. Further, we could possibly use a realization of V_p where $p_1 < p < p_2$ to fill such a hole in our set of training images. Of course, different realizations of V_p for some fixed choice of $p \in (0,1)$ could be quite different. (A similar procedure is described in [7] where sections of training images are randomly selected and weighted to form what is called a synthetic reference object.)

Second, probing can be used to measure the robustness of a pattern recognition algorithm. For a training image T and a nonnegative value θ, let U_θ be an image (i.e. a random object taking values in $\{0,1,\ldots,M-1\}^{N^2}$) such that each pixel value in U_θ has a mean equal to the corresponding pixel value in T and has a variance equal to θ. Note that $U_0 = T$ a.s. The average value of θ at which T is no longer recognized provides an indication of the robustness of our algorithm with respect to that particular training image. Presumably this value should be large and should not vary widely among the different training images. Tests of this sort allow us to distinguish between two procedures that perform identically when the inputs are always training images.

Third, probing can be employed in which the pixel values in the synthesized image are *not* statistically independent. By introducing some sort of spatially localized dependence, we can analyze a situation in which a pixel in the synthesized image is more likely to be from a particular training image if its neighboring pixel values are also from that training image.

[6]In addition, one could further perturb the image by choosing for some given gray scale value M_0, a realization of a random variable with a unimodal distribution on $\{0,1,\ldots,M-1\}$ centered at M_0.

[7]Notice that from an information theoretic standpoint, the "uncertainty" is maximized when $p = 1/2$.

Synthetic Discriminant Functions The synthetic discriminant function approach uses a linear combination of training images to create a composite image that is cross correlated with the input to the system. The weights in the linear combination are chosen so that the cross correlation at the origin is the same for all training images from the same class. (The resulting filter is sometimes called an Equal Correlation Peak (ECP) Synthetic Discriminant Function (SDF).) For example, if we have N training images $s_1(x,y),\ldots,s_N(x,y)$, then our composite image would be of the form

$$h(x,y) = \alpha_1 s_1(x,y) + \cdots + \alpha_N s_N(x,y),$$

and the α_i's would be chosen so that

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x,y)s_i(x,y)\,dx\,dy = c_i$$

for $i = 1,\ldots,N$ where the c_i's are preselected constants. Modifications of the standard SDF approach exist that impose other constraints. For example, Minimum Variance SFD (MVSDF) minimizes the output noise variance and the Minimum Average Correlation Energy (MACE) filter attempts to produce sharp correlation peaks at the origin of the output.

Let $H(u,v)$ denote the Fourier transform of the composite image $h(x,y)$; that is, let

$$H(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(x,y)e^{-j2\pi(xu+yv)}\,dx\,dy.$$

Note that

$$h(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} H(u,v)e^{j2\pi(xu+yv)}\,du\,dv.$$

Also, let S_i denote the Fourier transform of the ith training image s_i for $i = 1,\ldots,N$. Further, for an element x from $\mathbb{R}^N$ or $\mathbb{C}^N$ let $\tilde{x}$ denote the corresponding element in $\mathbb{R}^{N\times N}$ or $\mathbb{C}^{N\times N}$ that is obtained by placing x along the diagonal and 0 elsewhere. That is, let

$$\tilde{x}_{ij} = \begin{cases} x_i & \text{if } i = j \\ 0 & \text{if } i \neq j. \end{cases}$$

Finally, assume that the input to our system is corrupted with an additive, zero mean, wide sense stationary noise process $N(x,y)$ with power spectral density $P_N(u,v)$. That is,

$$P_N(u,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathrm{E}[N(x+\tau, y+\lambda)N^*(x,y)]\,e^{-j2\pi(\tau u+\lambda v)}\,d\tau\,d\lambda$$

where the asterisk denotes the complex conjugate operation. We will now list several popular performance criteria.

1. **The Output Noise Variance**

$$\mathrm{ONV} = H^{*}(u,v)^{\mathsf{T}} P_N(u,v) H(u,v)$$

2. **The Average Similarity Measure**

$$\mathrm{ASM} = H^{*}(u,v)^{\mathsf{T}} \left[\frac{1}{N} \sum_{i=1}^{N} \left(\tilde{S}_i(u,v) - M_S(u,v) \right)^{*} \left(\tilde{S}_i(u,v) - M_S(u,v) \right) \right] H(u,v)$$

$$\text{where } M_S(u,v) = \frac{1}{N} \sum_{i=1}^{N} \tilde{S}_i(u,v)$$

3. **The Average Correlation Energy**

$$\mathrm{ACE} = H^{*}(u,v)^{\mathsf{T}} \left[\frac{1}{N} \sum_{i=1}^{N} \tilde{S}_i^{*}(u,v) \tilde{S}_i(u,v) \right] H(u,v)$$

4. **The Average Correlation Height**

$$\mathrm{ACH} = \left| \frac{1}{N} \sum_{i=1}^{N} H(u,v)^{\mathsf{T}} S_i(u,v) \right|$$

Topics for Further Research If the previous performance criteria were our only concern then our goal in choosing h would be to minimize ONV, ACE, and ASM, and to maximize ACH. Some immediate questions that arise are:

1. Is it possible to optimize any or all of these parameters simultaneously? (In general, the answer is no for the performance criteria listed above. However, this question and those that follow should be considered whenever additional performance criteria are included.)

2. If not, then can we fix one or more of the parameters and then optimize those that remain? Note that a similar procedure is used to obtain a Neyman–Pearson test. Since it is not generally possible to maximize the power and minimize the size of a test, a Neyman–Pearson test maximizes the power while keeping the size constant. (The proof of this result follows from standard techniques of variational calculus.)

3. There are numerous paradoxes that arise in the Neyman–Pearson theory. Do similar difficulties arise here? For example, in a Neyman–Pearson test, strange results can occur if the false alarm rate is chosen to be large in a test that is almost singular. Conversely, situations exist in which the power exceeds the false alarm rate by an arbitrarily small value.

4. An *optimal trade-off filter* is obtained by fixing all but one of the parameters and optimizing the other. Is this necessary? Might it be possible to fix fewer parameters and then optimize those that remain?

Minimum Euclidean Distance Optimal Filters Our goal in this section is to extend the results in [5] to include composite filters. In particular, we will first seek an algorithm by which the output of the MEDOF algorithm can be classified by statistical inference into training image classes. The following steps follow the procedure suggested in [1]:

1. Separate the training images into object classes. The first step is to select the training images and object classes. Each object class should correspond to a different object that we wish to recognize. The training images within each object class should be chosen to adequately describe the different expected orientations of the object they represent.

2. Create the filters by which these training images will be distinguished. One way in which this step could be achieved would be to create a composite image from each object class based upon the training images in that object class. These composite images could then be used to create composite filters whose combined outputs would comprise the components of our output random vector X. (That is, we would let $N_F = N_0$.)

3. Estimate the mean and covariance matrix for each class. We have assumed that the output of our system given that the input is a specific training image T_{jk} will be multivariate Gaussian with mean m_{jk} and covariance matrix C_{jk}. We will estimate these parameters via standard techniques. In particular, if we observe $x_1, \ldots, x_N$ when training image T_{jk} is our input then we will estimate m_{jk} via

$$\widehat{m}_{jk} = \frac{1}{N} \sum_{i=1}^{N} x_i,$$

and we will estimate C_{jk} via

$$\widehat{C}_{jk} = \frac{1}{N-1} \sum_{i=1}^{N} \left(x_i x_i^{\mathsf{T}} - \widehat{m}_{jk} \widehat{m}_{jk}^{\mathsf{T}} \right).$$

4. Calculate the generalized weighted Gaussian sum for each class. The Bayes test consists in selecting the object class for which equation (5) is largest. In our case, however, we must substitute the estimates we just obtained in place of the true parameters that are unknown. Thus, for each object class ω_k from the set of N_0 different object classes, we calculate the ratio

$$\frac{\sum_{j=1}^{N_k} \Pi_{jk} |\widehat{C}_{jk}|^{-1/2} \exp\left(-\frac{1}{2}(x - \widehat{m}_{jk})^\top \widehat{C}_{jk}^{-1}(x - \widehat{m}_{jk})\right)}{\sum_{i=1}^{N_0} \sum_{l=1}^{N_k} \Pi_{il} |\widehat{C}_{il}|^{-1/2} \exp\left(-\frac{1}{2}(x - \widehat{m}_{il})^\top \widehat{C}_{il}^{-1}(x - \widehat{m}_{il})\right)}. \tag{6}$$

5. Choose the class with the largest weighted sum. We select the object class for which the corresponding term found in expression (6) is the largest. Our test then announces that the input to our system belongs to this object class.

Caveats

1. The calculation of expression (6) is very computationally intensive. This problem can be lessened somewhat by appropriate coordinate transformations.
2. The insertion of the estimates in place of the true parameters may generally be expected to remove any optimality condition that the original test was designed to satisfy.
3. The procedure is based upon an initial assumption of normality that may or may not be justified. In particular, an important concern is the robustness of our test with regard to perturbations in the underlying distributions. Also, what modifications would be required in order to remove the assumption of normality?
4. The procedure requires one to possess the *a priori* probability associated witth each training image. These probabilities are generally not known and hence must be either estimated or assumed. Again, the robustness of our procedure with respect to these values is an important concern.

Conclusion

We have presented an overview of our research in two areas of optical pattern recognition. First, we have considered the use of probing to map the image space and to measure the robustness of an optical correlator with respect to deviations in the input from training images. Second, we have considered the use of Bayesian inference in the design of composite correlators in which images are assigned to object classes consisting of a collection of training images.

References

[1] Bret F. Draayer, Gary W. Carhart, and Michael K. Giles, "Optimum classification of correlation-plane data by Bayesian decision theory," *Applied Optics,* Vol. 33, No. 14, May 10, 1994, pp. 3034–3049.

[2] Kenneth Augustyn, "A new approach to automatic target recognition," *IEEE Transactions on Aerospace and Electronic Systems,* Vol. 28, No. 1, January 1992, pp. 105–114.

[3] Richard O. Duda and Peter E. Hart, *Pattern Classification and Scene Analysis,* Wiley–Interscience, New York, 1973.

[4] Keinosuke Fukunaga, *Introduction to Statistical Pattern Recognition,* Second edition, Academic Press, Boston, 1990.

[5] Richard D. Juday, "Optimal realizable filters and the minimum Euclidean distance principle," *Applied Optics,* Vol. 32, No. 26, September 10, 1993, pp. 5100–5111.

[6] Bahram Javidi and Joseph L. Horner, eds., *Real–Time Optical Information Processing,* Academic Press, Boston, 1994.

[7] Jerome Knopp, "Spatial weighting of synthetic reference objects for joint correlation by random pixel mixing," *Proceedings of SPIE,* Orlando, Florida, April 16–18, 1990, Vol. 1295, Real–Time Image Processing II, pp. 138–145.

93-31
4476

N95-32431

THE BACKGROUND AND THEORY OF INTEGRATED RISK MANAGEMENT

Final Report
NASA/ASEE Summer Faculty Fellowship Program - 1994
Johnson Space Center

Prepared by:	John L. Hunsucker, Ph.D., P.E.
Academic Rank:	Associate Professor
College and Department:	University of Houston Department of Industrial Engineering

NASA/JSC

Program:	Space Station
Office:	Integrated Risk Management
JSC Colleague:	James E. Van Laak
Date Submitted:	August 5, 1994
Contract Number:	NGT-44-005-803