Behavior Research Methods, Instruments, & Computers
1988, 20 (1), 37-40

COMPUTER TECHNOLOGY

VIDEOLOGGER: A computerized multichannel event recorder for analyzing videotapes

ROBERT M. KRAUSS, PALMER MORREL-SAMUELS, and JONATHAN HOCHBERG
Columbia University, New York, New York

A microcomputer-assisted system that permits multiple events to be coded from videotape to a common time base is described. The system allows an operator, by pressing a button, to record the onset and offset times of any number of events. It requires a video recorder, an Apple II microcomputer, a John Bell 6522 timing card, and a pair of switches. The software package consists of five programs: one to make timing signals on the videotape, one to record the onset time and duration of the operator's buttonpresses, one to read and print the coding data from disk, one to operate the timer, and one to process the compiled code used in the software package. All necessary wiring diagrams are shown. Software is available gratis from the authors on request.

Behavioral researchers often need to record the occurrence of a discrete event over time. For example, an investigator may want to record the frequency and onset time of barpresses or to extract quantitative data from the multiplicity of complex events that make up an ostensibly simple two-person interaction. As a general rule, provided the number of coding categories is small and the subtlety of the events limited, a conventional event recorder will do. As Verhave (1966) predicted, many researchers are now capitalizing on the ready compatibility of event recorders with microcomputers to collect this kind of data.

Indeed, reliance on event recorders has become so pervasive that statistical programs have been developed specifically for the observational data the recorders produce (e.g., Quera & Estany, 1984), and writers are addressing the methodological issues that uniquely pertain to this work (e.g., Duncan, Kanki, Mokros, & Fiske, 1984). Moreover, the diversity of this new generation of event recorders is noteworthy: systems have been designed that are portable (e.g., Bakeman & Brown, 1980), nonelectric (e.g., Bakeman & Helmreich, 1975), capable of processing impressively large amounts of information (e.g., Marshall-Goodell & Gormezano, 1985), inexpensive (e.g., Allen & Crawford, 1985), and especially suitable for instructional purposes in college lab courses (e.g., Balsam, Fifer, Sacks, & Silver, 1984).

We are grateful to Julian Hochberg and Lois E. Putnam for their helpful suggestions on software design, and to Rafael Gomez for constructing the switch circuits. The machine language COUNTER subroutine was adapted from one described by Rhodes (1982). The VIDEOLOGGER program was compiled with The Applesoft Compiler (TASC) by Microsoft.

Address correspondence to Robert M. Krauss, Department of Psychology, Schermerhorn Hall, Box 29, Columbia University, New York, NY 10027.

But event recorders, even when interfaced with microcomputers, are subject to a number of important limitations. Many of the currently available systems were developed for analyzing instrumental conditioning (e.g., Bozarth, 1983; Norman & Jongerius, 1985) and consequently were designed to record response frequency rather than amplitude. Operationally, this has meant that most computer-assisted recorders code events as though they were spike-like phenomena of negligible duration. In many cases (e.g., Arnold & Hastings, 1984), it is entirely suitable that only one output pulse is generated by each switch closure; however, as Fowler (1985) pointed out, this makes it difficult or impossible to extract data on graded responses such as force and duration. The problem is not always intractable, however, because in cases where recording graded responses is necessary transducers often can be used to represent the desired dimension. For example, Fowler (1985) used a force transducer and an A/D converter to measure the peak force and duration of operant responses on an Apple II.

A second, more serious problem stems from the fact that event recorders, computerized or not, must be able to accommodate occasional floods of incoming information. This is, in fact, the central problem for any multichannel event recorder, that is, one designed to monitor the state of more than one independent sensor or switch. Typically there have been three responses to this problem: (1) using a memory buffer; (2) limiting the number, burden or temporal resolution of the channels; and (3) printing a time code directly on the observational record. The first option typically interposes a FIFO (first in, first out) buffer between the sensors and the event recorder so that bursts of incoming information can be accommodated without exceeding the recorder's capacity (Annan & Savvides, 1977; Kaplan, 1978). The second approach is to use only one channel and/or to keep the number of coding categories low. In many cases this is quite practical; for example, Bakeman and Brownlee (1980) were able to analyze videotapes of children playing by using only five mutually exclusive coding categories. A variant of this approach is to keep the number of channels relatively high but to lower the temporal resolution of the system by having the event recorder monitor the state of each sensor less frequently. Campbell and Forest (1984) adopted this approach in their study of insect locomotion by using eight channels with the state of each infrared sensor being assessed only two times per second. The third option for dealing with this problem is to make an observational record of the behavior (e.g., a videotape or audiotape) and to stamp a time code directly on it. For example, one can place a timer next to the subject and in the camera's view (e.g., Gould & Salaun, 1986), or print the time every few seconds on an EEG or polygraph record (Ives, 1984), or mark each field of the videotape with the SMPTE time code, that is, a 32-bit binary-coded digit that is generated and read electronically and conforms to nationally established standards (e.g., Adamson & Bakeman, 1985; Bakeman & Adamson, 1984).

However, this third option, as proponents will readily admit, defers rather than solves the problem: the investigator must still extract the data from the EEG or magnetic tape, and this can be an extremely arduous process. If the researcher wants to retain the benefit of a time-stamped observational record and still use an event recorder, then both the observational record and the event recorder have to be coordinated to the same time base. For example, when each field of a videotape is assigned a unique number, the event recorder must have some way of reading that address so that the extracted data produced by the event recorder can retain the temporal information that pertains to each coded behavior. The need for this common time base becomes even more pressing if the number of behaviors to be coded exceeds the ability of the sensor (or experimenter) to acknowledge them during a single pass through the observational record.

It may appear that one could simply code the first behavior on the event recorder, rewind its chart paper, restart the video recorder and the event recorder at the same instant, and code the second behavior on a different channel. Unfortunately, this approach is not feasible, because the two independent mechanical drive systems—video recorder and chart recorder—would not remain in synchrony for very long, even if they are started simultaneously. Consequently, the records of the onset and offset of the two behaviors would fall out of synchrony in a relatively short time, and, as a result, the two records would be coded to different time bases. This will be true of any two mechanical systems that run independently.

In our efforts to solve this problem, we have developed a microcomputer-assisted system for coding visible events from videotape. Rather than give each video field its own electronic address, as SMPTE does, our system uses the second audio track to record a starting signal and a stopping signal, which mark the beginning and end of the video segment that is being coded. These signals can be placed at any point in the videotape so that one video field is designated as the temporal reference for all subsequent fields in the segment. Because the microcomputer in our system can read these signals and use them to reset an internal clock, it can store in its memory the onset time and duration of any keypresses made while the segment is being played. Unlike most SMPTE systems, ours can be installed in a microcomputer for about $200, it places virtually no limit on the number of events that can be coded, and, with widely available software, it can send its stored data directly to a mainframe computer for analysis.[1]

The software consists of five programs—SIGNAL MAKER, VIDEOLOGGER, READER, COUNTER, and RUNTIME[2]—which together make an integrated system. These programs, respectively, set stopping and starting signals on the audio track of a videotape, record the onset and offset time of buttonpresses, read and print the collected response data from disk, and control the computer's internal clock. The last program, RUNTIME, prepares the microcomputer so that VIDEOLOGGER can be run properly.

System Requirements

The software was designed to run on an Apple II series microcomputer with one disk drive. It utilizes a 6522 Parallel Interface card[3] in slot 3, serving as a counting timer (Rhodes, 1982). The 6522 card is directly interfaced with the videocassette recorder (VCR) through one of the recorder's audio-output jacks (which we will arbitrarily refer to as channel 1). If vocal content is of no interest, a recorder with a single audio channel having audio-dub capability (i.e., the ability to record a sound track without simultaneously erasing a previously recorded video signal) works satisfactorily. If the vocal content is to be preserved, a recorder with two audio tracks is necessary. If a recorder with automatic gain control (AGC) is used, it is desirable that the AGC be manually overridden. Two monitors are needed, one for viewing the videotaped material that is to be coded and the other for receiving instructions from the programs. Two shielded audio cables are required. One connects pin 3 and pin 11 (ground) on jumper #3 of the 6522 card to the audio output jack of channel 1. The other runs from annunciator 0 at pins 15 and 8 (ground) of the microcomputer's I/O port to the VCR's audio input for channel 1. Finally, two normally closed spring-loaded momentary-contact switches are needed to signal the observed presence of coded events. Their type is not particularly important, as long as they are comfortable to use and provide positive indication of when they are disengaged. Button 1 should introduce 56 kΩ of resistance into the circuit between pins 6 and 1 of the I/O port; button 2 should introduce 91 kΩ. These switches are read by the computer at the memory location reserved for paddle 0. The wiring configuration is

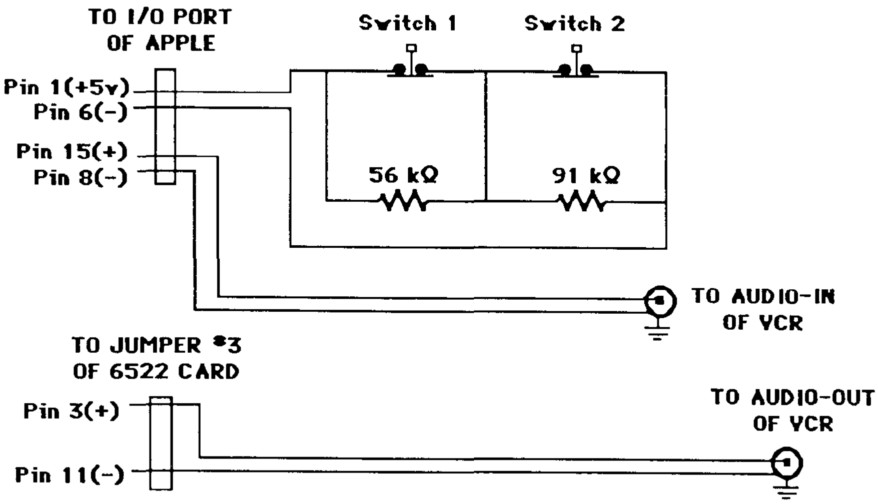

Figure 1. Wiring diagram for buttons and cables.

shown in Figure 1. The two switches permit two events to be coded simultaneously. If only one event is being coded, either switch may be used for input.

Procedure

The initial step in using the system is to mark on the audio track the beginning and end of the segment that is to be coded. The segment may be of whatever length the investigator desires, as long as the number of events to be recorded (onsets plus offsets) does not exceed 255 per button. If it seems likely that the number of events will exceed this value, the segment can be divided into two or more subsegments that can be combined when the data are analyzed. Before the start and stop signals can be recorded, all of the components and cables pictured in Figure 1 must be installed so that the microcomputer and VCR are fully interfaced.[4] Then the SIGNAL MAKER program is loaded. When running, this program prompts the operator to start the video recorder in the audio-dub mode. The press of any key on the Apple keyboard produces a start signal and a second keypress produces a stop signal, both of which are recorded on the selected audio track. The program allows the operator the option of having the stopping signal generated after a specified period of time.

After the tape has been rewound and the machine language programs COUNTER and RUNTIME have been B-loaded, coding may begin. This is done by running the program VIDEOLOGGER. The coder(s) should be seated in front of a video monitor connected to the video recorder with the switch(es) located in a comfortable position. When the VIDEOLOGGER program is running, it prompts the operator to start the video recorder. When the starting signal on the videotape is encountered, the program generates a cue that can be heard from the microcomputer's internal speaker and seen on its monitor, but not on the video monitor. The start signal resets and activates the timing circuits of the 6522 card, and, from that point on, the times of all buttonpresses and releases are recorded in the Apple's memory. Two events may be coded on each pass through the videotape by one or two operators.[5] This is accomplished by designating which event is to be coded by which switch and then pressing (and holding closed) that switch whenever the event occurs. When the stop signal is received from the video recorder, the system no longer records switch openings, and the data that has been recorded may be saved to disk for later analysis. The videotape can then be rewound, and two additional behaviors can be coded.[6] Because the timing signals reset the internal clock each time they are encountered, there is no limit as to the number of times a segment of tape can be recoded.

The final step is to recover the coded data from the disk. This is accomplished by means of the READER program, which permits a hard copy of the data to be produced by a printer connected to slot 1 of the Apple.[7] Data is furnished in the following format: The operation of the two switches is tabulated separately, and each switch opening is numbered sequentially. For every switch opening, the onset and offset time (relative to the starting signal) is provided both in milliseconds and in standard video frame numbers. This makes it relatively simple to compute distributional statistics and indices of cooccurrence directly from these data files. Of course, it is also possible to have the coding data uploaded to a mainframe computer for further analysis. (See Faraone & Dorfman, 1987, for an informed discussion of methods for analyzing data of this sort.)

Applications

VIDEOLOGGER was designed for coding patterns of cooccurrences of human nonverbal behaviors, but it can also be employed in a variety of other applications in which videotaped information must be quantified. It also should be noted that coders (or operators) need not be trained observers; the system can equally well be used to record the responses of naive subjects. For example, Newtson, Engquist, and Bois (1977) have studied how naive observers segment (i.e., break up into units) the complex motor acts of others, and Buck, Baron, and Barrette (1982) have had subjects indicate when videotaped facial expressions changed. VIDEOLOGGER permits 2 subjects to be run simultaneously on such tasks.

Investigators may obtain a copy of the five programs (including RUNTIME) by sending us an uninitialized 5¼-in. soft-sector single-sided double-density diskette, along with a stamped, self-addressed envelope suitable for returning the diskette.[8] We also are happy to provide upon request copies of the unoptimized BASIC source code (so that these programs can be converted to run on other microcomputers) and copies of the optimized source code. Readers who are interested in converting the VIDEOLOGGER programs to IBM-compatible machines are referred to Lien (1986) or a similar cross-index reference that could be used, in conjunction with our programs, to write new code.

REFERENCES

Adamson, L., & Bakeman, R. (1985). Affect and attention: Infants observed with mothers and peers. *Child Development*, **56**, 582-593.

Allen, J., & Crawford, L. (1985). Inexpensive microcomputer networking for laboratory research. *Behavior Research Methods, Instruments, & Computers*, **17**, 160-162.

Annan, A. G., & Savvides, M. (1977). An interface system for asynchronous data. *Behavior Research Methods & Instrumentation*, **9**, 274-276.

Arnold, N. D., & Hastings, L. (1984). A bit-addressable I/O interface for microcomputers. *Behavior Research Methods, Instruments, & Computers*, **16**, 135-136.

Bakeman, R., & Adamson, L. (1984). Coordinating attention to people and objects in mother-infant and peer-infant interaction. *Child Development*, **55**, 1278-1289.

Bakeman, R., & Brown, J. (1980). Early interaction: Consequences for social and mental development at three years. *Child Development*, **51**, 437-447.

Bakeman, R., & Brownlee, J. R. (1980). The strategic use of parallel play: A sequential analysis. *Child Development*, **51**, 873-878.

Bakeman, R., & Helmreich, R. (1975). Cohesiveness and performance: Covariation and causality in an undersea environment. *Journal of Experimental Social Psychology*, **11**, 478-489.

Balsam, P., Fifer, W., Sacks, S., & Silver, R. (1984). Microcomputers in psychology laboratory courses. *Behavior Research Methods, Instruments, & Computers*, **16**, 150-152.

Bozarth, M. (1983). A computer approach to measuring shuttle box activity and conditioned place preference. *Brain Research Bulletin*, **11**, 751-753.

Buck, R., Baron, R., & Barrette, D. (1982). Temporal organization of spontaneous emotional expression: A segmentation analysis. *Journal of Personality & Social Psychology*, **42**, 506-517.

Campbell, D. J., & Forest, J. (1984). The KIM-1 microcomputer as an eight-channel data logger. *Behavior Research Methods, Instruments, & Computers*, **16**, 315-319.

Duncan, S., Kanki, B., Mokros, H., & Fiske, D. (1984). Pseudounilaterality, simple-rate variables, and other ills to which interaction research is heir. *Journal of Personality & Social Psychology*, **46**, 1335-1348.

Faraone, S. V., & Dorfman, D. D. (1987). Lag sequential analysis: Robust statistical methods. *Psychological Bulletin*, **101**, 312-323.

Fowler, S. C. (1985). Amplitude measures of operant responses: Implementation with Apple Pascal. *Behavior Research Methods, Instruments, & Computers*, **17**, 301-306.

Gould, J., & Salaun, J. (1986). Behavioral experiments on handmarkings (IBM Research Report). *Research Communications*, 12083 (54449) Yorktown Heights, NY.

Ives, J. (1984). Time-scribe: A universal time writer for any EEG/polygraph chart recorder. *Electroencephalography & Clinical Neurophysiology*, **57**, 388-391.

Kaplan, H. (1978). An output spooling system for continuous datalogging paradigms. *Behavior Research Methods & Instrumentation*, **10**, 285-290.

Lien, D. (1986). *The BASIC handbook: An encyclopedia of the BASIC language* (3rd ed.). San Diego: Compusoft.

Marshall-Goodell, B., & Gormezano, I. (1985). Apple II/FIRST-Corvus networks. *Behavior Research Methods, Instruments, & Computers*, **17**, 168-172.

Newtson, D., Engquist, G., & Bois, J. (1977). The objective basis of behavior units. *Journal of Personality & Social Psychology*, **35**, 847-864.

Norman, W., & Jongerius, J. (1985). Apple Picker: Computer software for studying human responding on concurrent and multiple schedules. *Behavior Research Methods, Instruments, & Computers*, **17**, 222-225.

Quera, V., & Estany, E. (1984). ANSEC: A BASIC package for lag sequential analysis of observational data. *Behavior Research Methods, Instruments, & Computers*, **16**, 303-306.

Rhodes, N. (1982, March). John Bell Engineering's Apple II parallel interface board. *Byte*, pp. 414ff.

Verhave, T. (1966). *The experimental analysis of behavior*. Englewood Cliffs, NJ: Prentice-Hall.

NOTES

1. This is considerably less expensive than a comparable system using SEMPTE time code would be. For example, the Four-A Model TGR 3300, a typical time code generator and reader (Four-A Corp., 320 Nevada St., Newton, MA 02160) costs $6,500.

2. RUNTIME is copyright by Microsoft and used by permission.

3. John Bell Engineering, Inc., Box 338, Redwood City, CA 94064. For this application it is necessary that pins 7 and 8 on jumper #2 be connected.

4. If the selected audio track contains a previously recorded signal, it must be erased before the track can be used for timing. On most machines, recording the timing signal automatically erases whatever else the track contains, but on some that permit overdubbing it may be necessary to run the tape through the machine in the audio-dub mode with a dummy plug in the input jack of the selected channel.

5. Of course, the same event may also be coded by two operators, as may be done in determining coder reliability. The two coders may work either simultaneously or at different times.

6. The reader may wonder why, if it is true that two independently running mechanical systems will not remain in sync over time, it is reasonable to expect the video recorder to remain in synchrony with itself on successive passes. The reason is that the recorder's drive mechanism is controlled by the sync signal recorded on the videotape. This ensures that within quite broad limits there will be no variation from pass to pass due to friction in the drive system, stretching of the tape, and so forth.

7. It is not necessary that the other hardware be connected while this is being done.

8. Requests should be sent to the senior author. The diskette should be packed carefully to protect it from being damaged in the mail.

(Manuscript received July 8, 1987;
revision accepted for publication November 11, 1987.)