the number of v-tuples which our method requires be investigated in order to determine the column designators of the representatives of all the nondegenerate equivalence classes. It is interesting to note that in at least one case, $v=4$ and $r=4$, no inspection of the rows obtained from any v-tuple is necessary since

$$\binom{2^{r-1}-1}{v}$$

equals Gilbert's number, (5).

In Table II, we present all of the rectangular arrays obtained when $r=4$ and $v=3$. Note that the 3-tuples (1, 2, 3), (1, 4, 5), (1, 6, 7), (2, 4, 6), (2, 5, 7), and (3, 4, 7) result in arrays having a pair of equivalent rows.

R. BIANCHINI
Ford Instrument Co.
New York, N. Y.
C. FREIMAN
IBM Res. Ctr.
Yorktown Heights, N. Y.
Formerly with Dept. Elec. Engrg.
Columbia University
New York, N. Y.

Further Results on the N-tuple Pattern Recognition Method*

Highleyman and Kamentsky, of Bell Telephone Laboratories, have given the results of some work[1] on the pattern recognition method (the n-tuple method) introduced by Browning and Bledsoe at the Eastern Joint Computer Conference, December, 1959.[2]

Evidently, Highleyman and Kamentsky did not understand that the parameter n (in the n-tuple method) should be chosen to best suit the particular data being read, because they used only $n=2$ in their computations. Some studies at Sandia Corporation in February, 1960, on these same data (provided by Highleyman) with $n=6$, 8 and 12 yielded results comsiderably different from those given. The result of some of this work is summarized in Tables I and II, where the numbers represent the per cent recognized.

The variability of the handwritten characters used in these studies is high and not well represented by the 50 alphabets used. For example, the last ten alphabets are considerably different than the first 40. For this reason, it will be necessary to have a much larger sample (perhaps 1000 alphabets) before one can decide with any certainty how successfully the n-tuple method will read characters with this much variability.

Other more powerful techniques, such as described in the original paper,[2] were not

TABLE I
HANDWRITTEN LETTERS AND NUMERALS

n	40 Alphabets of Experience 10 Alphabets Read (Different)		50 Alphabets of Experience 50 Alphabets Read (Same)	
	Not Centered	*Centered*	*Not Centered*	*Centered*
6	25	32	71	80
8	29	40	87	86
12	31	Not computed	98	Not computed

TABLE II
MACHINE PRINTED NUMERALS

n	40 Alphabets of Experience 10 Alphabets Read (Different)	40 Alphabets of Experience 40 Alphabets Read (Same)
	Not Centered	*Not Centered*
6	94	100
8	98	100

tried on these characters but would undoubtedly improve these percentages.

W. W. BLEDSOE
Advanced Research
Palo Alto, Calif.

* Received by the PGEC, September 16, 1960.

[1] W. H. Highleyman and L. A. Kamentsky, "Comments on a character recognition method of Bledsoe and Browning," IRE TRANS. ON ELECTRONIC COMPUTERS, vol. EC-9, p. 263; June, 1960.

[2] W. W. Bledsoe and I. Browning, "Pattern recognition and reading by machine," *Proc. EJCC*, pp. 225–232; December, 1959.

A Possibly Misleading Conclusion as to the Inferiority of One Method for Pattern Recognition to a Second Method to which it is Guaranteed to be Superior*

Highleyman and Kamentsky[1] recently reported having repeated portions of work done by Bledsoe and Browning[2] in a way that may lead the reader to what appear to be unfortunate conclusions. Bledsoe and Browning identified input patterns by matching the states into which they threw randomly chosen 2-tuples with similar lists of states for previously processed patterns. For each state, a 1 was stored if any example of the pattern in memory had ever thrown that 2-tuple into that state; otherwise, a 0 was stored. Bledsoe and Browning reported 78 per cent success with this method over an array of five different hand-printed alphabets. Highleyman and Kamentsky report only 20 per cent success over 50 different alphabets hand-printed by 50 different people. The second experiment does appear to indicate the weakness of this method, in its present state of sophistication, as soon as restrictions on transformations over the input matrix are relaxed. (In the first experiment, one presumably friendly person, while in the second, 50 quite likely skeptical people, prepared the input materials.) This is precisely what Kirsch,[3] in his discussion of the first paper, had suggested.

But a comparison test was made with a second method which, on the same set of inputs, gave 77 per cent success. Not enough detail is given to make absolutely clear what this method was. But the report makes it sound like a method that would, in fact, have Bledsoe and Browning's success rate as a theoretical upper limit. This, of course, is not quite an accurate statement, as the empirical results indicate. But the point is that the superiority of the second method could not possibly lie in its basic logic, but must, therefore, result only from its retention and use of a greater amount of information about previously processed patterns in its memory lists. That is, information and correlation methods that could just as well be used by either method were used only by the second, and the improvement in the second guarantees at least as great improvement in the first. Thus, the comparative experiment does not show that one logic was superior to another, but does show that additional use of information by one method led to great improvements in results.

The Highleyman-Kamentsky procedure "... involves the comparison of an unknown input pattern ... to a set of average characters. The average characters are described by a set of 12×12 matrices (one for each character) in which each element represents the probability of occurrence of a mark in that element for the character which it represents." This statement is taken to mean that over the 50 examples of the alphabet given the program in its learning phase, for each cell in the input matrix, the percentage of times that cell had been filled by the input pattern was stored as its probability for that pattern. For the Bledsoe-Browning method, the one of the four possible states of the $N/2$ random 2-tuples of cells in the matrix was given a probability of 1 if *any* of the five examples threw that 2-tuple into that state. Thus, the differences between the two methods appear to lie in 1) looking at 1-tuples as opposed to 2-tuples, plus the extraneous differences (in that they could be used with either method), 2) computing probabilities on a scale with n intervals rather than possibilities on a scale with 2 intervals (possible vs never yet), 3) using 50 vs 5 previous trials, and 4) examining probabilities vs cross correlating.

But the first difference is not a difference at all for the state of two randomly conjoined cells is simply the state of the conjunction of the same two individual cells. The 2-tuple and 1-tuple methods should, in fact, give identical results in the case of one alphabet. When several variant alphabets are learned, the 2-tuple method is bound to be superior, simply because information is stored not only about which state of a single cell is produced by an input pattern but also about which state of a randomly chosen second cell is produced in conjunction with this first cell state by this pattern.

The Highleyman-Kamentsky method is, in fact, exactly the same as the "1-tuple method" as used by Bledsoe and Browning in their original experiments, and the methods of among others, Uttley and, probably, Rosenblatt. Bledsoe and Browning ran several comparison experiments, in which they found what could be predicted theoretically—that with one alphabet there is no differ-

* Received by the PGEC, October 8, 1960; revised manuscript received, October 27, 1960.

[1] W. H. Highleyman and L. A. Kamentsky, "Comments on a character recognition method of Bledsoe and Browning," IRE TRANS. ON ELECTRONIC COMPUTERS, vol. EC-9, p. 263; June, 1960.

[2] W. W. Bledsoe and I. Browning, "Pattern recognition and reading by machine," *Proc. EJCC*, pp. 225–232; December, 1959.

[3] R. A. Kirsch, "Discussion of problems in pattern recognition," *Proc. EJCC*, pp. 233–234; December 1–3, 1959.