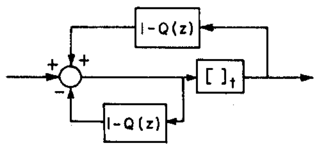

Fig. 6. A more general class of ESSQ.

data length, the HP-ESSQ does have speed advantage in this case.

IV. Conclusion

In this correspondence we have presented two error spectrum shaping quantizers which were shown to improve the performance of narrow-band filters. They are examples of the more general class of error spectrum shaping quantizers given in Fig. 6. The function $Q(z)$ in Fig. 6 may be chosen to minimize (15). However, except for the two cases discussed in this correspondence, it is not economical to implement these ESSQ's.

References

[1] B. Liu, "Effects of finite word length on the accuracy of digital filters–A review," *IEEE Trans. Circuit Theory*, vol. CT-18, pp. 670–677, Nov. 1971.

[2] T. Ishiguro and H. Kaneko, "A nonlinear DPCM codec based on ΔM/DPCM code conversion with digital filter," in *Conf. Rec., IEEE Int. Conf. Commun.*, 1971, pp. 1-27-1-32.

[3] T. Ishiguro, T. Oshima, and H. Kaneko, "Digital DPCM codec for TV signals based on ΔM/DPCM digital conversions," *IEEE Trans. Commun.*, vol. COM-22, pp. 970–976, July 1974.

[4] T. Claasen and L. Kristiansson, "Improvement of overflow behavior of 2nd-order digital filters by means of error feedback," *Electron. Lett.*, vol. 10, pp. 240–241, June 1974.

[5] L. B. Jackson, J. F. Kaiser, and H. S. McDonald, "An approach to the implementation of digital filter," *IEEE Trans. Audio Electroacoust.*, vol. AU-16, pp. 413–421, Sept. 1968.

[6] R. A. Gabel, "A parallel arithmetic hardware structure for recursive digital filtering," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-22, pp. 255–258, Aug. 1974.

[7] A. Peled and B. Liu, "A new hardware realization of digital filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-22, pp. 456–462, Dec. 1974.

[8] Trần-Thống and B. Liu, "A recursive digital filter using DPCM," *IEEE Trans. Commun.*, vol. COM-24, pp. 2–11, Jan. 1976.

[9] L. B. Jackson, "An analysis of roundoff noise in digital filters," Ph.D. dissertation, Stevens Inst. Technol., Hoboken, NJ, 1969.

A Class of Translation Invariant Transforms

M. D. WAGH AND S. V. KANETKAR

Abstract—A class of translation invariant transforms containing the R-transform is defined, and it is shown that a particular member of this class is superior to the R-transform for pattern recognition applications.

Rapid advances in digital technology have stimulated a great interest in the transforms of discrete sequences. Transforms which do not change with cyclic shifts in the sequences are called translation invariant and are useful for pattern recognition. Examples of transforms of this kind are the cyclic autocorrelation of a sequence, the modulus of the DFT of a sequence (the square root of its power spectrum) [1], and the power spectrum of the BIFORE transform of a sequence [2]. In this note, an infinite class of transforms is defined and its translation invariance is proved. Since this class is infinite, "best" transforms could conceivably be selected for different applications where emphasis may be on different properties such as speed, hardware realizability, memory requirements, etc.

Manuscript received April 10, 1975; revised September 17, 1975, May 18, 1976, and December 14, 1976.

The authors are with the Department of Electrical Engineering, Indian Institute of Technology, Bombay, India.

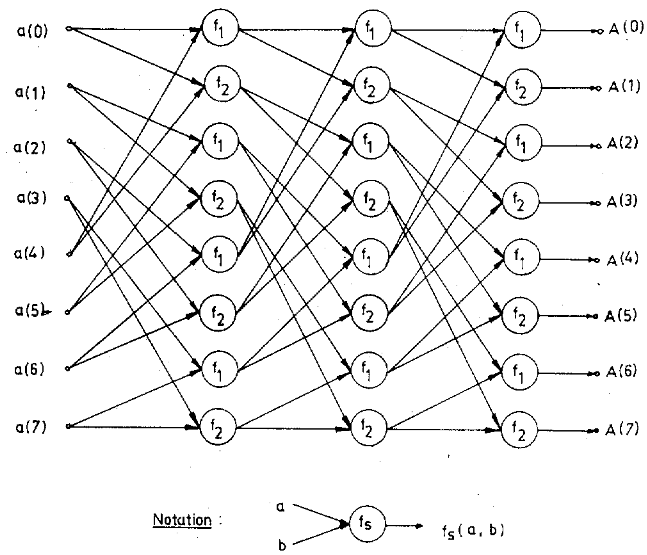

Fig. 1. Algorithm for calculating transform sequence $A(K)$ from the input sequence $a(I)$ of eight components. f_1 and f_2 are any two symmetric functions.

A function $f(a, b)$ in two variables is said to be symmetric if $f(a, b) = f(b, a)$, e.g., $f(a, b) = a + b$, $|a - b|$, max (a, b), min (a, b), $a^2 + b^2$, ab, etc. Moreover, when a and b are binary, logical functions, such as $a \cdot$ AND $\cdot\, b$, $a \cdot$ OR $\cdot\, b$, etc., are also symmetric functions. Each transform in the class of translation invariant transforms reported in this note is based upon a pair of symmetric functions, as the following definition shows.

Definition: For a pair of symmetric functions f_1 and f_2 and a given sequence $a(I)$, $I = 0, 1, \cdots, 2^n - 1$ of 2^n elements, the Kth component of its transform $A(K)$, $K = 0, 1, \cdots, 2^n - 1$ can be obtained as follows.

Let the n-bit binary expansion of K be $k_0 k_1, \cdots, k_{n-1}$; then compute sequences $y_0(I), y_1(I), \cdots, y_n(I)$ of $2^n, 2^{n-1}, \cdots, 2^0$ elements, respectively, as

$$y_0(I) = a(I)$$

and

$$y_{r+1}(I) = f_s(y_r(I), y_r(I + 2^{n-(r+1)}))$$

where

$$\begin{aligned} f_s &= f_1, \quad \text{if } k_r = 0 \\ &= f_2, \quad \text{if } k_r = 1, \qquad \text{for } r = 0, 1, \cdots, n-1. \end{aligned}$$

The Kth component of the transform is then obtained as

$$A(K) = y_n(0).$$

Although this definition expresses an isolated transform component, an efficient algorithm for the calculation of all the transform samples simultaneously can easily be devised, as shown in Fig. 1. The translation invariance of this transform

TABLE I
A COMPARISON OF THE R-TRANSFORM AND THE M-TRANSFORM FOR BINARY INPUT PATTERNS

Sequence Length	Number of Distinct Patterns		Transform Volume (bits)		Computation Time[a] in a General-Purpose Digital Minicomputer, HP 2100 (μs)	
	RT	MT	RT	MT	RT	MT
4	6	6	8	4	337.12	15.68
8	21	20	20	8	1011.36	47.04
16	86	168	48	16	2696.96	125.44

[a]HP 2100 average times for relevant operations (in BASIC)
Addition: 41.65 μs
Subtraction: 42.63 μs
AND/OR 1.96 μs

is proved in the following theorem.

Theorem 1: $A(K)$ is invariant under cyclic shifts in $a(I)$.

Proof: Let the length of $a(I)$ be 2^n. We proceed by the method of mathematical induction over the exponent n. When $n = 1$, $a(I)$ has only two elements and the translation invariance of $A(K)$ follows from the symmetry of f_1 and f_2. If the theorem is true in the case of sequences of length 2^{n-1}, its truth in the case of sequences of length 2^n can be established as follows. For a sequence of length 2^n,

$$\begin{aligned} y_1(I) &= f_s(y_0(I), y_0(I+2^{n-1})) \\ &= f_s(a(I), a(I+2^{n-1})). \end{aligned}$$

From this, it can be seen that a cyclic shift in $a(I)$, i.e., $a(I) \to a((I+1) \bmod 2^n)$, is equivalent to a cyclic shift in $y_1(I)$ because for $I < 2^{n-1} - 1$,

$$\begin{aligned} y_1(I) &= f_s(a(I), a(I+2^{n-1})) \\ &\to f_s(a(I+1), a(I+1+2^{n-1})) = y_1(I+1) \end{aligned}$$

and

$$\begin{aligned} y_1(2^{n-1}-1) &= f_s(a(2^{n-1}-1), a(2^n-1)) \\ &\to f_s(a(2^{n-1}), a(0)) = y_1(0). \end{aligned}$$

From the definition, it follows that $A(K)$ is the $K \bmod 2^{n-1}$th sample of the transform of $y_1(I)$. Therefore, $A(K)$ is invariant under a cyclic shift in $y_1(I)$, a sequence of length 2^{n-1}, and is therefore invariant under cyclic shifts in $a(I)$.

It can be shown that this class of transforms is also invariant under inversion, as stated in the following theorem.

Theorem 2: $A(K)$ is invariant under inversion in $a(I)$.

Proof: If the length of $a(I)$ is 2^n, the inversion of $a(I)$ means $a(I) \to a(2^n - 1 - I)$. In this case then,

$$\begin{aligned} y_1(I) &= f_s(a(I), a(I+2^{n-1})) \\ &\to f_s(a(2^n-1-I), a(2^{n-1}-I-1)) \\ &= f_s(a(2^{n-1}-1-I), a(2^n-1-I)) \\ &\qquad \text{from symmetry of } f_s \\ &= y_1(2^{n-1}-1-I). \end{aligned}$$

But this is the inversion of $y_1(I)$ as this sequence has only 2^{n-1} elements. Thus, inversion in $a(I)$ corresponds to inversion in $y_1(I)$. The theorem can then be proved by arguments similar to that of Theorem 1. Q.E.D.

An important property of this class of translation invariant transforms is their recursive nature of calculations (see Fig. 1), similar to that of the FFT or fast Hadamard transformation. This simplifies both the software and hardware implementation.

The transform obtained by setting $f_1(a, b) = a + b$ and $f_2(a, b) = |a - b|$ is well known as the R-transform (RT) [3] and is found to be useful for pattern recognition [4]. Although the R-transform calculations are very rapid because of their dependence on addition and subtraction alone [3], this very dependence makes the hardware implementation difficult and costly. This is so firstly because the adder circuits used are not only costlier, but also slower compared to basic AND/OR gates. Further, it is known [5] that the different components of the RT exhibit different amplitude bounds even when the pattern component amplitude bounds are uniform. The ratio of the maximum to minimum amplitude bounds is as large as the length of the sequence. Generally, the transforms of all the possible patterns are stored, and the transform of the unknown pattern is compared with these stored transforms for its identification. The volume required for the transform storage is therefore an important consideration. If the transform storage is designed according to the individual component amplitude bounds, it occupies $(n+2) \cdot 2^{n-1}$ bits for binary patterns of length 2^n [6]. But in this case, the comparison of the transform of an unknown pattern and a stored transform is complicated because of the nonuniform representation of the transform components (with different amplitude bounds) in the storage. If uniform representation is used in the transform storage, the volume is even larger, 2^{2n} bits. Finally, the total number of distinct transforms is also an important factor because it determines the maximum number of patterns which can be distinguished from one another using that transform technique. For example, in the case of the RT, if the length of the binary sequence is 16 bits, even though there are 65 536 possible binary patterns, one can choose at the most 86 of these to form a valid pattern set (i.e., a set in which the patterns can be distinguished from one another on the basis of the RT) because there are only 86 distinct transforms.

The work reported in this note indicates that the arithmetical functions $a + b$ and $|a - b|$ which occur in the definition of the RT can be replaced by any other symmetric functions so that the transform has other desirable properties besides translation invariance. We investigate here the transform resulting from $f_1(a, b) = \max\{a, b\}$ and $f_2(a, b) = \min\{a, b\}$, to be denoted as the M-transform (MT), in the case of binary patterns. It should be immediately obvious that in the binary case, the functions f_1 and f_2 reduce to the logical operations $f_1(a, b) = a \cdot \text{OR} \cdot b$ and $f_2(a, b) = a \cdot \text{AND} \cdot b$ which are very simple and cheap to implement by hardware and are computed many times faster than the functions $a + b$ and $|a - b|$. Further, the transform components have a uniform amplitude bound of 1 bit, resulting in a smaller transform volume and also simpler transform comparison circuits. The performances of the RT and MT are compared in Table I for various sequence lengths $N = 2^n$. This table shows that even if the RT volume is based upon the individual component amplitude bounds, it is larger than the MT volume by a factor $1 + n/2$. The transform computation time in the case of the RT is 21.5 times as much as that in the case of the MT. Although these times are highly dependent upon the machine used, it is obvious that the evaluation of the logical operations AND/OR will always be much faster than the arithmetical operations of addition or subtraction. The number of distinct transforms in the case of the MT (determined by computer search) is also nearly equal to that of the RT (from [3]) when $N = 4$ and 8 and about double when $N = 16$.

The MT can also be extended to two-dimensional patterns, as has been done in case of the RT [3], [7]. It is thus clearly superior to the RT and can replace the RT in most binary pattern recognition work.

References

[1] J. W. Cooley and J. W. Tukey, "An algorithm for the machine computation of complex Fourier series," *Math. Comput.*, vol. 19, pp. 296–301, 1965.
[2] N. Ahmed and K. R. Rao, "A phase spectrum for binary Fourier representation," *Int. J. Comp. Math.* (Sect. B), vol. 3, pp. 85–101, 1971.
[3] H. Reitboeck and T. P. Brody, "A transform with invariance under cyclic permutation for applications in pattern recognition," *Inform. Contr.*, vol. 15, pp. 130–154, 1969.
[4] P. O. Wang and R. C. Shiau, "Machine recognition of printed Chinese characters via transformation algorithms," *Pattern Recognition*, vol. 5, pp. 303–321, 1973.
[5] P. S. Moharir, "Translation invariant transform with uniform amplitude bounds for pattern recognition," *J. Inst. Telecommun. Eng.* (India), vol. 19, pp. 163–164, 1973.
[6] M. D. Wagh, "*R*-transform amplitude bounds and transform volume," *J. Inst. Telecommun. Eng.* (India), vol. 21, pp. 501–502, 1975.
[7] M. D. Wagh and S. V. Kanetkar, "A multiplexing theorem and generalization of *R*-transform," *Int. J. Comp. Math.* (Sect. B), vol. 5, pp. 163–171, 1975.

Corrections to "Letter-to-Sound Rules for Automatic Translation of English Text to Phonetics"

HONEY S. ELOVITZ, RODNEY JOHNSON, ASTRID McHUGH, AND JOHN E. SHORE

In the above paper[1] the last sentence in the abstract should read: "It gives overall performance figures and detailed statistics showing the importance of each rule."

On page 448, the two displayed rules in column 1 should be ' C[O]M = /AA/' and ' : [E] = /IY/'. In column 2, the rule on line 6 should be ' [RE] Λ#=/R IY/', and the rule in the first line of the fourth full paragraph should be ' [O] = /OW/'.

In Table VIII, the following headings should be inserted above the eight columns of numbers:

Most Frequent 8000 Words				1000-Word Sample of Low-Frequency Words			
Number of Words Matched		Total Frequencies of Words Matched		Number of Words Matched		Total Frequencies of Words Matched	
Abs.	Relative (%)	Abs.	Relative (%)	Abs.	Relative (%)	Abs.	Relative (%)

Manuscript received January 1, 1977.
The authors are with the Naval Research Laboratory, Washington, DC 20375.
[1]H. S. Elovitz, R. Johnson, A. McHugh, and J. E. Shore, *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 446–459, Dec. 1976.

Correction to "Real-Time Adaptive Linear Prediction Using the Least Mean Square Gradient Algorithm"

In the above paper,[1] the photograph of Dennis R. Morgan was inadvertently omitted. The complete biography and photograph appear below.

Dennis R. Morgan (S'63–S'68–M'69) was born in Cincinnati, OH, on February 19, 1942. He received the B.S. degree in electrical engineering from the University of Cincinnati, Cincinnati, OH, in 1965, and the M.S. and Ph.D. degrees in electrical engineering from Syracuse University, Syracuse, NY, in 1968 and 1970, respectively.

He is now a Senior Engineer at the Electronics Laboratory, General Electrical Company, Syracuse, NY, where he is involved in the analysis and design of signal processing systems.

[1]D. R. Morgan and S. E. Craig, *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 494–507, Dec. 1976.