Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10–13, 2007, MCT/INPE, v. 2, p. 41–42.
http://urlib.net/dpi.inpe.br/ismm@80/2007/07.20.05.17

Exploring window selection strategies for two-level binary image operator design

Carlos da Silva dos Santos and
Roberto Hirata Jr.

Department of Computer Science
Institute of Mathematics and Statistics
University of São Paulo (USP), Brazil
csantos, hirata@ime.usp.br

1. Introduction

Statistical learning is a promising approach for designing image operators, since it avoids the trial-and-error process usually involved in manual design. This work addresses the design of translation invariant and locally defined binary image operators, which are characterized by a Boolean function over variables defined by an image window. In the supervised learning setting adopted here, operators are specified by a training sample of (*input*, *output*) image pairs. Window selection plays a critical role in estimating the joint distribution of sample images: while a small window will reduce operator performance, a too large window might compromise estimation precision. To deal with this problem, a two-level design approach was recently proposed [2] which is able to improve operator performance considerably. In this new approach, outputs from several operators (defined on diverse sub-windows) are combined in a second level-operator. Still, the selection of sub-windows remains a task for the experimenter and is highly dependent on his/her skill level. In this work we analyze novel strategies for automatic window selection for two-level design of operators, based on information theory concepts.

2. Binary image operator design

Let $(\mathbf{S}, \mathbf{I})$ be an (*input*, *output*) image pair. Following previous work [2, 4], we consider $(\mathbf{S}, \mathbf{I})$ to be generated by a jointly stationary random process. Given a n-point window W, we define $\mathbf{X} = \{X_1, \ldots, X_n\}$ as the random vector of input image pixels seen through W. Let Y be the output image pixel associated with $\mathbf{X}$. Designing an operator $\Psi(\mathbf{S}) = \hat{\mathbf{I}}$ involves estimating the probability distribution $P(\mathbf{X}, Y)$ from the training sample. A generalization algorithm is used then to synthesize a Boolean function $\psi(\mathbf{X}) = \hat{Y}$ which characterizes the operator. The synthesis is carried out by minimizing a loss function with respect to $P(\mathbf{X}, Y)$. For the two-level design, the above process is repeated k times for windows $W_1, \ldots, W_k \subset W$. A new random vector $\mathbf{X}^* = \{\psi_1(\mathbf{X}), \ldots, \psi_k(\mathbf{X})\}$ is formed and a second-level operator is generated by repeating the training process for $(\mathbf{X}^*, Y)$.

3. Strategies for window decomposition

In [4], a heuristic was proposed for building sub-windows $W_i \subset W$, based on Interaction Information [3]. The heuristic works by iteratively clustering elements from W while trying to control both information loss and redundancy caused by window partitioning. An alternative goal, sought here, is creating windows such that the joint posterior distribution can be factored:

$$P(\mathbf{X}_1, \ldots, \mathbf{X}_k, Y) = P(Y) \prod_{i=1}^{k} P(\mathbf{X}_i | Y) \qquad (1)$$

Where $\mathbf{X}_i$ is the random vector of image pixels seen through W_i. When equation 1 holds, the joint distribution $P(\mathbf{X}_1, \ldots, \mathbf{X}_k, Y)$ can be estimated by parts. As in [4], we replace estimation of $P(\mathbf{X}_i)$ with $P(\pi(\mathbf{X}_i))$ where π is the *parity function* for a vector of binary variables. This simplification is justified as vectors $\mathbf{X}_i$ can be built in a way that the use of parity functions cause no loss of information about the output [1]. Thus, we will seek windows which better approximate the following condition:

$$P(\pi(\mathbf{X}_1), \ldots, \pi(\mathbf{X}_k), Y) = P(Y) \prod_{i=1}^{k} P(\pi(\mathbf{X}_i) | Y) \qquad (2)$$

Now we define a iterative procedure that sequentially reduces the estimation error associated with assuming that condition 2 holds. The procedure employs the *conditional entropy* function [1]:

$$H(X|Y) = - \sum_{Y \in \{0,1\}} \sum_{X \in \{0,1\}} P(X, Y) \log P(X|Y) \qquad (3)$$

The input to the procedure is a sample of training pairs $(\mathbf{X}, Y)$, $\mathbf{X} = \{X_1, \ldots, X_n\} \in \{0,1\}^n$, $Y \in \{0,1\}$. Initially, we set $\mathbf{X}_i \leftarrow X_i$, for $i \in 1, \ldots, n$.

[1] Detailed proofs are provided in http://www.vision.ime.usp.br/~csantos/ismm2007

Proceedings of the 8th International Symposium on Mathematical Morphology,
Rio de Janeiro, Brazil, Oct. 10–13, 2007, MCT/INPE, v. 2, p. 41–42.
http://urlib.net/dpi.inpe.br/ismm@80/2007/07.20.05.17

Table 1. Mean Absolute Error (MAE) obtained in [4] with Interaction Information criterion and current result with Conditional Entropy criterion.

	MAE
Interaction Information	0.0817
Conditional Entropy	0.1320

At each iteration, we choose $i, j \in \{1, \ldots, n\}$, $i \neq j$. Define $\mathbf{X}_0 = \mathbf{X}_i \cup \mathbf{X}_j \setminus \mathbf{X}_i \cap \mathbf{X}_j$ (in a abuse of notation, here we treat $\mathbf{X}_i$ also as the set containing the pixels which make up the vector $\mathbf{X}_i$). Conditional entropies are then calculated:

$$\begin{aligned} H_0 &= H(\pi(\mathbf{X}_0)|Y) \\ H_i &= H(\pi(\mathbf{X}_i)|Y) \\ H_j &= H(\pi(\mathbf{X}_i)|Y) \end{aligned}$$

If the maximum entropy corresponds to H_0 we just advance to the next iteration. Otherwise we set: $\lambda = \arg\max_{i,j}(H_i, H_j)$ and update $\mathbf{X}_\lambda \leftarrow \mathbf{X}_0$. This process is repeated till a specified number of iterations is reached.

As the conditional entropies $H(\pi(\mathbf{X}_i)|Y)$ are reduced at each step, the distribution $P(\pi(\mathbf{X}_1), \ldots, \pi(\mathbf{X}_k)|Y)$ is more concentrated than $P(X_1, \ldots, X_k|Y)$. It follows that, in the space span by parity functions, vectors $\mathbf{X}$ with the same classification ($Y = 0$ or $Y = 1$) are represented more *compactly*. This might have interesting consequences for designing operators in the space of parity functions.

4. Experimental results

We designed a two-level operator in a texture recognition task, using the above procedure to generate sub-windows W_i (or equivalently, their associated vectors $\mathbf{X}_i$). Figure 1 shows an example of input, ideal and resulting images obtained. Table 1 compares current results with those obtained with the Interaction Information heuristic for window selection [4], in terms of Mean Absolute Error (MAE). In this dataset Interaction Information provides considerably better results. Further experiments are necessary to assess the relation between these two methods.

5. Concluding remarks

The current work extends a line of investigation started in [4] about information theoretic methods for window selection. A new criterion was proposed,

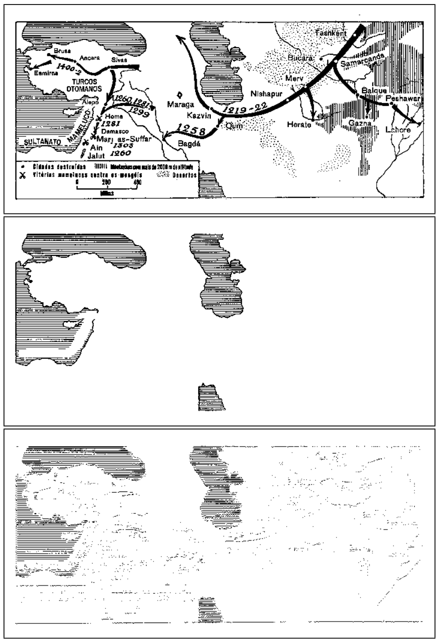

Figure 1. Result of texture recognition. Test, ideal and two-level operator result.

which minimizes the error involved in assuming a factorization of the posterior probability of image pixels. While the results obtained are inferior to the previous heuristic proposed in [4], the new method has interesting theoretical characteristics of its own (as stated in section 3) which justifies further investigation.

Acknowledgements

C. S. Santos acknowledges support from FAPESP (grant 05/04614-7) and from CNPq. R. Hirata Jr. acknowledges support from CNPq.

References

[1] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley and Sons, 1991.

[2] N. S. T. Hirata, *Binary Image Operator Design Based on Stacked Generalization*, Proceedings of the XVIII SIBGRAPI, 2005, pp. 63-70.

[3] W. J. McGill, *Multivariate information transmission*, Psychometrika **19** (1954), 97-116.

[4] C. S. Santos, N. S. T. Hirata, and R Hirata Jr., *Two-stage Binary Image Operator Design: an Approach Based on Interaction Information*, Proceedings of the XX SIBGRAPI, 2007.