Automatic Window Design for Gray-Scale Image Processing Based on Entropy Minimization

David C. Martins Jr., Roberto M. Cesar Jr., and Junior Barrera

USP–Universidade de São Paulo,
IME–Instituto de Matemática e Estatística,
Computer Science Department,
Rua do Matão, 1010 - Cidade Universitária,
CEP: 05508-090, São Paulo, SP, Brasil

Abstract. This paper generalizes the technique described in [1] to gray-scale image processing applications. This method chooses a subset of variables W (i.e. pixels seen through a window) that maximizes the information observed in a set of training data by mean conditional entropy minimization. The task is formalized as a combinatorial optimization problem, where the search space is the powerset of the candidate variables and the measure to be minimized is the mean entropy of the estimated conditional probabilities. As a full exploration of the search space requires an enormous computational effort, some heuristics of the feature selection literature are applied. The introduced approach is mathematically sound and experimental results with texture recognition application show that it is also adequate to treat problems with gray-scale images.

1 Introduction

The paper [1] discusses a technique based on information theory concepts to estimate a good W-operator to perform binary image transformations (e.g. noisy image filtering). A W-operator is an image transformation that is locally defined inside a window W and translation invariant [2]. This means that it depends just on shapes of the input image seen through the window W and that the transformation rule applied is the same for all image pixels. A remarkable property of a W-operator is that it may be characterized by a Boolean function which depends on $|W|$ variables, where $|W|$ is the cardinality of W.

Here, the W-operator will be extended to be applied to gray-scale images. For this, instead of considering it as a Boolean function, we will consider it as a function whose domain is a vector of integer numbers (gray levels) and the output is a integer number (one of the considered classes). Then, the method developed in [1] can be extended to deal with this problem in a similar way to the design of W-operators for binary image transformations.

In order to build the training set, the adopted window collects feature vectors (vectors of integer numbers representing gray levels) translating over the input gray-scale images. From this training set, a gray-scale W-operator is estimated.

M. Lazo and A. Sanfeliu (Eds.): CIARP 2005, LNCS 3773, pp. 813–824, 2005.

This task is an optimization problem. The training data gives a sample of a joint distribution of the observed feature vectors and their classification. A loss function measures the cost of a feature vector missclassification. An operator error is the expectation of the loss function under the joint distribution. Given a set of operators, the target operator is the one that has minimum error. As, in practice, the joint distribution is known just by its samples, it should be estimated. This implies that operators error should also be estimated and, consequently, the target operator itself should be estimated. Estimating an operator is an easy task when the sampling of the joint distribution considered is large. However, this is rarely the case. Usually, the problem involves large windows with non concentrated probability mass joint distributions requiring prohibitive amount of training data.

The fact that each pixel in gray-scale images contains more than two possible values worsens the problem of lack of training data. Because of this, an approach for dealing with the lack of training data becomes even more required. By constraining the considered space of operators, less training data is necessary to get good estimations of the best candidate operator [3]. However, depending on how many gray levels exists in an image, the constraint may be so excessive that even the best operator of such space lead to very bad classification results. Therefore, quantization is usually necessary.

In this paper, we discuss how to apply the criterion function used in [1] to estimate an sub-window W^* that gives one of the best operators to perform classification over images with arbitrary number of gray levels and arbitrary number of classes.

The search space of this problem is the powerset of W, denoted $\mathcal{P}(W)$. The criterion to be minimized is the degree of mixture of the observed classes. The mean conditional entropy is adopted as a measure of this degree. The important property of entropy explored here is that when the probability mass of a distribution becomes more concentrated somewhere in its domain, the entropy decreases. This means that when a given feature vector defined in a window has a majoritary label (i.e. it is classified almost always in a same class), its entropy of the conditional distribution should be low. Thus, the optimization algorithm consists in estimating the mean conditional entropy for the joint distribution estimated for each sub-window and choosing the one that minimizes this measure.

Each observed feature vector has a probability and a corresponding conditional distribution from which the entropy is computed. The mean conditional entropy is the mean of the computed entropies, weighted by the feature vector probabilities.

As $\mathcal{P}(W)$ has an exponential size in terms of the cardinality of W, we adopted some heuristics to explore this space in reasonable computational time. The adopted heuristic was the SFFS feature selection algorithm [4].

Following this Introduction, Section 2 recalls the mathematical fundamentals of W-operators design with extension to gray-scale images. Section 3 introduces the definitions and properties of the mean conditional entropy and presents the

proposed technique for generating the minimal window and, consequently, choosing a minimal family of operators. Section 4 presents results of the application of the proposed technique to recognize textures with multilevel gray tone. Finally, Section 5 presents some concluding remarks of this work.

2 W-Operator Definition and Design

In this section, we recall the notion of W-operator and the main principles for designing W-operators from training data.

2.1 W-Operator Definition and Properties

Let E denote the integer plane and $+$ denote the vector addition on E. The opposite of $+$ is denoted $-$. An *image* is a function f from E to $L = \{1, ..., k\}$, where k is the number of gray tones.

The *translation* of an image f by a vector $h \in E$ is the image $f(x)_h$. An *image classification* or *operator* is a mapping Ψ from L^E into Y^E, where $Y = \{1, ..., c\}$ is the set of labels (classes).

An operator Ψ is called *translation invariant* iff, for every $h \in E$ and $f \in L^E$,

$$\Psi(f_x) = (\Psi(f))_x \ . \tag{1}$$

Let W be a finite subset of E. A *constraint class* of f over W, denoted $C_{f|W}$, is the family of functions whose constraint to W results in $f|W$, i.e.,

$$C_{f|W} = \{g \in L^E : f|W = g|W\} \ . \tag{2}$$

An operator $\Psi : L^E \to Y^E$ is called *locally defined in the window* W iff, for every $x \in E$, $f \in L^E$.

$$\Psi(f)(x) = \Psi(g), \forall g \in C_{f_{-x}|W} \ . \tag{3}$$

An operator is called a *W-operator* if it is both translation invariant and locally defined in a finite window W. Given a W-operator $\Psi : L^E \to Y^E$, exists one characteristic function $\psi : L^W \to Y$ such that:

$$\Psi(f)(x) = \psi(f_{-x}|W), \forall x \in E \ . \tag{4}$$

2.2 W-Operator Design

Designing an operator means choosing an element of a family of operators to perform a given task. One formalization of this idea is as an optimization problem, where the search space is the family of candidate operators and the optimization criteria is a measure of the operator quality. In the commonly adopted formulation, the criteria is based on a statistical model for the images associated to a measure of images similarity, the loss function.

Let $\mathbf{S}$ and $\mathbf{I}$ be two discrete random functions defined on E, i.e. realizations of $\mathbf{S}$ or $\mathbf{I}$ are images obtained according with some probability distribution on L^E. Let us model image transformations in a given context by the joint random process $(\mathbf{S}, \mathbf{I})$, where the process $\mathbf{S}$ represents the input images and $\mathbf{I}$ the output images. The process $\mathbf{I}$ depends on the process $\mathbf{S}$ according to a conditional distribution.

Given a space of operators $\mathcal{F}$ and a loss function ℓ from $L \times L$ to $\Re^+$, the error $Er[\Psi]$ of an operator $\Psi \in F$ is the expectation of $\ell(\Psi(\mathbf{S}), \mathbf{I})$, i.e., $Er[\Psi] = E[\ell(\Psi(\mathbf{S}), \mathbf{I})]$. The *target* operator Ψ_{opt} is the one of minimum error, i.e., $Er[\Psi_{opt}] \leq Er[\Psi]$, for every $\Psi \in \mathcal{F}$.

A joint random process $(\mathbf{S}, \mathbf{I})$ is jointly stationary in relation to a finite window W, if the probability of seeing a given feature vector in the input image through W together with a given value in the output image is the same for every translation of W, that is, for every $x \in E$,

$$P((S|W_x, I(x)) = P(S|W, I(o)) \ , \tag{5}$$

where S is a realization of $\mathbf{S}$, I is the function equivalent to a realization of $\mathbf{I}$, and o is the origin of E.

In order to make the model usable in practice, from now on suppose that $(\mathbf{S}, \mathbf{I})$ is jointly stationary w.r.t the finite window W. Under this hypothesis, the error of predicting an image from the observation of another image can be substituted by the error of predicting a pixel from the observation of a feature vector through W and, consequently, the optimal operator Ψ_{opt} is always a W-operator. Thus, the optimization problem can be equivalently formulated in the space of functions defined on L^W, with joint random processes on (L^W, Y) and loss functions ℓ from $L \times L$ to $\Re^+$.

In practice, the distributions on (L^W, Y) are unknown and should be estimated, which implies in estimating $Er[\psi]$ and ψ_{opt} itself. When the window is small or the distribution has a probability mass concentrated somewhere, the estimation is easy. However, this almost never happens. Usually, we have large windows with non concentrated mass distributions, thus requiring prohibitive amount of training data.

An approach for dealing with the lack of data is constraining the search space. The estimated error of an operator in a constrained space can be decomposed as the addition of the error increment of the optimal operator (i.e., increase in the error of the optimal operator by the reduction of the search space) and the estimation error in the constrained space. A constraint is beneficial when the constraint estimation error decreases (i.e., w.r.t the estimation error in the full space) more than the error increment of the optimal operator. The known constraints are heuristics proposed by experts.

3 Window Design by Conditional Entropy Minimization

Information theory has its roots in Claude Shannon's works [5] and has been successfully applied in a multitude of situations. In particular, mutual information is a useful measure to characterize the stochastic dependence among discrete

random variables [6] [7] [8]. It may be applied to feature selection problems in order to help identifying good subspaces to perform pattern recognition [9] [10]. For instance, Lewis [11] explored the mutual information concept for text categorization while Bonnlander and Weigend used similar ideas for dimensionality reduction in neural networks [12]. Additional works that may also be of interest include [13] [14]. An important concept related to the mutual information is the mean conditional entropy, which is explored in our approach.

3.1 Feature Selection: Problem Formulation

Given a set of training samples T where each sample is a pair $(\mathbf{x}, \mathbf{y})$, a function ψ from L^n to $Y = \{1, ..., c\}$, called a *classifier*, may be designed. Feature selection is a procedure to select a subset Z of $\mathcal{I} = \{1, 2, ..., n\}$ such that $\mathbf{X_Z}$ be a good subspace of $\mathbf{X}$ to design a classifier ψ from $L^{|Z|}$ to Y.

The choice of Z creates a constrained search space for designing the classifier ψ. Z is a good subspace, if the classifier designed in Z from a training sample T has smaller error than the one designed in the full space from the same training sample T.

Usually, it is impossible to evaluate all subsets Z of $\mathcal{I}$. Two different aspects involve searching for most suitable subsets: a criterion function and a search algorithm (often based on heuristics in order to cope with the combinatorial explosion) [15]. There are many of such algorithms proposed in the literature and the reader should refer to [16] for a comparative review.

Next section explains how we explore the mean conditional entropy as a criterion function to distinguish between good and bad feature subsets.

3.2 Mean Conditional Entropy as Criterion Function

Let X be a random variable and P be its probability distribution. The *entropy* of X is defined as:

$$H(X) = -\sum_{x \in X} P(x) log P(x) \ , \tag{6}$$

with $log 0 = 0$. Similar definitions hold for random vectors $\mathbf{X}$. The motivation for using the entropy as a criterion function for feature selection is due to its capabilities of measuring the amount of information about labels (Y) that may be extracted from the features ($\mathbf{X}$). The more informative is $\mathbf{X}$ w.r.t. Y, the smaller is $H(Y|\mathbf{X})$. The basic idea behind this method is to minimize the conditional entropy of Y w.r.t the instances $\mathbf{x_{Z_i}}$ of $\mathbf{X_Z}$.

The criterion function adopted by the algorithm is the mean conditional entropy as described in [1] (Equation 7).

$$\hat{E}[H(Y|X_Z)] = \sum_{i=1}^{|L|^{|Z|}} \frac{\hat{H}(Y|X_{Z_i}) \cdot (o_i + \alpha)}{\alpha |L|^{|Z|} + t} \ , \tag{7}$$

where $\hat{H}(Y|\mathbf{X_{Z_i}})$ is the entropy of the estimated conditional probability $\hat{P}(Y|\mathbf{X_{Z_i}})$, o_i is the number occurrences of $\mathbf{X_{Z_i}}$ in the training set, t is the total

number of training samples, $|L|^{|Z|}$ is the number of possible instances of $\mathbf{X_Z}$ and α is a weight factor used to model $P(\mathbf{X_Z})$ in order to circumvent problems when some instances of $\mathbf{X_Z}$ are not observed in the training data. These non observed instances lead to prior entropy of Y ($\hat{H}(Y)$), which is slightly different from the criterion defined by [1] based on the entropy of the uniform distribution (maximum entropy).

Thus, feature selection may be defined as an optimization problem where we search for $Z^* \subseteq \mathcal{I}$ such that:

$$Z^* : H(Y|X_{Z^*}) = min_{Z \subseteq \mathcal{I}}\{\hat{E}[H(Y|X_Z)]\} \ , \tag{8}$$

with $\mathcal{I} = \{1, 2, ..., n\}$.

Dimensionality reduction is related to the U-curve problem where classification error is plotted against feature vector dimension (for an *a priori* fixed number of training samples). This plot leads to a U-shaped curve implying that an increasing dimension initially improves the classifier performance. Nevertheless, this process reach a minimum after which estimation errors degrades the classifier performance [15]. As it would be expected, the mean conditional entropy with α positive and conditional entropies of non observed instances conveniently treated reflects this fact, thus corroborating its use for feature selection [1].

4 Experimental Results

This section presents a method for texture classification that uses the SFFS algorithm with mean conditional entropy to design W-operators that classify

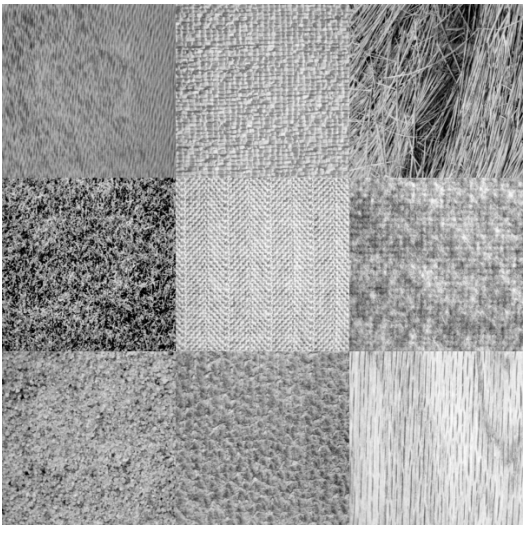

Fig. 1. Textures with 256 gray levels used in this experiment

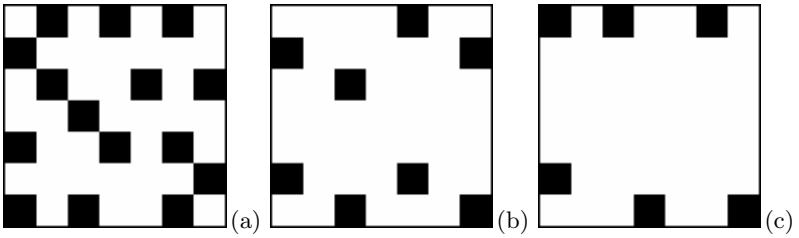

Fig. 2. Typical subwindows obtained using the textures of the Figure 1 to design the W-operator. (a) $k' = 2$, 20% of pixels to form the training set; (b) $k' = 4$, 20% of pixels to form the training set; (c) $k' = 8$, 40% of pixels to form the training set.

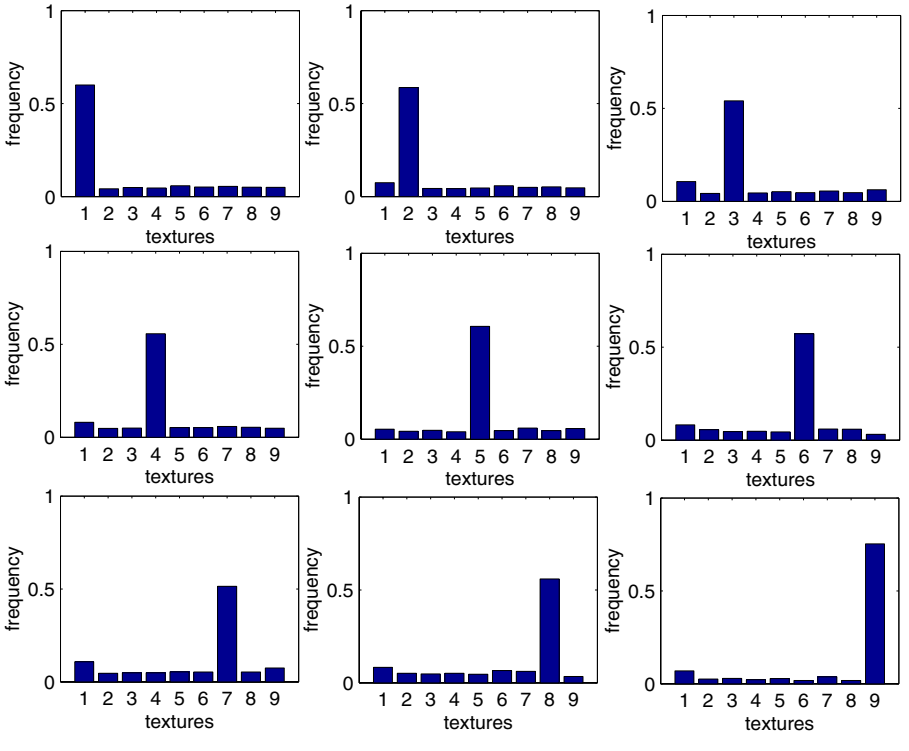

Fig. 3. Histograms of label frequency after the classification performed by the W-operator for each region of the Figure 1 (40% of pixels used to form the training set; $k' = 8$). The textures are numbered from 1 to 9 and the histograms are placed in raster order by these numbers.

gray-scale textures. Figure 1 shows an example containing 9 textures with 256 gray tones ($c = 9$ and $k = 256$).

The training set used to choose the window points and design the W-operator under this window is obtained from input textures. A window of fixed dimen-

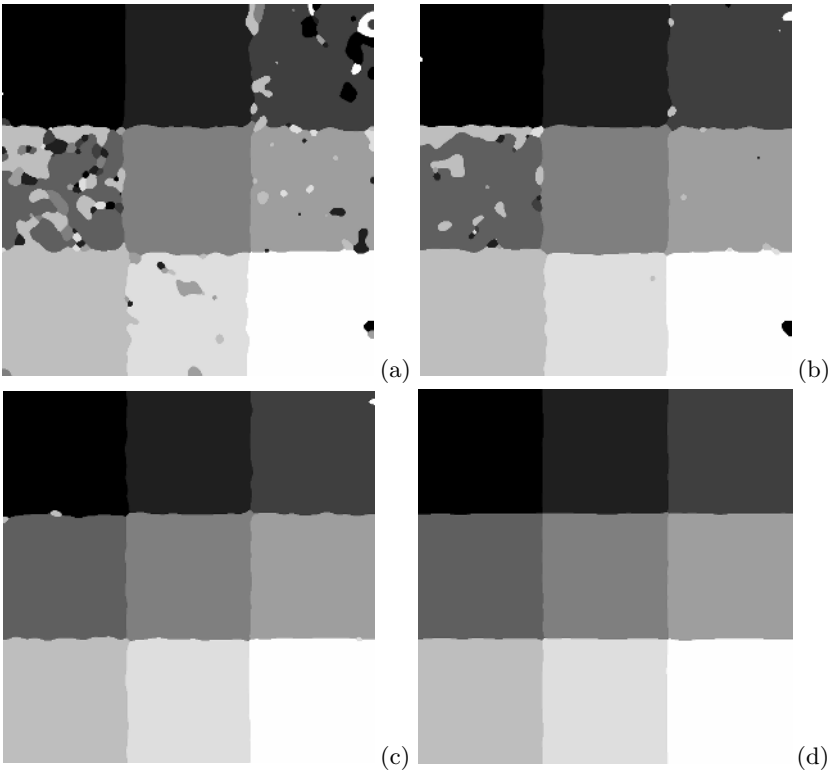

Fig. 4. Final results after applying the mode filters post-processing. (a) $k' = 2$, 10% of pixels to form the training set, MAE = 0.1020; (b) $k' = 2$, 20% of pixels to form the training set, MAE = 0.0375; (c) $k' = 4$, 20% of pixels to form the training set, MAE = 0.0095; (d) $k' = 8$, 40% of pixels to form the training set, MAE = 0.0037.

sions is centered at each selected pixel collecting the feature vector observed and its respective label (texture). Each feature vector is quantized in order to avoid excessive constraining in the space of W-operators that can be estimated adequately. Given a quantization degree $k' < k$, the lowest and highest gray levels observed in the considered feature vector form a interval which is divided in k' intervals of equal size. Then, these intervals are used to do the quantization of the collected feature vector. Thus, each quantized feature vector together with its label form a training sample.

The feature selection algorithm used to choose the window points is the Sequential Floating Forward Selection (SFFS). This algorithm has a good cost-benefit, i.e., it is computationally efficient and returns a very good feature subspace [4]. The criterion function used to drive this method is the mean conditional entropy as defined by Equation 7.

We have analyzed the MAE (Mean Absolute Error) obtained by application of our technique using as input nine textures presented in Figure 1 with increas-

Table 1. Average, standard deviation, minimum and maximum for MAE results after 10 executions for increasing number of training samples (% of pixels) and increasing quantization level k

		Training samples		
		10%	20%	40%
$k' = 2$	avg	0.0899	0.0345	0.0151
	±std	±0.0099	±0.0049	±0.0019
	min	0.0723	0.0281	0.0121
	max	0.1020	0.0420	0.0182
$k' = 4$	avg	0.0711	0.0097	0.0087
	±std	±0.0082	±0.0008	±0.0010
	min	0.0628	0.0085	0.0071
	max	0.0859	0.0110	0.0100
$k' = 8$	avg	0.0270	0.0176	0.0038
	±std	±0.0033	±0.0019	±0.0003
	min	0.0197	0.0157	0.0033
	max	0.0308	0.0218	0.0043

ing quantization degrees k' (2, 4 and 8), increasing number of training samples (10%, 20% and 40% of pixels of each texture randomly chosen) and a 7 by 7 window (49 features in total). The designed W-operator observes and quantizes the feature vectors through a subset of the window points (chosen by SFFS with mean conditional entropy) to label the pixel centered at this window. The results presented here took as the image test, the image of the Figure 1. Typical subwindows obtained are ilustrated by Figure 2.

In all cases, each region correponding to one of the textures received the correct label with significant majority. Figure 3 shows a histogram for pixel classification of the nine considered regions, using $k' = 8$ and 40% of the image to form the training data. These histograms do not take into account the undefined labels.

In order to remove the undefined labels and improve the final texture segmentation, one step of post-processing is proposed. This step is an application of the mode filter multiple times for decreasing window dimensions. The mode filter is a window-based classifier that translates a window over all pixels of the labeled image produced by the designed W-operator and attributes the most frequent label observed to its central pixel. We propose the application of mode filter to windows with the following dimensions in the same order as they appears: 15×15, 13×13, 11×11, 9×9, 7×7, 5×5, 3×3. Assuming that there are many more correct labels than incorrect ones (see Figure 3), this step helps to eliminate errors, although, depending on similarity among textures in certain regions, there is a risk to propagate errors.

Figure 4 presents the final texture segmentation result of the image presented by Figure 1, for 4 distinct pair values (k', % of training samples). Results obtained using the textures of the Figure 1 as input after 10 executions for each considered pair (k', % of training samples) are summarized in the Table 1. Note

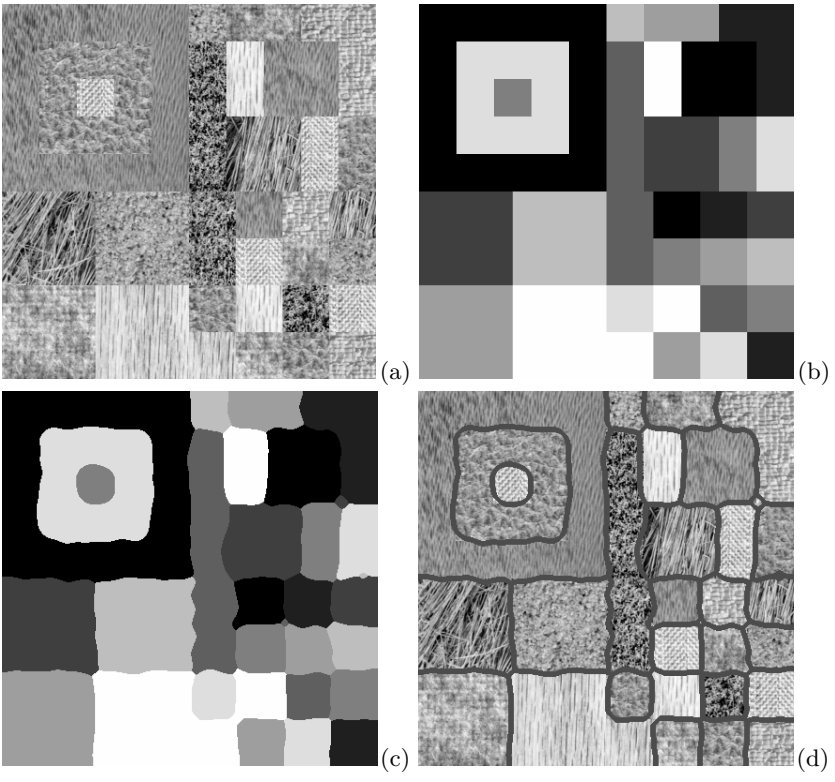

Fig. 5. (a) Mosaic of textures obtained from the Figure 1; (b) Correponding template of labels; (c) Final result using $k' = 4$ and 20% of pixels from the textures of Figure 1 (MAE = 0.0380); (d) Corresponding texture segmentation

that the results are satisfactory even taking small training samples to design the W-operators. Also is important to note that quantizations $k' = 4$ and $k' = 8$ lead to better results than those obtained by $k' = 2$, although this last quantization level already presents good results. Finally, a result obtained from the mosaic of the Figute 5(a) using $k' = 4$ and 20% of pixels from the textures of Figure 1 to design the W-operator is ilustrated by Figure 5(c), showing that our method is adequate for segmentation of small textures. Figure 5(b) shows its corresponding template of labels and Figure 5(d) shows the corresponding texture segmentation.

5 Concluding Remarks

This paper presents an extension for the design of W-operators from training data to be applied to gray-scale image analysis. A hypothesis for applying the presented approach is that the conditional probabilities of the studied pattern recognition problem have mass concentrated in one class when the problem has

a good solution. Experimental results with texture recognition have been presented.

The proposed technique is general and may be applied in a wide range of image processing problems besides texture segmentation, including document analysis and color image processing.

For the estimation of the conditional entropy it is required the estimation of the conditional probabilities $P(Y|\mathbf{X_Z})$ and the prior distribution $P(\mathbf{X_Z})$. The conditional probabilities are estimated based on simple counting of the observed classifications of a given feature vector. The entropy for $\mathbf{X_Z}$ is computed from the estimated distribution $\hat{P}(Y|\mathbf{X_Z})$. The distribution of $P(Y|\mathbf{X_Z})$ when $\mathbf{X_Z}$ is not observed in training set were considered uniform in [1]. But the conditional entropy $H(Y|\mathbf{X_Z})$ can not be higher than the entropy *a priori* of Y $(H(Y))$, since the information *a priori* about Y cannot decrease.

The parameter α in Equation 7 gives a determined probability mass for the non-observed instances. We have verified empirically that this parameter fixed as 1 leads to a very good balance between error due to noise in feature vector classification and estimation error. However, this parameter could be estimated from the training data in order to obtain better results. We are currently working on this problem to improve the proposed technique.

A branch and bound feature selection algorithm that explores the "U-curve" effect by our mean conditional entropy estimator [1] is under development. The goal is to obtain the optimal feature subspace in reasonable computational time. Results will be reported in due time.

Acknowledgements

The authors are grateful to FAPESP (99/12765-2, 01/09401-0 and 04/03967-0), CNPq (300722/98-2, 52.1097/01-0 and 468413/00-6) and CAPES for financial support. This work was partially supported by grant 1 D43 TW07015-01 from the National Institutes of Health, USA. We also thank Daniel O. Dantas by his complementing post-processing idea (mode filter applied more than once).

References

1. D. C. Martins-Jr, R. M. Cesar-Jr, and J. Barrera. W-operator window design by maximization of training data information. In *Proceedings of XVII Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI)*, pages 162–169. IEEE Computer Society Press, October 2004.
2. J. Barrera, R. Terada, R. Hirata-Jr., and N. S. T. Hirata. Automatic programming of morphological machines by pac learning. *Fundamenta Informaticae*, pages 229–258, 2000.
3. E. R. Dougherty, J. Barrera, G. Mozelle, S. Kim, and M. Brun. Multiresolution analysis for optimal binary filters. *J. Math. Imaging Vis.*, 14(1):53–72, 2001.
4. P. Pudil, J. Novovicov, and J. Kittler. Floating search methods in feature selection. *Pattern Recognition Letters*, 15:1119–1125, 1994.

5. C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656, July, October 1948.
6. T. M. Cover and J. A. Thomas. Elements of information theory. In *Wiley Series in Telecommunications*. John Wiley & Sons, New York, NY, USA, 1991.
7. S. Kullback. *Information Theory and Statistics*. Dover, 1968.
8. E. S. Soofi. Principal information theoretic approaches. *Journal of the American Statistical Association*, 95:1349–1353, 2000.
9. R. O. Duda, P. E. Hart, and D. Stork. *Pattern Classification*. John Wiley & Sons, NY, 2000.
10. M. A. Hall and L. A. Smith. Feature selection for machine learning: Comparing a correlation-based filter approach to the wrapper. In *Proc. FLAIRS Conference*, pages 235–239. AAAI Press, 1999.
11. D. D. Lewis. Feature selection and feature extraction for text categorization. In *Proceedings of Speech and Natural Language Workshop*, pages 212–217, San Mateo, California, 1992. Morgan Kaufmann.
12. B. V. Bonnlander and A. S. Weigend. Selecting input variables using mutual information and nonparametric density estimation. In *Proc. of the 1994 Int. Symp. on Artificial Neural Networks*, pages 42–50, Tainan, Taiwan, 1994.
13. P. Viola and W. M. Wells, III. Alignment by maximization of mutual information. *Int. J. Comput. Vision*, 24(2):137–154, 1997.
14. M. Zaffalon and M. Hutter. Robust feature selection by mutual information distributions. In *18th International Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 577–584, 2002.
15. T. E. Campos, I. Bloch, and R. M. Cesar-Jr. Feature selection based on fuzzy distances between clusters: First results on simulated data. In S. Singh, N. Murshed, and W. Kropatsch, editors, *Proc. ICAPR'2001 - International Conference on Advances in Pattern Recognition*, volume 2013 of *Lecture Notes in Computer Science, Springer-Verlag Press*, pages 186–195, Rio de Janeiro, Brasil, 2001.
16. A. Jain and D. Zongker. Feature selection - evaluation, application, and small sample performance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(2):153–158, 1997.