A NOVEL SIMILARITY MEASURE FOR COMPRESSION and CLASSIFICATION

Yusuf Ozturk Hüseyin Abut
Department of Electrical and Computer Engineering,
San Diego State University,
San Diego, CA 92182-1309

ABSTRACT

In this study we propose a new architecture for texture classification based on pair-wise pixel associations as an extension of the recently developed Multivalued Recursive Network (MAREN) architecture. Maybe more critically we propose a novel similarity measure and classification algorithm to be used with this network. The proposed fidelity criterion has been observed to be tightly coupled with the ubiquitous mean-square error (MSE) distance measure. Both SOAR and MAREN structures can be considered extensions of the associative memory concept frequently used in neural networks. Our proposed similarity measure is based on the principle of directional divergence of interpixel relationships in a given texture and promises a number of advantages over the MSE measure. In this paper, SOAR will be discussed within the framework of a texture classification problem, but we believe it would be very easy to extend to other applications where interpixel relationship is the primary focus.

1. INTRODUCTION

The primary objective in texture identification is to find different type of patterns present in a given image, to identify them, and to separate them into different classes. Normally, this is achieved by extracting a texturally meaningful feature set with respect to a mathematically tractable and computationally efficient fidelity criterion in the form of a distance measure or a similarity function.

In the image processing literature various feature sets have been used with varying degree of success. Frequently used features include co-occurrence matrix [2], multi-channel multidimensional linear prediction coefficients [3], vector codebooks, neural networks, and wavelets. Almost all of the models proposed in the literature have an open-loop, sometimes, even off-line data collection and parameter extraction stage for obtaining respective parameters needed in later stages. After that, a search process is employed to find the best match for the parameters of a given texture block among a collection of predetermined templates stored in the system with respect to a performance measure. In some applications, there are provisions for rejecting all of the candidate templates or enlarging the size of the template memory. In a number of these cases, the problem is turned into a hypothesis-testing problem as it is done in the classical applied statistics. Various feature sets, such as mean-square error (MSE), mean-absolute difference (MAD), spectral distortion measures, and their extensions, have been successfully used in the identification stage.

Here we propose a System of Associative Relationships (SOAR) to learn the underlying structure in a given texture. This architecture is an extension of a recently developed technique called Multivalued Recursive Network (MAREN)[1]. MAREN is a novel integer-valued recurrent, nonlinear associative memory structure inspired from Hopfield and Tank's binary associative memory [4]. As in all other pattern matching techniques, the SOAR needs a fidelity measure in the texture identification stage. Instead of using the traditional difference-based measures, we propose to use a new similarity measure, which attempts to emphasize the similarity between two patterns, rather than the distance between them. It is an integer quantity and it will not involve any costly operations such as multiplication or matrix inversion. It will further be demonstrated that the new measure is tightly coupled with the traditional mean-square error measure. Even though, this measure is developed for texture identification it can be easily extended to applications in motion estimation, data compression based on popular vector quantization and others.

2. SYSTEM of ASSOCIATIVE RELATIONS (SOAR)

As we have stated above, the system of associative relations makes use of the pair-wise pixel associations. The pixels defined by a token over an image are used for determining associations among pixels. An analysis token is basically an irregular mask that will determine the underlying structure of associations. The size and shape of the token will depend on the target application. The token can be simply a 3x3 window centered at a pixel whose associations are explored. It can be of any shape such as a T or a circular window. In Figure 1, we present two possible token shapes that can be used to specify the region of interest for associations.

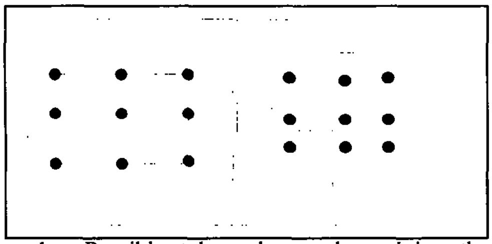

Figure.1 : Possible token shapes determining the region of interest for recording associations between pixels.

Once we decide on the token shape we can proceed with the definition of the SOAR architecture. In this work, we will use rectangular windows of 3x3 , 4x4 and 8x8. Since the structure of SOAR does not depend on the shape of the chosen token, it can be readily used for other token shapes.

To explore the associations among pixel elements defined by the token we map each pixel to a processing node and let the internal status of this processing node be equal to the intensity of that particular pixel. For instance, the token in Figure1.a will require 9 processing nodes, whereas the token in Figure1.b needs 5 processing elements. In the next step, we present a communication model or an interconnection model for a network of processing elements each representing one pixel within the token. For completeness let us assume that each processing element is connected to all other processing elements by unidirectional links. For the token defined in Figure 1.a, we will need 72 unidirectional links. For reasons that will be apparent later, we need only half of these links. The unidirectional links between processing elements –inherently those of pixels-- will store the associations between pixel pairs. The architecture defined in this way is in a sense similar to that of Hopfield [4], where each processing element is connected to all others with connection strengths *T(i,j;k,l)*.

Suppose pixel intensities in a block (token) are : { V_{ij}; i,j=1,2,....q }. We can encode the larger/smaller information called the "interpixel connection strength" between pixels in this block as the signs of differences as

$$T(i,j;k,l) = \operatorname{sgn}\{V_{ij}, V_{kl}) \quad \text{where } 1 \le i,j,k,l \le q \quad \text{except}: (i,j) \ne (k,l) \qquad (1)$$

The signum function here is the ternary sign function of discrete mathematics:

In (1) we have formulated a prescription to encode interpixel relationships in a block. Similarly we can define an ensemble interpixel association over P number of blocks, which can be the blocks obtained by sliding the token over the entire image.

$$T_P(i,j;k,l) = \sum_{p=1}^{P} \operatorname{sgn}(V_{ij}^p - V_{kl}^p) \qquad (2)$$

As it can be seen from (2) the connection strength T(i, j; k, l) between the pixel at location *(i,j)* and the one at *(k,l)* increases if $V_{ij} > V_{kl}$; but it decreases if $V_{ij} < V_{kl}$, and remains unchanged if $V_{ij} = V_{kl}$. Furthermore, we would like to point out that the connection strengths are anti-symmetric, i.e., *T(k,l;i,j)=-T(i,j;k,l)*.

3. SIMILARITY MEASURE

Suppose a token has been placed at any position in a given image and the pixel associations between pixels have been determined using (1). In this case, we can define the similarity measure among the initial token which has been stored in connections *T(i,j;k,l)* and a new token at another location by:

$$E = \sum_i \sum_j \sum_k \sum_l T(i,j;k,l) * \operatorname{sgn}(V_{ij} - V_{kl}) \qquad (3)$$

If we use (2) for the term *T(i,j;k,l)* in (3) we can write:

$$E = \sum_i \sum_j \sum_k \sum_l \operatorname{sgn}(V_{ij}^1 - V_{kl}^1) * \operatorname{sgn}(V_{ij} - V_{kl}) \qquad (4)$$

It is not difficult to deduce from (4) that E will attain its maximum value if the two patterns are correlated with a correlation coefficient of "1." But it will reach its minimum value if the two patterns are correlated with a correlation coefficient of "–1." Finally, E will have a value near zero if the two patterns are not correlated. That is, there is no resemblance between ordering of the pixels within the token. It can also be seen that the similarity measure we have just proposed is a *correlation indicator function* between two specific patterns. But as we will show later, this similarity measure is more than a correlation indicator.

Now let us assume a set of P patterns are stored in the system using (2). Next, the token is placed at some arbitrary point on the image plane. Here the question will be the similarity of this new pattern to any one of the patterns stored in the previous step. We can extend the similarity function to search for an ensemble of patterns stored in connections T_P(i,j;k,l) via:

$$E = \sum_p \sum_i \sum_j \sum_k \sum_l T_p(i,j;k,l) * \operatorname{sgn}(V_{ij} - V_{kl}) \qquad (5)$$

When we substitute (2) in (5) we have a more explicit form of the similarity function:

$$E = \sum_p \sum_i \sum_j \sum_k \sum_l \operatorname{sgn}(V_{ij}^p - V_{kl}^p) * \operatorname{sgn}(V_{ij} - V_{kl}) \qquad (6)$$

If the test pattern is equal to one of the stored patterns, say p_1, the similarity function will be close to its theoretical maximum, which is simply:

$$E_{max} = q^2 - q \qquad (7)$$

where q is the number pixels within a chosen analysis window.

This similarity measure is analogous to correlation in the case of single pattern comparisons. Furthermore, as the equation (6) indicate it would be more pertinent in the case of ensemble pattern associations. To test the dynamics of this similarity measure against the well-known mean square error (MSE) we have run a 4x4 token over the texture image of Figure 2 and computed the MSE and the proposed similarity measure values. Later we have computed the correlation of the errors generated by MSE and the outcome of the similarity measure over a sequence of token pairs. The results are shown in Figure 2, which uniformly indicate a very high degree of correlation between the MSE value and the SOAR similarity measure.

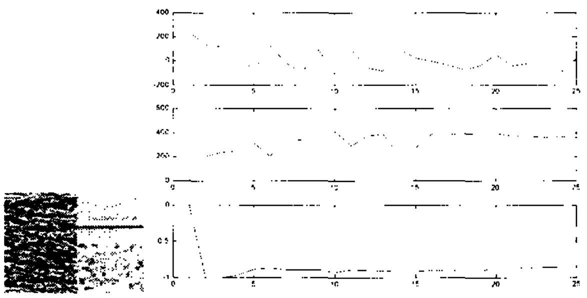

Figure 2. Left: the texture of size 32x32 pixels. Right: MSE and Similarity Measure Plots. Top: MSE, Middle: Similarity Measure and Bottom: Correlation Between the Two measures.

As stated above the SOAR similarity measure attains its maximum value when the two compared patterns have a similar ordering of the pixels. In addition, we would like to point out that the SOAR similarity measure is independent of the mean. That is, it does not matter how much bias exists in the data, the ordering between pixels for similar textures will be the same.

4. LEARNING AND CLASSIFICATION

Pattern classification methods have two key components. They are a meaningful distance or similarity measure that will summarize the inter-/intra-relations between objects or vectors and a method to incorporate similar objects into a single token that will represent the two objects reliably. A number of unsupervised classification/ clustering techniques including Generalized Lloyd Algorithm (GLA)[5] have been extensively studied in the literature and applied successfully to problems in data compression and pattern recognition. Here we present a learning scheme that will learn the interrelationships among tokens generated over a texture and will attempt to classify them into a predetermined number of classes.

During the design of the algorithm we have chosen to follow the same path as the generalized Lloyd algorithm and replaced the computational modules with new ones when necessary. Before explaining the proposed algorithm, we would like to point out its similarities in two areas to the generalized Lloyd algorithm. First, both algorithms are iterative in nature. Second, both start with a rate zero codebook and higher rate codebooks are then designed using a splitting technique. As in the data compression task, splitting technique can be replaced by one of many initial guess codebook assignments. We would like to emphasize that our proposed classification algorithm based on SOAR has no other similarities with the GLA and other iterative clustering techniques.

SOAR LEARNING ALGORITHM

Step 0: Assume that we are given a token shape and size, such as a 3x3 rectangular grid and the extend of connectivity like full connectivity or partial connectivity.

Step 1. Assuming P tokens are generated, set the number of codewords : Cword = 1 ; and compute

$$C_{p} = T_p(i, j; k, l) = \sum_{p=1}^{P} \mathrm{sgn}(V_{ij}^{p} - V_{kl}^{p}) \tag{8}$$

Step 2. Split the codebook by using two masks. The masks M_0 and M_1 should be logical complements of each other.

$$C_{p+Cword} = C_p \wedge M_1 \quad , \quad C_p = C_p \wedge M_0 \tag{9}$$

Cword = Cword * 2;

Step 3. For each token in the training set find the codeword which satisfies

$$E_{max} = \max\left[\sum_{i}\sum_{j}\sum_{k}\sum_{l} T_p(i, j; k, l) * \mathrm{sgn}(V_{ij} - V_{kl})\right] \tag{10}$$

Here p is the codebook index and we assign the token to the group satisfying (10)

Step 4. Once all tokens are assigned to one of the P classes, we compute an ensemble association among the patterns assigned to each cluster using (8) for each cluster center.

Step 5. Compute the Mean of the similarity values for this iteration. If the Mean is not significantly different than its predecessor go to step 3. That is,

$$\left[\frac{E_{av}^{current} - E_{av}^{previous}}{E_{av}^{previous}}\right] < \delta \tag{11}$$

Where the threshold δ is a stopping criterion as in the Generalized Lloyd Algorithm.

Step 6. If the number of codewords has reached the maximum size allowed then exit with this codebook. Otherwise, go to step 2.

To test the classification system we have used segments of texture images and brain images. We have choosen a token size of 3x3 and full connectivity among the pixels within the token. The token has been slided over the image to cover all the pixels in the image. We have generated one token per pixel in the image where the pixel under estimation is at the center of the token. The tokens generated have later been classified into 2 classes, 4, classes and 8 classes. In Figure 3 we present the results of this experiment with different images.

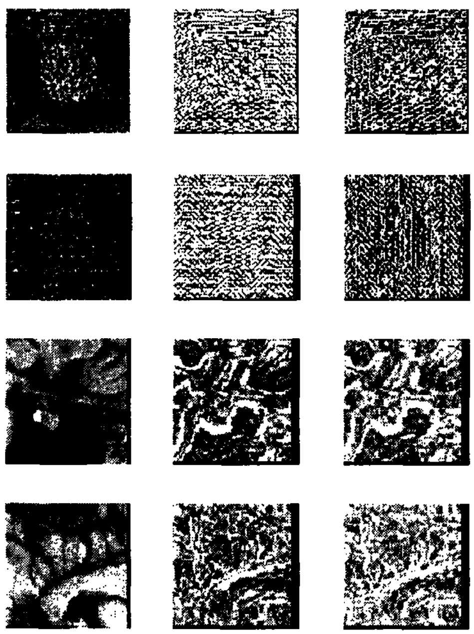

Figure 3: Texture synthesis using a token of size 3x3 and codebook sizes into 2, 4, 8 and 16 classes. (Top row: original image, synthesized with 2 classes, synthesized with 4 classes. Second row: original image, synthesized with 2 classes, synthesized with 4 classes. Third row: original image, classified into 4 classes, classified into 8 classes. Fourth row: original image, classified into 4 classes, classified into 8 classes)

As in all image classsification techniques, the success of the algorithm or its structural dynamics can not be achieved by visual observations. The conclusion of that type would be incomplete and may signal erroneous messages. To explore the dynamics of our learning algorithm, we have traced carefully the signature of similarity measurements over ten iterations during the codebook design stage. The change in the mean similarity ,as a function of number of iterations, throughout the classification process is shown in Figure 4. As it can be seen from this plot the mean similarity has a smooth increase during the iterations with the same codebook and has jumps when the codebook is split. This is the same dynamic one would observe with other clustering/classification systems with iterative nature. Using this dynamics we can conclude that the clustering system is converging to a solution.

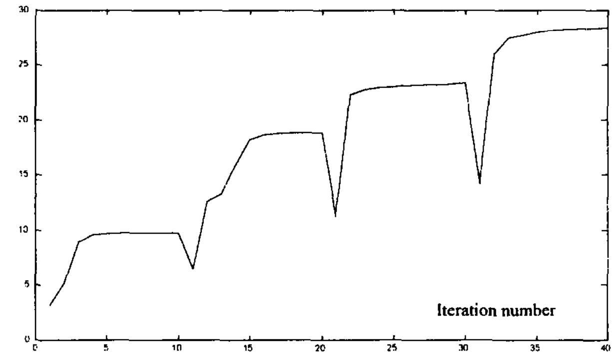

Figure 4: Dynamics of the Classification/Learning Algorithm (Mean similarity vs. number of iterations)

Conclusion and Future Work

Here we have proposed a novel method for feature extraction based on interpixel relationships. We have also developed a similarity measure, which tends to be tightly coupled with the MSE distance. We have developed a novel clustering algorithm that stores associations between pixels in an ensemble of texture blocks. Although we feel, the foundations are properly established, there are a number of areas to be explored including the selection of different tokens, their sizes and shapes. We are also interested in multilevel segmentation by using tokens with different association ranks, distributing the information stored by a single large cluster center onto a number of sub-cluster centers and in providing a means of robustness.

References

[1] M. Erdem and Y. Ozturk, "A New Family of Multivalued Networks ", Journal of Neural Networks, Vol. 9, No.6, pp. 979-989, August 1996.

[2] R.M. Haralick and L.G. Shapiro, *Computer and Robot Vision*, Addison Wesley, 1992.

[3] Y. Ozturk and H. Abut, "Applications of Itakura-Saito Type Spectral Distortion Measures to Image Analysis and Classification", Proceedings of the Twenty-fourth Asilomar Conference on Signals, Systems & Computers, November 5-7, 1990 Pacific Grove, California.

[4] J.J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Abilities", Proceedings of Nat. Ac. of Sci. USA, Vol. 79, April 1982.

[5] A. Gersho and R. M. Gray, Vector *Quantization and Signal Compression*, Kluwer-Academic Publishers, Norwall, MA. 1992.