A Maximum-likelihood Approach for Multiresolution W-operator Design

Daniel André Vaquero, Junior Barrera, Roberto Hirata Jr.
USP–Universidade de São Paulo
IME–Instituto de Matemática e Estatística
Rua do Matão, 1010 - Cidade Universitária
CEP: 05508-090, São Paulo, SP, Brasil
{daniel, jb, hirata}@vision.ime.usp.br

Abstract

The design of W-operators from a set of input/output examples for large windows is a hard problem. From the statistical standpoint, it is hard because of the large number of examples necessary to obtain a good estimate of the joint distribution. From the computational standpoint, as the number of examples grows memory and time requirements can reach a point where it is not feasible to design the operator. This paper introduces a technique for joint distribution estimation in W-operator design. The distribution is represented by a multiresolution pyramidal structure and the mean conditional entropy is proposed as a criterion to choose between distributions induced by different pyramids. Experimental results are presented for maximum-likelihood classifiers designed for the problem of handwritten digits classification. The analysis shows that the technique is interesting from the theoretical point of view and has potential to be applied in computer vision and image processing problems.

1. Introduction

The design of W-operators from a set of input/output examples for large windows is a hard problem. From the statistical standpoint, it is hard because of the large number of examples necessary to obtain a good estimate of the joint probability distribution. From the computational standpoint, as the number of examples grows memory and time requirements can reach a point where it is not feasible to design the operator. Some approaches have been proposed to obtain estimates of the distribution for configurations that don't appear in the training samples. From previous works in the field, the most successful one is the multiresolution design [6, 11]. This method uses a pyramidal structure to represent the designed operators, and performs conditional distribution estimation by using data observed at lower resolutions when there is not enough data in the samples to obtain a good estimate in a higher resolution.

In this work, we introduce an algorithm that uses the pyramidal framework from previous methods to estimate the joint probability distribution of W-patterns observed through a window and the output values of the operator. This is done by estimating the probability distribution of W-patterns and the conditional distribution of output values given a W-pattern. Then, a maximum-likelihood approach can be directly derived.

The choice of the pyramidal structure has direct influence in the quality of the designed operator. In practice, that choice is done in an *ad-hoc* manner. Another contribution of this paper is the proposal of the mean conditional entropy as a criterion to choose between different pyramids, under the assumption that the conditional distributions to be estimated have probability mass concentrated in one of the classes.

This paper is organized as follows. Section 2 recalls the definition of W-operators. In Section 3, we review related work in the field of W-operator design from input/output pairs, and in Section 4 the definition of maximum-likelihood classifiers is remembered. In Section 5 we propose a novel multiresolution joint distribution estimation algorithm, and in Section 6 the mean conditional entropy is suggested as a criterion to choose a pyramidal structure that induces a joint distribution. Experimental results are presented in Section 7, and conclusions and further research are discussed in Section 8.

2. W-operators

Digital signals (or images) can be formally defined and represented by functions from a finite rectangle E to a non-empty interval L. Usually, E is a subset of $\mathbb{Z} \times \mathbb{Z}$ and L is a subset of the positive integers $[0, l-1]$, $l \in \mathbb{Z}^+$. Binary images can be represented by elements of the collection of subsets of E, denoted by $\mathcal{P}(E)$. They can also be represented as a function from E to $[0, 1]$. The set of all functions from E to L will be denoted L^E. A mapping Ψ from L^E to L'^E will be called an image *operator* or *filter*, where L' is the interval $[0, l'-1]$, with $l' \in \mathbb{Z}^+$.

A finite subset W of E will be called a *window* and the number of points in W will be denoted by $|W|$. A *configuration* is a function from W to L and the space of all possible configurations from W to L will be denoted by L^W. A configuration is also called a W-pattern and usually results from translating a window W by t, $t \in E$, and observing the values of a signal $h \in L^E$ within the translated window, W_t. If $W = \{w_1, w_2, \ldots, w_n\}$, $n = |W|$, and we associate the points of W to a n-tuple $(w_1, w_2, \ldots, w_n)$, then a configuration $h(W_t)$ is given by

$$h(W_t) = (h(t + w_1), h(t + w_2), \ldots, h(t + w_n)). \quad (1)$$

Digital signals can be modelled by digital random functions [5], and, in this sense, $h(W_t)$ is a realization of a random vector $\mathbf{X} = (X_1, X_2, \ldots, X_n)$, that is, $h(W_t) = \mathbf{x} = (x_1, x_2, \ldots, x_n)$, where $\mathbf{x}$ denotes a realization of $\mathbf{X}$. An important subclass of operators from L^E to L'^E is the class of W-operators [2]. They are translation invariant (t.i.) and locally defined (l.d.) within a window W. If an image operator Ψ is a W-operator, then it can be characterized by a function $\psi : L^W \to L'$, called a *characteristic function*, by

$$\begin{aligned} \Psi(h)(t) &= \psi(h(t + w_1), h(t + w_2), \ldots, h(t + w_n)) \\ &= \psi(\mathbf{x}). \quad (2) \end{aligned}$$

3. Design of W-operators

Solving an image processing problem can be a very complex task. It relies primarily on the knowledge of the problem domain and on the knowledge, experience and intuition of an image processing expert. This complexity is a real motivation for some research groups to create automatic techniques to “imitate” the image expert. These techniques use a set of input/output image pairs in order to obtain an optimal characteristic function in relation to an error measure [1].

Formally, the problem can be stated as: given two random images on E, h to be observed and g to be estimated, find a W-operator Ψ that minimizes an error measure between $\Psi(h)(t)$ and $g(t)$, $t \in E$. More specifically, if $\mathbf{X}$ is a random variable over L^W, Y is a random variable over L' and $\Phi = \{\psi : \psi \text{ is a function from } L^W \text{ to } L'\}$, the problem consists in finding a characteristic function $\psi_{opt} \in \Phi$ such that $E[l(Y, \psi_{opt}(\mathbf{X}))] \leq E[l(Y, \psi(\mathbf{X}))]$ for all $\psi \in \Phi$, where $l(Y, \psi(\mathbf{X}))$ is the error measure that quantifies the difference between the ideal value $y \in Y$ and the value $\psi(\mathbf{x})$ returned by the operator. In practice, the joint probability distribution $p(\mathbf{X}, Y)$ is unknown, and it is estimated from a sample of $p(\mathbf{X}, Y)$, obtained from the input/output pairs (in this paper, we generally use a lowercase $p(\cdot)$ to denote a probability density function and an uppercase $P(\cdot)$ to denote a probability mass function).

The usual error measure for operators from $\mathcal{P}(E)$ to $\mathcal{P}(E)$ (binary image operators) is the Mean Absolute Error (MAE) [5]. For operators from $\mathcal{P}(E)$ to $[0, l-1]^E$ (binary image classifiers), it is the number of all misclassified objects or points. For operators from L^E to L'^E (gray-level operators), it's the Mean Square Error (MSE) [5].

In this context, we can cite some previous works that present methods for designing binary operators [1], binary classifiers [3] and gray-level operators [12] to solve image processing problems. More recently, multiresolution techniques have also been employed to facilitate the designing [6, 11].

4. Maximum-likelihood classifiers

Let W be a window, $\mathbf{X}$ a random variable over L^W, and Y a random variable over L'. Suppose that we want to design a characteristic function ψ that maps a configuration $\mathbf{x} \in \mathbf{X}$ to a value $y \in Y$. Given the conditional probability distribution $p(\mathbf{X}|Y)$, a well-known technique from statistical pattern recognition is the maximum-likelihood method [7]: choose ψ as the function ψ_{ml} that assigns to $\mathbf{x}$ the value y_{ml} of Y such that $P(\mathbf{x}|y_{ml}) \geq P(\mathbf{x}|y)$, for all $y \in Y$. If we had $p(\mathbf{X}|Y)$, it would be straightforward to assign a value to a W-pattern $\mathbf{x}$: compute $P(\mathbf{x}|y)$ for all $y \in Y$ and choose the y that gives the maximum value. In practice, $p(\mathbf{X}|Y)$ is unknown, thus it has to be estimated. In the next section, we introduce a method to estimate $p(\mathbf{X}|Y)$ from training examples and a pyramidal structure.

5. Joint distribution estimation

The task of estimating $p(\mathbf{X}|Y)$ from training samples can be very hard in practice, due to the exponential relationship between $|W|$ and the number of examples necessary to have a good estimate. The number of observed samples, in general, is much smaller than $|\mathbf{X}| = |L|^{|W|}$. Hence, the learning technique must be capable of determining good values of $P(\mathbf{X}|Y)$ for samples that don't appear in the training set.

In [6], the authors proposed a technique for binary filter design based in a pyramidal framework, which assigned estimates of $p(Y|\mathbf{x})$ for $\mathbf{x}$ not present in the training set by using data in multiple resolutions. Later, an extension to gray-level operators was developed using aperture operators [11].

In Section 5.1, we briefly recall the pyramidal framework introduced in [6]. In that work, the pyramidal structure is used to estimate the conditional probability distribution $p(Y|\mathbf{X})$. We will use that framework in Section 5.2 to estimate the densities $p(\mathbf{X})$ and $p(Y|\mathbf{X})$. With both estimates, one can easily derive $p(\mathbf{X}|Y)$ by

$$P(\mathbf{X}|Y) = \frac{P(Y|\mathbf{X}) \cdot P(\mathbf{X})}{P(Y)}, \tag{3}$$

as in general $p(Y)$ is easily estimated from the examples.

5.1. Pyramidal multiresolution analysis

Let W_0, W_1, …, W_r be a sequence of windows such that $W_{i+1} \subseteq W_i$, for $0 \leq i \leq r-1$, and let $D_0 = L_0^{W_0}$, $D_1 = L_1^{W_1}$, …, $D_r = L_r^{W_r}$ be the configuration spaces that can be observed through the windows W_0, W_1, …, W_r, where L_i, for $i \in \{0, \ldots, r\}$ are intervals in the form $[0, l_i - 1]$, $l_i \in \mathbb{Z}^+$. Define a sequence of resolution mappings $\rho_{01} : D_0 \rightarrow D_1$, $\rho_{12} : D_1 \rightarrow D_2$, …, $\rho_{(r-1)r} : D_{r-1} \rightarrow D_r$ which induce nested partitions $\mathcal{X}^1, \mathcal{X}^2, \ldots, \mathcal{X}^r$ of the space D_0 by the equivalence relations $\mathbf{x} \sim_1 \mathbf{x}' \Leftrightarrow \rho_{01}(\mathbf{x}) = \rho_{01}(\mathbf{x}')$, for $\mathbf{x}, \mathbf{x}' \in D_0$, $\mathbf{x} \sim_2 \mathbf{x}' \Leftrightarrow \rho_{12}(\rho_{01}(\mathbf{x})) = \rho_{12}(\rho_{01}(\mathbf{x}'))$, and so on. Moreover, define the partition $\mathcal{X}^0 = \{\{\mathbf{x}_1\}, \{\mathbf{x}_2\}, \ldots, \{\mathbf{x}_{|D_0|}\}\}$, where each set in $\mathcal{X}^0$ contains one element of D_0.

In [6], the conditional distribution $p(Y|\mathbf{X})$ is estimated using the nested partitions. To obtain $p(Y|\mathbf{x})$ for a configuration $\mathbf{x} \in \mathbf{X}$, a multiresolution analysis is employed. It consists in using the greater resolution for which we have sufficient observations of $\mathbf{x}$ in the examples to make the estimation, that is, if we don't have a good estimate of $p(Y|\mathbf{x})$ at the resolution of D_0, the resolution mapping ρ_{01} is applied to $\mathbf{x}$ and the estimate at the resolution of D_1 is verified. If we still don't have a good estimate, ρ_{12} is applied to $\rho_{01}(\mathbf{x})$, and so on.

In the remainder of this paper, we will refer to a sequence of windows $W_0, \ldots, W_r$ accompanied by a sequence of resolution mappings $\rho_{01} : D_0 \rightarrow D_1$, $\rho_{12} : D_1 \rightarrow D_2$, …, $\rho_{(r-1)r} : D_{r-1} \rightarrow D_r$ as a *window pyramid.*

5.2. Estimation algorithm

In this section, our goal is to obtain an estimate of the joint distribution, by estimating $p(\mathbf{X})$ and $p(Y|\mathbf{X})$. First, we introduce some notation. Let W be a finite window and $p(\mathbf{X})$ be a probability distribution on D_0. We assume that there exists a partition $\mathcal{X} = \{\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_n\}$ of D_0 and a probability distribution $\Gamma(\mathcal{X})$ on $\mathcal{X}$ such that

$$\forall \mathbf{x}_i \in \mathcal{X}_i, P(Y|\mathbf{x}_i) = P(Y|\mathcal{X}_i) \tag{4}$$

and

$$\Gamma(\mathcal{X}_i) = \sum_{\mathbf{x} \in \mathcal{X}_i} P(\mathbf{x}), \tag{5}$$

for all i in $\{1, \ldots, n\}$. Equation 4 means that all configurations in a part $\mathcal{X}_i$ of $\mathcal{X}$ have the same conditional distribution $p(Y|\mathcal{X}_i)$, and Equation 5 states that the probability of a set $\mathcal{X}_i$ is the sum of the probabilities for each configuration in it.

Let $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_m, y_m)$ be a sampling of $p(\mathbf{X}, Y)$ (these are obtained from the input/output pairs), and let Δ be a window pyramid. We will denote the nested partitions induced by Δ, in decreasing resolution order, by $\mathcal{X}^0 = \{\mathcal{X}_1^0, \ldots, \mathcal{X}_{n_0}^0\}, \ldots, \mathcal{X}^r = \{\mathcal{X}_1^r, \ldots, \mathcal{X}_{n_r}^r\}$. Now consider a non-empty subset $\mathcal{S}$ of D_0. We define

$$N_{\mathcal{S}} = \sum_{k=1}^{m} c_{\mathcal{S}}(\mathbf{x}_k), \tag{6}$$

where

$$c_{\mathcal{S}}(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x} \in \mathcal{S} \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

$N_{\mathcal{S}}$ is the total number of times that the configurations in $\mathcal{S}$ appear in the sample pairs.

The estimation of $p(Y|\mathcal{S})$ is given by

$$P(Y|\mathcal{S}) = \frac{\sum_{k=1}^{m} l_{\mathcal{S}}(\mathbf{x}_k, y_k)}{N_{\mathcal{S}}}, \tag{8}$$

where

$$l_{\mathcal{S}}(\mathbf{x}_k, y_k) = \begin{cases} 1, & \text{if } \mathbf{x}_k \in \mathcal{S} \text{ and } y_k = Y \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

That is, the conditional probability distribution for $\mathcal{S}$ is calculated by simply taking the distribution of Y for the configurations in $\mathcal{S}$.

In order to estimate $p(\mathbf{X})$, we must assign probability masses to the configurations in D_0. We will do this for groups of configurations (the sets in $\mathcal{X}$). For a non-empty subset $\mathcal{S}$ of D_0, its probability is given by

$$M_{\mathcal{S}} = \frac{T_\alpha(N_{\mathcal{S}})}{\sum_{k=1}^{n_r} T_\alpha(N_{\mathcal{X}_k^r})}, \tag{10}$$

where

$$T_\alpha(n) = \begin{cases} n, & \text{if } n \geq \alpha \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

The number α is a parameter to the algorithm, and it must be greater than or equal to 2. The intuition behind α is to consider, at a particular resolution level i, only the sets $\mathcal{X}_k^i$ of $\mathcal{X}^i$ such that the number of observations in $\mathcal{X}_k^i$ is greater than or equal to α. As we go from resolution 0 to r, the partition $\mathcal{X}^i$ gets progressively coarser, hence the number of elements in the sets $\mathcal{X}_k^i$ grows.

We are now ready to describe an algorithm to estimate $\Gamma(\mathcal{X})$ and $p(Y|\mathcal{X})$. It receives α, Δ and $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_m, y_m)$ as inputs, and returns a set $\mathcal{R}$ of triples in the form $(\mathcal{X}_i, \Gamma(\mathcal{X}_i), p(Y|\mathcal{X}_i))$. $\mathcal{R}$ is a representation of the joint distribution, and the sets $\mathcal{X}_i$ that appear in triples of $\mathcal{R}$ are the ones in $\mathcal{X}$ such that $\Gamma(\mathcal{X}_i) > 0$. We work under the assumption that $\sum_{k=1}^{n_r} T_\alpha(N_{\mathcal{X}_k^r}) > 0$, which means that the partition $\mathcal{X}^r$ has at least one set $\mathcal{X}_i^r$ whose elements appear at least α times in the samples.

Initialize $\mathcal{R}$ as an empty set. Then, begin processing the first resolution. For the sets $\mathcal{X}_i^0$, $i \in \{1, \ldots, n_0\}$, compute its probability mass $M = M_{\mathcal{X}_i^0}$. If $M > 0$, then $\Gamma(\mathcal{X}_i^0) = M$, compute $p(Y|\mathcal{X}_i^0)$ as in Equation 8 and add the triple $(\mathcal{X}_i^0, \Gamma(\mathcal{X}_i^0), p(Y|\mathcal{X}_i^0))$ to $\mathcal{R}$.

The other resolutions are processed iteratively. For each resolution j from 1 to r, compute, for each set $\mathcal{X}_i^j$, $i \in \{1, \ldots, n_j\}$, the set

$$\mathcal{K} = \{\mathcal{Z} : \text{ there exists a triple } (\mathcal{Z}, \cdot, \cdot) \in \mathcal{R} \text{ and } \mathcal{Z} \subseteq \mathcal{X}_i^j\}. \tag{12}$$

$\mathcal{K}$ contains the subsets of $\mathcal{X}_i^j$ whose probability masses were already assigned during the processing of a previous resolution. The mass M is calculated by

$$M = M_{\mathcal{X}_i^j} - \sum_{K \in \mathcal{K}} \Gamma(K). \tag{13}$$

Let $\mathcal{U}$ be the union of the sets in $\mathcal{K}$. If $M > 0$, then $\Gamma(\mathcal{X}_i^j - \mathcal{U}) = M$ and compute $p(Y|\mathcal{X}_i^j)$ as in Equation 8. The distribution $p(Y|\mathcal{X}_i^j - \mathcal{U})$ will be equal $p(Y|\mathcal{X}_i^j)$. Add the triple $(\mathcal{X}_i^j - \mathcal{U}, \Gamma(\mathcal{X}_i^j - \mathcal{U}), p(Y|\mathcal{X}_i^j - \mathcal{U}))$ to $\mathcal{R}$. Equation 13 guarantees that, when assigning probability masses at a certain resolution, the mass from subsets that has been already assigned in previous resolutions doesn't get included.

After the resolution r is processed, return $\mathcal{R}$. It is a representation of the joint distribution, as it contains the sets of $\mathcal{X}$ that have $\Gamma(\mathcal{X}_i) > 0$, its associated conditional probability distributions and the values of $\Gamma(\mathcal{X}_i)$. An important observation is that we are estimating $\Gamma(\mathcal{X})$, but not the complete distribution $p(\mathbf{X})$. In maximum-likelihood classification the value of $P(\mathbf{X})$ appears as a constant (Equation 3), and we will see in Section 6.3 that $\Gamma(\mathcal{X})$ is sufficient for the estimation quality criterion that we are proposing. But if the values of $P(\mathbf{X})$ are necessary in a special application, one can consider any probability model that satisfies Equation 5.

6. Pyramid choice

As we have seen so far, the choice of the window pyramid determines the structure of the nested partitions, hence it has large impact on the quality of the designed operator. In practice, the use of a specific pyramid can be a bad choice in some problems, but lead to good results in others; that is, a good choice also depends on the problem at hand.

In this section, we introduce a criterion based on concepts of information theory [14] that provides a way to choose a good pyramid from a set of predefined ones, given a set of input/output image pairs that contain information about the desired solution.

6.1. Entropy

If $\mathbf{X}$ is a discrete random variable and $p(\mathbf{X})$ is its probability distribution, the entropy $H(\mathbf{X})$ of $\mathbf{X}$ is defined as

$$H(\mathbf{X}) \quad = \quad -\sum_{\mathbf{x} \in \mathbf{X}} P(\mathbf{x}) log_2(P(\mathbf{x})). \tag{14}$$

Entropy can be seen as a measure of concentration of the probability mass. If the probabilities concentrate over a value, then the entropy should be low. If the probabilities distribute in a rather uniform way, it should be high.

An important measure for conditional distributions is the *mean conditional entropy*. Given the distribution $p(Y|\mathbf{X})$, the mean conditional entropy of $Y|\mathbf{X}$ is defined as:

$$E[H(Y|\mathbf{X})] \quad = \quad \sum_{\mathbf{x} \in \mathbf{X}} P(\mathbf{x}) \cdot H(Y|\mathbf{x}). \tag{15}$$

6.2. Mutual information

Mutual information is a measure of dependency between two random variables $\mathbf{X}$ and Y. It is defined as [13]:

$$I(\mathbf{X}, Y) = H(Y) - H(Y|\mathbf{X}). \tag{16}$$

If $\mathbf{X}$ and Y are independent, $I(\mathbf{X}, Y) = 0$. The derivation of the following expression is straightforward from Equation 16:

$$E[I(\mathbf{X}, Y)] = H(Y) - E[H(Y|\mathbf{X})]. \tag{17}$$

A direct consequence is the equivalence between minimizing the mean conditional entropy and maximizing the mutual information. As the dependency between $\mathbf{X}$ and Y increases, the conditional distribution $p(Y|\mathbf{X})$ tends to have more probability mass concentrated around some values of Y. Therefore, conditional distributions whose mass concentrates in one of the possible classes have lower entropy.

6.3. Quality criterion

In classification problems, it is usually desired that the true conditional distributions $p(Y|\mathbf{x})$ have probability mass concentrated in one of the possible classes $y \in Y$. This way, for a feature vector $\mathbf{x} \in \mathbf{X}$ there will be a class $y \in Y$ that will be more probable, leading to less uncertainty in the classification.

In order to establish a criterion to choose a joint distribution represented by a window pyramid (as seen in Section 5.2), we work with the assumption that the true conditional distributions (that we wish to estimate from examples) have probability mass concentrated in one of the classes. This is reasonable in problems for which there exists a good solution. The search for this distribution can be done by finding a window pyramid which represents a joint distribution that has small mean conditional entropy.

Recall from Section 5.2 the partition $\mathcal{X} = \{\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_n\}$, whose elements that have non-zero probabilities are computed by the estimation algorithm. The mean conditional entropy in a distribution represented by a window pyramid can be obtained from $\mathcal{X}$ by

$$\begin{aligned} E[H(Y|\mathbf{X})] &= \sum_{\mathbf{x}\in\mathbf{X}} P(\mathbf{x}) \cdot H(Y|\mathbf{x}) \\ &= \sum_{i=1}^{n} \sum_{\mathbf{x}\in\mathcal{X}_i} P(\mathbf{x}) \cdot H(Y|\mathbf{x}) \\ &= \sum_{i=1}^{n} \Big(\sum_{\mathbf{x}\in\mathcal{X}_i} P(\mathbf{x})\Big) \cdot H(Y|\mathcal{X}_i) \\ &= \sum_{i=1}^{n} \Gamma(\mathcal{X}_i) \cdot H(Y|\mathcal{X}_i). \qquad (18) \end{aligned}$$

Thus, the computation of $E[H(Y|\mathbf{X})]$ can be done directly from the set $\mathcal{R}$ (the result of the estimation algorithm). The four equalities shown above come, respectively, from Equation 15, the fact that $\mathcal{X}$ is a partition of D_0, Equation 4 and Equation 5.

As the number of possible window pyramids is huge, it isn't computationally feasible to estimate the joint distribution and calculate the mean conditional entropy for all of them, then choose the pyramid that minimizes it. In practice, the image processing specialist should define a set of candidate pyramids, which will be compared using the entropy criterion. A heuristic way of defining the candidate set is by using resolution mappings which are known for providing good results in some applications, e.g., mappings from image pyramids theory [4, 8, 9], that are employed in areas such as image compression and coding. The entropy criterion will choose the pyramid whose induced conditional distributions in average have more concentrated mass. Given our assumption of concentrated mass in the real conditional distributions, the criterion will choose a good pyramid, assuming that there exists one candidate in the set that induces conditional distributions with concentrated mass.

Another important remark about the entropy measure is the fact that it reflects the mass concentration, but says nothing about the values in which the mass is concentrated. Hence, if more than one pyramid in the candidate set induce distributions with small mean conditional entropy, another quality measure (e.g. experimental MAE) should be used to differentiate them.

7. Experimental results

In order to verify the effectiveness of the entropy criterion to choose a good pyramid in a real application, we have conducted some experiments with the problem of handwritten digits recognition. For each one of the 10 digits, 10 binary images of approximately 672 digits each have been obtained by digitalizing (200 dpi) paper forms with samples taken from various subjects. Some examples of the digits can be seen at Figure 1. The images used in our experiments are available in the World Wide Web, at `http://www.vision.ime.usp.br/~daniel/sibgrapi2005/`. When collecting the data, our objective was to simulate the writing of digits in postal envelopes. The individuals have been asked to write several digits inside square regions.

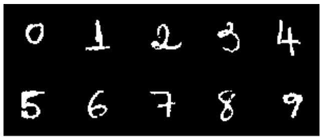

Figure 1. Examples of handwritten digits.

The objective is to design a W-operator that when applied to similar images gives as output a value in the set $\{0, 1, \ldots, 9\}$ for pixels that have value 1 in the input image.

7.1. Test procedure

To define the training set, 70% of the images from each digit were selected. The remaining 30% were used to evaluate the performance of the designed classifiers. A set of 11 window pyramids based on subsampling operators has also been specified, with base windows (the greater window in the pyramid) varying from $9\times 9, 11\times 11, 13\times 13$ to 17×17. Each pyramid has been labeled as an integer from 1 to 11. Figure 2 shows the specifications of the pyramids, where each pyramid is represented as a window that determines the shape of W_0 (the base window). The subsequent windows W_i, $i \geq 1$ can be obtained by taking only the points whose values are greater than or equal to $i + 1$.

First, a preprocessing is done to determine the regions that correspond to each digit. The images have been dilated by a radius five Euclidean disk, and the connected components whose bounding box had width or height lesser than 17 or greater than 73 pixels have been eliminated. Finally, the intersection of each remaining connected component

1	1	1	1	1	1	1	1	1	1	1
1	2	2	2	2	2	2	2	2	2	1
1	2	3	3	3	3	3	3	3	2	1
1	2	3	4	4	4	4	4	3	2	1
1	2	3	4	5	5	5	4	3	2	1
1	2	3	4	5	6	5	4	3	2	1
1	2	3	4	5	5	5	4	3	2	1
1	2	3	4	4	4	4	4	3	2	1
1	2	3	3	3	3	3	3	3	2	1
1	2	2	2	2	2	2	2	2	2	1
1	1	1	1	1	1	1	1	1	1	1

Pyramid 1

1	1	1	1	1	1	1	1	1
1	2	2	2	2	2	2	2	1
1	2	3	3	3	3	3	2	1
1	2	3	4	4	4	3	2	1
1	2	3	4	5	4	3	2	1
1	2	3	4	4	4	3	2	1
1	2	3	3	3	3	3	2	1
1	2	2	2	2	2	2	2	1
1	1	1	1	1	1	1	1	1

Pyramid 2

					1					
				1	2	1				
			1	2	3	2	1			
		1	2	3	4	3	2	1		
	1	2	3	4	5	4	3	2	1	
1	2	3	4	5	6	5	4	3	2	1
	1	2	3	4	5	4	3	2	1	
		1	2	3	4	3	2	1		
			1	2	3	2	1			
				1	2	1				
					1					

Pyramid 3

				1				
			1	2	1			
		1	2	3	2	1		
	1	2	3	4	3	2	1	
1	2	3	4	5	4	3	2	1
	1	2	3	4	3	2	1	
		1	2	3	2	1		
			1	2	1			
				1				

Pyramid 4

1	1	1	2	2	2	2	2	1	1	1
1	2	2	3	4	4	4	3	2	2	1
1	2	4	5	6	6	6	5	4	2	1
2	3	5	7	8	9	8	7	5	3	2
2	4	6	8	10	11	10	8	6	4	2
2	4	6	9	11	12	11	9	6	4	2
2	4	6	8	10	11	10	8	6	4	2
2	3	5	7	8	9	8	7	5	3	2
1	2	4	5	6	6	6	5	4	2	1
1	2	2	3	4	4	4	3	2	2	1
1	1	1	2	2	2	2	2	1	1	1

Pyramid 5

1	1	1	1	1	1	1	1	1	1	1
3	3	3	3	3	3	3	3	3	3	3
1	1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2
1	1	1	1	1	1	1	1	1	1	1
4	6	4	5	4	7	4	5	4	6	4
1	1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2
1	1	1	1	1	1	1	1	1	1	1
3	3	3	3	3	3	3	3	3	3	3
1	1	1	1	1	1	1	1	1	1	1

Pyramid 6

1	1	1	1	1	1	1	1	1	1	1
1	3	1	2	1	3	1	2	1	3	1
1	1	1	1	1	1	1	1	1	1	1
1	2	1	2	1	2	1	2	1	2	1
1	1	1	1	1	1	1	1	1	1	1
1	3	1	2	1	4	1	2	1	3	1
1	1	1	1	1	1	1	1	1	1	1
1	2	1	2	1	2	1	2	1	2	1
1	1	1	1	1	1	1	1	1	1	1
1	3	1	2	1	3	1	2	1	3	1
1	1	1	1	1	1	1	1	1	1	1

Pyramid 7

3	1	2	1	3	1	2	1	3
1	1	1	1	1	1	1	1	1
2	1	2	1	2	1	2	1	2
1	1	1	1	1	1	1	1	1
3	1	2	1	4	1	2	1	3
1	1	1	1	1	1	1	1	1
2	1	2	1	2	1	2	1	2
1	1	1	1	1	1	1	1	1
3	1	2	1	3	1	2	1	3

Pyramid 8

2	1	2	1	2	1	2	1	2	1	2	1	2
1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	3	1	2	1	3	1	2	1	3	1	2
1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	2	1	2	1	2	1	2	1	2	1	2
1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	3	1	2	1	4	1	2	1	3	1	2
1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	2	1	2	1	2	1	2	1	2	1	2
1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	3	1	2	1	3	1	2	1	3	1	2
1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	2	1	2	1	2	1	2	1	2	1	2

Pyramid 10

4	1	2	1	3	1	2	1	4	1	2	1	3	1	2	1	4
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	1	2	1	3	1	2	1	3	1	2	1	3	1	2	1	3
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	2	1	3	1	2	1	5	1	2	1	3	1	2	1	4
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	1	2	1	3	1	2	1	3	1	2	1	3	1	2	1	3
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	2	1	3	1	2	1	4	1	2	1	3	1	2	1	4

Pyramid 9

8	1	3	1	5	1	3	1	7	1	3	1	5	1	3	1	8
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1
3	1	4	1	3	1	4	1	3	1	4	1	3	1	4	1	3
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1
5	1	3	1	6	1	3	1	5	1	3	1	6	1	3	1	5
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1
3	1	4	1	3	1	4	1	3	1	4	1	3	1	4	1	3
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1
7	1	3	1	5	1	3	1	9	1	3	1	5	1	3	1	7
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1
3	1	4	1	3	1	4	1	3	1	4	1	3	1	4	1	3
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1
5	1	3	1	6	1	3	1	5	1	3	1	6	1	3	1	5
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1
3	1	4	1	3	1	4	1	3	1	4	1	3	1	4	1	3
1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1
8	1	3	1	5	1	3	1	7	1	3	1	5	1	3	1	8

Pyramid 11

Figure 2. Window pyramids used in the experiments.

and the original image is defined as a single digit. Therefore, the result is an image that contains unaltered digits from the original image, except very large or very small structures. The morphological dilation provides a way to put together in the same digit pixels separated by little cracks.

When collecting W-patterns from images, it is not desirable that, when observing a certain configuration that is part of a digit, pixels from neighboring digits also get included as part of the W-pattern. To avoid this, a new image is created by increasing the distance between digits. We also haven't considered W-patterns such that the center of the translated window is a background pixel, since the resulting W-operator will be applied only to configurations whose central pixel has value 1.

The next part of the procedure is the joint distribution estimation for each pyramid, using the algorithm from Section 5.2. In our experiments, $\alpha = 2$. One joint distribution is estimated for each pyramid. From each distribution, the mean conditional entropy is calculated and a maximum-likelihood classifier is designed.

The classifiers have been applied to all images in the test set, resulting in gray-level images. A postprocessing is done to assign a unique value to all pixels that belong to the same digit. The most frequent value among them has been chosen for this purpose. Then an error value for each classifier is obtained by computing the number of misclassified digits divided by the total number of digits.

7.2. Results

The pyramid labels can be seen in Table 1, ordered by mean conditional entropy values from their corresponding joint distributions. In Table 2, the error rates in maximum-likelihood classification obtained from the test set are listed. The total number of digits considered in the test stage is 18874.

7.3. Analysis

In Tables 1 and 2, we can observe that the pyramid of least entropy coincides with the pyramid of smaller error on the test set. This shows that the mean conditional entropy seems to be a good criterion to choose between different joint distributions. The pyramid labeled with the number 11 has base window of 17 × 17 pixels, and is built using a quincunx sampling scheme [10] to determine the subsequent windows.

We can also see that the ordering of the pyramids by mean conditional entropy is not the same as the ordering by error. The analysis of this phenomenon is a delicate problem, since there are many variables that can influence the results, such as training and test set size, estimation errors, sample quality (are the training and test samples good representatives of the real distribution?), the candidate set of pyramids, base window sizes, the fact that the mean conditional entropy computes an average value, etc. To have a significant conclusion about the relationship between the mean conditional entropy and the experimental error, a more rigorous statistical analysis is needed, from both theoretical and experimental standpoints.

Table 1. Mean conditional entropies

Pyramid	$E[H(Y\|\mathbf{X})]$
11	1.053517
9	1.463839
5	1.595312
6	1.803493
1	1.899382
10	1.926978
8	1.993410
2	2.029149
3	2.164285
7	2.257981
4	2.679645

Table 2. Classification errors

Pyramid	Test error
11	11.40%
9	14.44%
8	28.48%
7	29.61%
10	31.22%
6	31.56%
3	45.14%
5	45.23%
2	48.28%
1	48.30%
4	49.47%

However, our technique is based in sound mathematical foundations, and the results show that it can be used to build a tool to help the image processing specialist in the pyramid choice. This tool receives as input the set of candidate pyramids and a set of input/output pairs, and returns the pyramid that induces a joint distribution of minimum entropy. Then the pyramid could be used to design the operator, avoiding the need to design operators for all pyramids in the candidate set and evaluating their errors, which can be a tedious and expensive procedure.

The choice of the set of candidate pyramids is a key factor in the obtained results. As we are under the assumption that the real conditional distributions have probability mass concentrated in one of the classes, the entropy criterion will find a good distribution if it is one of the candidates, that is, if the pyramid set contains a candidate that induces a distribution with those properties. We have suggested in Section 6.3 that there are heuristic ways to choose the candidates.

8. Conclusion

Joint probability distribution estimation is a key problem in pattern classification. In this paper, we have proposed a technique to estimate the joint distribution for designing a W-operator. The estimator is based on a multiresolution pyramidal structure that induces a probability density by partitioning the W-pattern space in equivalence classes. A maximum-likelihood classifier can be designed in a straightforward way from the estimated distribution.

In pyramidal design of operators, the pyramid used in the estimation process has significant impact on the results, and has been chosen in an *ad-hoc* manner. Motivated by this fact, we have proposed the mean conditional entropy as a measure of the quality of the estimated distribution. It is based on the assumption that the real conditional distributions $p(Y|\mathbf{x})$ have probability mass concentrated in one of the possible classes $y \in Y$. This is a reasonable assumption for problems where we know that there exists a good solution. The entropy measure can be used to develop a tool to help the image processing specialist in the choice of a pyramid.

We had also shown some preliminary results based on the problem of classifying handwritten digits. The analysis confirm that the mean conditional entropy is a good criterion for choosing a pyramid. There is potential for better results by considering more sophisticated resolution mappings, such as the ones from image pyramids theory [4, 8, 9].

Therefore, the proposed technique is interesting from the theoretical point of view and has potential to be applied in many problems in computer vision and image processing. Further research includes experimentation with more sophisticated resolution mappings and a more rigorous analysis of the estimator from the statistical point of view, in order to have a better understanding of the relationship between the entropy measure and the experimental error. It is also important to note that although our experiments have had been done on binary images, the proposed technique is general and can be applied to gray-level and color images.

9. Acknowledgements

The authors are grateful to their supporting agencies. Daniel André Vaquero is supported by FAPESP (grant number 03/10066-7). Junior Barrera is partially supported by CNPq and by the National Institutes of Health, USA, under grant 1 D43 TW07015-01. Roberto Hirata Jr. is partially supported by CNPq.

References

[1] J. Barrera, E. R. Dougherty, and N. S. Tomita. Automatic Programming of Binary Morphological Machines by Design of Statistically Optimal Operators in the Context of Computational Learning Theory. *Electronic Imaging*, 6(1):54–67, January 1997.

[2] J. Barrera and G. P. Salas. Set Operations on Closed Intervals and Their Applications to the Automatic Programming of Morphological Machines. *Electronic Imaging*, 5(3):335–352, July 1996.

[3] J. Barrera, R. Terada, F. S. C. da Silva, and N. S. Tomita. Automatic Programming of Morphological Machines for OCR. In *Mathematical Morphology and its Applications to Image and Signal Processing*, pages 385–392, Atlanta, GA, May 1996. International Symposium on Mathematical Morphology, Kluwer Academic Publishers.

[4] P. J. Burt and E. H. Adelson. The Laplacian Pyramid as a Compact Image Code. *IEEE Transactions on Communications*, COM-31(4):532–540, April 1983.

[5] E. R. Dougherty. *Random Processes for Image and Signal Processing*. SPIE and IEEE Presses, Bellingham, 1999.

[6] E. R. Dougherty, J. Barrera, G. Mozelle, S. Kim, and M. Brun. Multiresolution Analysis for Optimal Binary Filters. *Journal of Mathematical Imaging and Vision*, (14):53–72, 2001.

[7] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley and Sons, 2001.

[8] J. Goutsias and H. Heijmans. Nonlinear multiresolution signal decomposition schemes. I. Morphological pyramids. *IEEE Transactions on Image Processing*, 9(11):1862–1876, November 2000.

[9] J. Goutsias and H. J. A. M. Heijmans. Multiresolution Signal Decomposition Schemes. Part 1: Linear and Morphological Pyramids. Technical Report PNA-R9810, CWI, October 1998.

[10] H. J. A. M. Heijmans and J. Goutsias. Morphological Pyramids and Wavelets Based on the Quincunx Lattice. In *Proceedings of the 5th International Symposium on Mathematical Morphology and Its Applications to Image and Signal Processing - ISMM'2000*, Palo Alto, June 2000.

[11] R. Hirata Jr., M. Brun, J. Barrera, and E. R. Dougherty. Multiresolution Design of Aperture Operators. *Journal of Mathematical Imaging and Vision*, 16(3):199–222, 2002.

[12] R. Hirata Jr., E. R. Dougherty, and J. Barrera. Aperture Filters. *Signal Processing*, 80(4):697–721, April 2000.

[13] D. J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.

[14] C. E. Shannon. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27:379–423, 623–656, July, October 1948.