Recoding low-level simulator data into a record of meaningful task performance: The integrated task modeling environment (ITME)

ROBERT KING AND SIMON PARKER
Defence Science and Technology Organisation, Fishermans Bend, Victoria, Australia

AND

KON MOUZAKIS, WINSTON FLETCHER, AND PATRICK FITZGERALD
Swinburne University of Technology, Hawthorn, Victoria, Australia

The Integrated Task Modeling Environment (ITME) is a user-friendly software tool that has been developed to automatically recode low-level data into an empirical record of meaningful task performance. The present research investigated and validated the performance of the ITME software package by conducting complex simulation missions and comparing the task analyses produced by ITME with task analyses produced by experienced video analysts. A very high interrater reliability (≥ 0.94) existed between experienced video analysts and the ITME for the task analyses produced for each mission. The mean session time:analysis time ratio was 1:24 using video analysis techniques and 1:5 using the ITME. It was concluded that the ITME produced task analyses that were as reliable as those produced by experienced video analysts, and significantly reduced the time cost associated with these analyses.

Designing, Testing, and Evaluating New Systems

All systems developed for military aircraft undergo a design, testing and evaluation process in order to maximize their effectiveness before introduction into service. Recent times have seen increasing importance placed on human factors engineering (HFE) during the design, test and evaluation process. Systems which meet all "equipment" specifications (e.g., processing speed, memory requirements and signal detection rates) will not be procured if human performance is degraded due to poor HFE design (Booher, 1988).

Designing and evaluating new systems creates a methodological dilemma for HF practitioners. On one hand, the classic laboratory-based experimental testing paradigm allows hypotheses about causal relationships between independent and dependent variable(s) to be tested, and accepted or rejected with an associated level of probability (of being correct). The down side of this approach is that while hypothesized relationships between (some) system variables and (some) human performance measures can be tested, all other variables are usually rigorously controlled in order to reduce variability and increase the likelihood of a statistically significant result. Inevitably, these studies are of limited value because the results are not necessarily predictive of performance in operational circumstances because the design and evaluation process does not include the rich interactions, interdependencies and contextual influences found in the real world.

HF specialists working with new systems agree that, ideally, these systems should be tested and evaluated in their intended operational settings with the measure of greatest interest being "complex, meaningful task performance" (Hennessy, 1990 p. 441). However, while the real world is authentic, it is not easily controlled or measured. The real world violates the assumptions of the classic experimental paradigm with the result that practitioners must examine observational data for correlational rather than causal relationships. While this is not a problem per se, the inability to exert experimental control often means that new systems are not tested and evaluated under a full range of operational scenarios, simply because some scenarios did not occur during the period of the test and evaluation process.

The problem of collecting observational data in the real world is also significant, particularly in the military aviation environment. Ideally, the observational data record should preserve a record of the operator's task performance and any environmental or system events which affect that performance. Any data collection apparatus proposed for use in a military aircraft is itself subjected to an exhaustive design, test and approval process. The apparatus cannot jeopardize the safe operation of the aircraft,

852

must be g-tolerant, crashworthy, not emit any electromagnetic interference and is also subject to weight, space and volume limitations. Often, the time required to design and test data collection apparatus exceeds the time frame allowed for the test and evaluation of the new system.

Measuring and Analyzing Task Performance

During the 1980s, advances in video technology alleviated some of the problems associated with collecting data in operational environments. The miniaturization of video cameras and the development of compact multiplexers, video recorders and time code generators made it easier to fit data collection apparatus to aircraft. It became possible to record and integrate information from different crewmembers and/or different sources (e.g., direct recording of displays to capture system or environmental events) and gather permanent time-coded observational records of operator task performance and environmental events in operational environments.

The existence of a permanent video data record led to the development of methods to measure and analyze task performance. Empirically validated task analysis (EVTA) generates an empirical time-stamped record of task performance from a detailed video analysis of the operator's observable activities and communications (Shaffer, 1989; Shaffer, Hendy, & White, 1988). Observational data which has been "coded" or transformed into an empirical record can then be examined using correlational statistical methods (King, Kieboom, & Manton, 1993) or sequence dependent analytic methods which have been collectively described as *exploratory sequential data analyses* (EDSAs: Fisher & Sanderson, 1993; Sanderson, 1991; Sanderson & Fisher, 1994; Sanderson, Scott, Johnston, Mainzer, Watanabe, & James, 1994). Information regarding task performance may emerge directly from the empirical data record, or by contrasting data collected under different conditions.

The development of objective analysis methods was seen as a significant step forward by HF practitioners, but the EVTA methodology has one major disadvantage, that video analysis is extremely labor and time intensive. The ratio of observational session time (ST) to analysis time (AT) has been quoted as varying between 1:10 and 1:100 (or greater), depending on the experience of the analyst and the granularity of the analysis. Efforts to reduce the ST:AT ratio have seen the development of MacIntosh and PC-based interactive software packages such as MacShapa (Sanderson, James, & Siedler, 1989; Sanderson et al., 1994), CABER (Patrick & McKenna, 1986) and ADIVA (Monterey Technologies, 2002) to aid video analysis. The most recently developed package, ADIVA (All Digital Interactive Video Analysis) allows the user to digitize the video, predefine a task taxonomy and build and manage a task performance database by "pointing and clicking" the mouse. However, improvement in the ST:AT ratio has been incremental and the ST:AT ratio is still very high.

Complex Simulation

In the late 1980s and 1990s, complex simulation was seen as the panacea for all the HF practitioners' design, test and evaluation problems. Advances in processing power, graphics systems, software and database design (see Baskett & Hennessy, 1993; Ferguson, 1990; Strachan, 1995) meant that aircraft and the operational environment could be realistically represented in high-fidelity "full-mission" simulators (Henderson, 1989; Voorhees, Bucher, Gossett, & Haworth, 1989). The advantages of the simulation environment over the real world for designing and testing new systems were immediately apparent because (1) while the testing environment is operational, some control could be exerted through appropriate scenario generation, and (2) all the problems associated with trying to fit data collection apparatus to real aircraft would be overcome, indeed, simulator architectures would make it possible to electronically log hundreds of environmental, system or task performance variables at sampling rates up to 60 Hz, producing gigabytes of observational data per hour.

However, while the simulation environment offers more control than the real world, and does alleviate the problems associated with fitting data collection apparatus to operational aircraft, the type of observational data that can be captured via electronic logging has proven to be too "low-level" (at the level of a buttonpress or switch closure) to be useful in evaluating meaningful task performance. For example, the status of any of the keys around a multifunction display (MFD) in the cockpit can be logged and whether the key was pressed or not pressed determined—but the keypress itself can fulfill different functions depending on what was actually displayed on the MFD at the time of the press. Thus, the knowledge of the keypress is only meaningful in the context of another piece of information—that is, what was displayed on the MFD at the time. As Hennessy (1990) pointed out after preliminary studies in the US Army's Crew Station Research and Development Facility, "a large amount of low-level data from numerous sources is difficult to analyze and interpret if there is no pre-established structure for aggregating the data to yield some higher level meaning" (p. 445).

Given the problems associated with electronically logged low-level data, video recording is still being used to obtain a meaningful time-coded observational data record. Any detailed or objective analysis of task performance still involves video analysis, with its attendant shortcomings.

Recoding Low-Level Data Into a Meaningful Record of Complex Task Performance

Clearly, problems exist with the time it takes to transform observational data into an empirical record using current methods (i.e., coding observational data via video analysis). At this time it seems that the greatest gain to be made in the objective measurement of meaningful task performance will be through the automation of this recoding process.

The viability of automatically recoding electronically logged low-level data has been demonstrated (King & Goss, 1995). This method involved logging time-coded low-level information directly from the 1553 databus in a Sikorsky S-70B-2 Seahawk helicopter during operational test and evaluation of systems in the aircraft. A rule-based program was implemented in PROLOG to parse these data and produce a time-coded record of meaningful task

performance. Observational data were also gathered via video and encoded using the "traditional" EVTA technique during this test and evaluation program. While it took two experienced analysts some three months of full-time work to analyze and code 13 critical mission segments (a total of 78 min of video record) for each of the three crew members in the aircraft, an equivalent coding was obtained in approximately 2 min using the automatic process and a low-end PC. It did take a number of weeks to write and debug the rule set, but this was a one-off cost that still took less time than the manual video analysis and encoding of the data described above.

The software tool described demonstrated the viability of automatically recoding low-level data into an empirical record of task performance. However, it did have two shortcomings. First, the software was developed for a specific system and the specific rule-set could not be used to analyze other systems. Second, the software was not "user-friendly" in that it was written in PROLOG, and would require a person with high-order software coding skills to implement rulesets for other systems.

THE INTEGRATED TASK MODELING ENVIRONMENT

The Integrated Task Modeling Environment (ITME) is a software tool that has been developed by DSTO and the Information Technology Innovation Group at Swinburne University of Technology. ITME is a user-friendly tool that allows an analyst to recode low-level data into an empirical record of meaningful task performance. ITME allows the analyst to easily model the system to be analyzed and construct the recoding rule set and subsequent output for any system using a graphical user interface (see Figure 1). ITME will analyze extremely large low-level data sets in any format.

In ITME, each rule contains a condition and a set of associated actions to be performed if the rule is met. The entire system model and governing rules are created using a high-level graphical user interface and the rule set is saved in Extensible Markup Language (XML) format, while the processing is performed by a rule processing engine written in Delphi 5.0.

ITME processes data from a data file by (1) reading a line of data from the file, (2) placing that data in appropriate data structures, (3) processing the data using the XML rules and engine, and (4) returning to read the next line of data from the data file.

When a line of data is read, the data is copied into an internal data structure, which maintains all the variables that are currently active—for example, the current task being undertaken. The engine will then evaluate all the rules (as set out in the XML structure) and process the actions for each rule whose condition is met. Actions consist
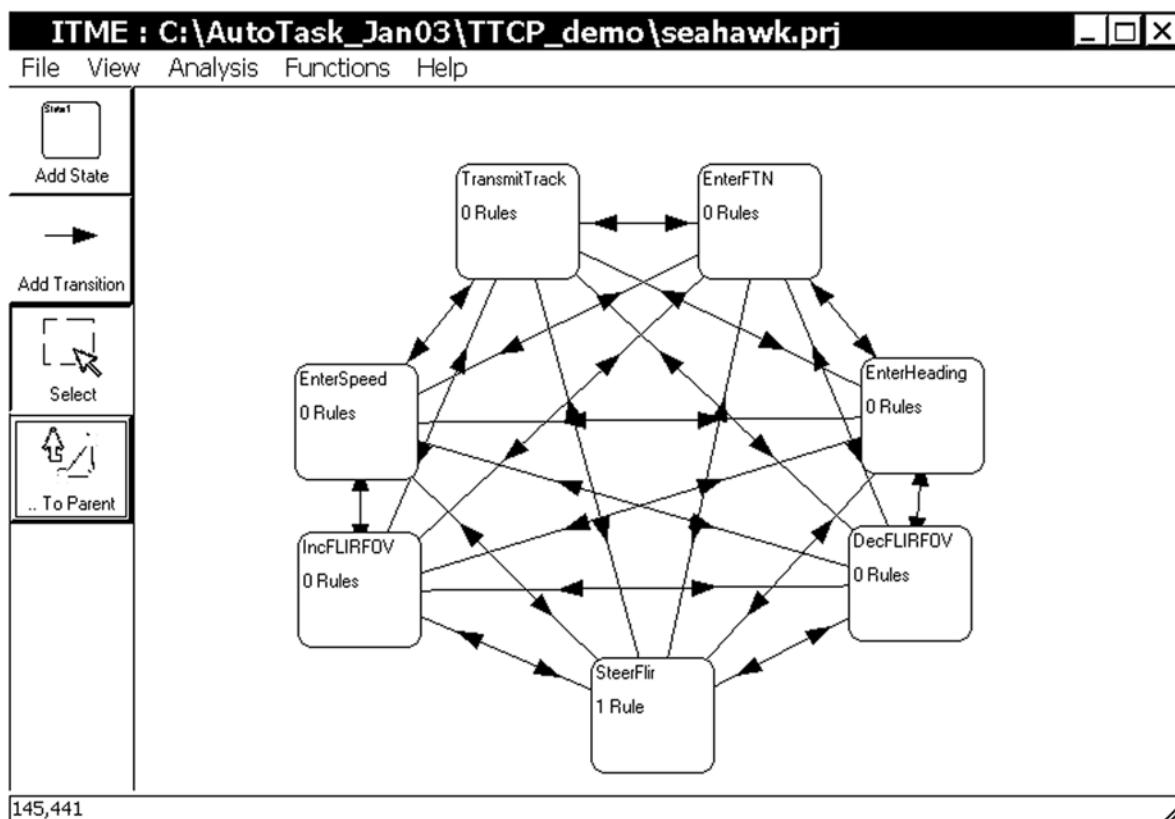

Figure 1. In ITME, rules for defining and transitioning between high-level tasks are easily built by using a graphical user interface. Rules are based on low-level data captured in the data file.

of operations such as setting the current task, setting a variables value, calling a function, writing data to output or incrementing task occurrence counters and times. Actions and outputs can be controlled by conditions, which can all be defined by the analyst.

ITME VALIDATION

In order to investigate and validate ITME's performance, a study was conducted in the Air Operations Simulation Center (AOSC) located at the Defence Science and Technology Organisation's facility in Melbourne, Australia. The aim of the study was to conduct complex simulation experiments and conduct task analyses of aircrew activity using "traditional" video analysis methods and ITME with a view to (1) comparing the task analyses produced by ITME against the task analyses produced by experienced video analysts, and (2) comparing the time it took to complete the task analyses using each method.

METHOD

In the simulation studies, four S-70B-2 Seahawk helicopter crews comprising a Tactical Co-Ordinator (TACCO) and Pilot flew an Anti-Surface Warfare (ASuW) mission segment of around 15 min, at low-level, at night. Each simulation run was initiated with the Seahawk being located 5 nautical miles from a flotilla of vessels, all of which were illuminated on a radar/tactical plot (R/TACP) plan position indicator (PPI) like display. The flotilla generally consisted of a mixture of large fishing vessels, cargo vessels and one patrol boat, fifteen vessels in all. The Seahawk's mission was to track and update track information on all targets and close, classify and identify targets (by vessel's name if possible) using a Forward Looking Infra Red (FLIR) sensor. Pilots flew the Seahawk under difficult conditions (low-level at 100-ft altitude, night). TACCO's were required to (1) navigate and put up fly-to points for the Pilot on the R/TACP display, (2) extract target range and bearing information from the R/TACP display, (3) send track updates on all targets by entering track numbers, courses and speeds of targets via a multifunction keyboard (MFK), and transmitting, and (4) control the FLIR (steer and change field-of-view).

The simulations were terminated after the TACCO had identified two vessels by name.

TACCO Task Taxonomy

For the purposes of the present study, the behavior of interest was the TACCO's performance of meaningful tasks. The TACCO's tasks were categorized according to the taxonomy shown in Table 1.

Video Data Capture

Four video views of TACCO activity were recorded. These separate video feeds were time synchronized and multiplexed so the analysts could see all four views on one screen.

The views comprised of (1) a wide angle shot from behind the TACCO (left-hand seat) and pilot (right-hand

Table 1
Task Taxonomy for TACCO's Tasks

1. Navigate
    1.1 Extract target range/bearing information from R/TACP display
    1.2 Alter radar range
        1.2.1 Decrease radar range
        1.2.2 Increase radar range
        1.2.3 Other
2. Detect/Identify Target
    2.1 Steer FLIR
    2.2 Alter FLIR field-of-view
        2.2.1 Zoom in
        2.2.2 Zoom out
        2.2.3 Other
3. Send Target Track Updates
    3.1 Enter target's force track number
    3.2 Enter target's heading
    3.3 Enter target's speed
    3.4 Transmit track update
    3.5 Other
4. Transmit Radio Message
    4.1 TACCO to pilot
    4.2 Pilot to TACCO
    4.3 TACCO to ship (transmit target identification)
    4.4 Other
5. Other

seat), which allowed the analyst to an overall view of the R/TACP display, and the controls the TACCO was manipulating; (2) a close-up view of the TACCO's right hand, which allowed the analyst to view the TACCO's use of the trackball unit and buttons (put up fly-to points on the R/TACP, extract target range and bearing information and control/steer FLIR) and MFK button use (send target track updates); (3) a direct feed of the R/TACP display, which allowed the analyst to determine whether the TACCO was undertaking the put up fly-to points task, or the extract target range and bearing information task; and (4) a direct feed of the FLIR display, which allowed the analyst to determine whether the TACCO was undertaking the steer FLIR task, or the alter FLIR field-of-view task.

All intercommunication system (ICS) and external radio transmissions were also recorded on the audio track of one of the video feeds, allowing the analyst to categorize all radio transmissions (TACCO to pilot, pilot to TACCO, TACCO to ship) that were undertaken over the course of the mission.

Electronic Data Logging

Data was also logged electronically during the simulation runs. Thirty-two data variables were captured in all. The variables recorded (1) the frame and run time (*frame* and *runtime* variables), (2) the position of the track ball unit (*flir_rad_x* and *flir_rad_y* variables), (3) the operation of any buttons associated with the track ball unit (*flir_az*, *flir_fov*, *rad_range*, *flir_active*, *flir_fov_dec*, *flir_fov_inc*, *flir_rad_reset*, *rad_active*, *rad_range_dec* and *rad_range_inc* variables), (4) the operation of any buttons on the MFK (*gps_line_1*, *gps_line_2*, *gps_line_3*, *gps_line_4*, *gps_button1*, *gps_button2*, *gps_button3*, *gps_button4*, *gps_button5*, *gps_button6*, *gps_button7*, *gps_button8*, *gps_button9*, *gps_button0*, *gps_clear* and *gps_ltr* variables), and (5) the operation of the radio button (*cp_radio* variable).

Table 2
The First Three Frames (or 0.05 Sec) of Electronically Logged Data From One Simulation Segment

DATA VARIABLES STATIC FOR THIS DATA RUN

Tacco_Number   2
Pilot_Number   2
Run_Number   1
Scenario_Number   2

Frame	run_time	rad_alt	flir_rad_x	flir_rad_y	flir_az		
	flir_fov	rad_range	flir_active	flir_fov_dec	flir_az		
	flir_fov_inc	flir_rad_reset	rad_active	rad_range_dec			
	rad_range_inc	gps_line1	gps_line2	gps_line3			
	gps_line4	gps_button1	gps_button2	gps_button3	gps_button4		
	gps_button5	gps_button6	gps_button7	gps_button8	gps_button9		
	gps_button0	gps_clear	gps_ltr	cp_radio			
0	0.0000	3.2808	0.0052	−0.0027	347.7538		
	19.5004	16	0	0	0	1	0	0	0
	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0
1	0.0167	3.2871	0.0052	−0.0027	347.7538		
	19.5004	16	0	0	0	1	0	0	0
	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0
2	0.0333	3.3033	0.0069	−0.0054	347.7538		
	19.5004	16	0	0	0	1	0	0	0
	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0
3	0.0500	3.3262	0.0069	−0.0054	347.7538		
	19.5004	16	0	0	0	1	0	0	0
	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0

The first three frames (or 0.05 sec) of electronically logged data from one simulation segment are shown in Table 2. Fifteen minutes of simulation will typically produce around 54,000 lines of data when produced and logged at 60 Hz (60 frames/sec).

RESULTS

The video data was analyzed by two experienced video analysts. Each analyst independently completed a video analysis of TACCO activity (using the task taxonomy described in Section 3.2) in the four mission segments using the ADIVA video analysis software package.

The electronically logged data was analyzed using the ITME software package. One of the analysts above constructed the rule set to define the tasks outlined in the taxonomy described in Table 1.

Reliability of Task Analyses

Pearson's product-moment correlations (Howell, 1997), based on the frequency (number of occurrences) of each task were calculated for the task analyses produced by Analyst 1, Analyst 2, and ITME for the mission segment flown by each crew. Pearson's r indicates how strong the rank ordering of tasks (based on frequency) is across the analysts and ITME. Pearson's r for each mission segment is shown in Table 3.

Table 3 shows a high correlation ($\geq$.94) existed between Analyst 1, Analyst 2, and ITME for the task analyses produced by each for each mission segment. The correlation coefficient was $r =$ .94 in Mission Segment 1 for ITME and Analyst 1, $r =$ .99 for ITME and Analyst 2, and $r =$ .94 for Analyst 1 and Analyst 2 (see Table 3). In Mission Seg-

ment 2, $r =$ .97 for ITME and Analyst 1, $r =$ .98 for ITME and Analyst 2, and $r =$ .99 Analyst 1 and Analyst 2 (see Table 3). In Segment 3, $r =$ .99 for ITME and Analyst 1, $r =$ .97 for ITME and Analyst 2, and $r =$ .95 for Analyst 1 and Analyst 2 (see Table 3). In Segment 4, $r =$ .97 for ITME and Analyst 1, $r =$ .99 for ITME and Analyst 2, and $r =$ .96 for Analyst 1 and Analyst 2 (see Table 3). All correlations were statistically significant at the $p <$ .05 level.

Intraclass correlations (Shrout & Fleiss, 1979) were also calculated for each mission segment. Intraclass correlation is sensitive to the magnitude (frequency of occur-

Table 3
Pearson's r and Intraclass Correlation Coefficients for the Task Analyses Produced by Analyst 1, Analyst 2, and ITME for the Mission Segment Flown by Each Crew

Mission Segment		Analyst 1	Analyst 2	ITME
1	Analyst 1	1		
	Analyst 2	.94*	1	
	ITME	.94*	.99*	1
Intraclass correlation = 0.94				
2	Analyst 1	1		
	Analyst 2	.99*	1	
	ITME	.97*	.98*	1
Intraclass correlation = 0.98				
3	Analyst 1	1		
	Analyst 2	.95*	1	
	ITME	.99*	.97*	1
Intraclass correlation = 0.97				
4	Analyst 1	1		
	Analyst 2	.96*	1	
	ITME	.97*	.99*	1
Intraclass correlation = 0.97				

*$p <$ .05.

rence) of each task and assesses the interrater reliability between Analyst 1, Analyst 2, and ITME. The reliability coefficients for each analyst and ITME for the task analyses produced by each for the four mission segments are also shown in Table 3.

Table 3 shows a very high interrater reliability ($\geq$.94) existed between Analyst 1, Analyst 2, and ITME for the task analyses produced by each for each mission segment. The interrater reliability coefficient was $r = .94$ in Mission Segment 1, $r = .98$ for Mission Segment 2, $r = .97$ for Mission Segment 3, $r = .97$ and $r = .97$ for Segment 4.

No statistically significant differences occurred between raters [in Mission Segment 1, $F(2,34) = 3.1073$; in Mission Segment 2, $F(2,34) = 0.9372$; in Mission Segment 3, $F(2,34) = 0.2012$; and in Mission Segment 4, $F(2,34) = 1.8706$].

However, there was a statistically significant "between-cases" effect in every mission segment indicating that there were differences in the frequency of occurrence of each task (although all analysts and ITME concurred on those frequencies). [In Mission Segment 1, $F(17,34) = 48.5756$, $p < .01$; in Mission Segment 2, $F(17, 34) = 140.1524$, $p < .01$; in Mission Segment 3, $F(2,34) = 101.5700$, $p < .01$; and in Mission Segment 4, $F(2,34) = 111.8249$, $p < .01$.]

Time to Conduct Empirically Validated Task Analyses

Table 4 shows the duration of each mission segment (in minutes), the time each analyst took to produce a task analysis using the video analysis technique, the time it took Analyst 2 to create the ruleset and produce a task analysis using ITME, and the respective analysis time to session time (ST:AT) ratios. As Table 4 shows, mission segment duration varied from 8.6 to 14.56 min, with the mean segment duration being 12.03 min.

Video analyses. Time to conduct the video analyses varied from 180 to 660 min, with the mean video analysis time being 285 min. The ST:AT ratio varied from 1:16 to 1:54 with the mean ST:AT ratio being 1:24, reflecting every hour of session time would require 24 h of analysis time.

ITME analyses. As Table 4 shows, it took 210 min for Analyst 2 to construct the analysis ruleset in ITME, and a further 2 min to analyze the logged data file from each

mission segment. The ST:AT ratio was 1:5, reflecting that every hour of session time required 5 h of analysis time.

DISCUSSION AND CONCLUSION

Human factors specialists generally agree that new military systems should be designed, tested and evaluated in the context of their operational settings with the measure of greatest interest being complex, meaningful task performance.

ITME is a software tool that has been developed to automatically recode electronically logged low-level data into an empirical record of meaningful task performance. ITME allows the analyst to construct the analysis rule set and subsequent output for any system via a graphical user interface and will analyze extremely large low-level data sets in any format.

The aim of the present research was to investigate and validate the performance of ITME by comparing task analyses produced by ITME with task analyses produced by experienced video analysts.

Present results show that the ITME software package successfully recoded electronically logged low-level data into an empirical record of meaningful task performance. Task analyses produced by ITME on data files produced in four simulator based missions were as reliable as those produced by two experienced analysts using traditional video analysis techniques, reflected in Pearson correlation scores and interrater reliability coefficients equal to or greater than .94.

ITME also significantly reduced the analysis time for the data recorded in the present experiment. Traditional video analysis of the video record produced a mean ST:AT ratio of 1:24, meaning every 1 h of session time required 24 h of analysis time. With ITME, the ST:AT ratio was 1:5, with every hour of session time requiring 5 h of setup and analysis time. In this instance, ITME was some 480% more time efficient than the traditional video analysis technique. It should also be noted that constructing an analysis ruleset in ITME is a one-off cost and more session time could be analyzed at very little extra cost (2 min or so for every logged data file). With ITME, the ST:AT ratio would further reduce and the efficiency of ITME would increase as larger amounts of data were analyzed, whereas the ST:AT ratio is a fixed overhead for video analysis.

Table 4
Mission Segment Duration, Video Analysis Time, ITME Analysis Time,
and Respective Session Time:Analysis Time (ST:AT) Ratios for the Four Mission Segments

Mission Segment	Segment Duration (min)	Video Analysis				ITME Analysis			
		Time (min)		ST:AT Ratio		Time (min)		ST:AT Ratio	
		Analyst 1	Analyst 2	Analyst 1	Analyst 2	Analyst 1	Analyst 2	Analyst 1	Analyst 2
							210 (= time to construct rule set)		
1	12.15	660	270	1:54	1:22		2		
2	8.60	210	210	1:24	1:24		2		
3	12.82	180	240	1:14	1:19		2		
4	14.56	270	240	1:19	1:16		2		
Total	48.13	2,280					218		
Mean	12.03	285		1:24			54.5		1:5

The ITME software is of potential interest to anyone designing, testing and evaluating new systems in environments (actual or simulated) where low-level data can be electronically logged.

AUTHOR NOTE

Correspondence concerning this article should be addressed to R. King, Air Operations Division, Defence Science and Technology Organisation, 506 Lorimer St., Fishermans Bend, Victoria, Australia 3207 (e-mail: robert.king@dsto.defence.gov.au).

REFERENCES

Baskett, F., & Hennessy, J. L. (1993). Microprocessors: From desktops to supercomputers, *Science*, **261**, 864-871.

Booher, H. R. (1988). Progress of MANPRINT—The Army's human factors program. *Human Factors Society Bulletin*, **31**, 1-3.

Ferguson, R. L. (1990). High performance computer image generation: A marriage of computer graphics and image processing. *Optical Engineering*, **1289**, 140-154.

Fisher, C., & Sanderson, P. M. (1993). Exploratory sequential data analysis: Traditions, techniques and tools. Report of the CHI '92 workshop. *SIGCHI Bulletin*, **25**, 31-40.

Henderson, B. W. (1989). Simulators play a key role in LHX contractor selection. *Aviation Week & Space Technology*, **131** (22), 34-37.

Hennessy, R. T. (1990). Practical Human Performance Testing and Evaluation. In H. R. Booher (Ed.), *MANPRINT: An approach to systems integration* (pp. 433-470). New York: Reinhold.

Howell, D. C. (1997). *Statistical methods for psychology* (4th Ed.). Belmont, CA: Duxbury.

King, R. B., & Goss, S. (1995). Automating task analysis. *Australian Journal of Psychology*, **47**, 14.

King, R. B., Kieboom, H. M., & Manton, J. G. (1993). *Human factors aspects of S-70B-2 and FFG operations* (Report No. ARL-RR-16). Melbourne, Australia: Aeronautical Research Laboratory.

Monterey Technologies (2002). All Digital Interactive Video Analysis user manual (Version 3.50). Monterey, CA: Monterey Technologies.

Patrick, J. D., & McKenna, M. J. (1986). A generalised computer system for sports analysis. *Australian Journal of Science & Medicine in Sport*, **18** (3).

Sanderson, P. M. (1991). *ESDA: Exploratory sequential data analysis* (Report EPRL-94-01). Urbana: University of Illinois at Urbana-Champagne, Department of Mechanical and Industrial Engineering.

Sanderson, P. M., & Fisher, C. (1994). Exploratory sequential data analysis: Foundations. *Human–Computer Interaction*, **9**, 251-317.

Sanderson, P. M., James, J. M., & Seidler, K. S. (1989). SHAPA: An Interactive Software Environment for Protocol Analysis. *Ergonomics*, **32**, 1271-1302.

Sanderson, P. M., Scott, J. J. P., Johnston, T., Mainzer, J., Watanabe, L. M., & James, J. M. (1994). MacSHAPA and the Enterprise of Exploratory Sequential Data Analysis (ESDA). *International Journal of Human–Computer Studies*, **41**, 633-681.

Shaffer, M. [T.] (1989). Using video for empirically validated task analysis (EVTA) of system–human interaction and performance. *MANPRINT Bulletin*, **4**, 12-13.

Shaffer, M. T., Hendy, K. C., & White, L. R. (1988). An empirically validated task analysis (EVTA) of low-level army helicopter operations. *Proceedings of the 32nd Annual Meeting of the Human Factors Society*, **1**, 178-182.

Shrout, P. E., & Fleiss, J. L. (1979). Intraclass correlations: Uses in assessing rater reliability. *Psychological Bulletin*, **2**, 420-428.

Strachan, I. W. (Ed.) (1995). *Jane's simulation and training systems 1995–1996* (8th Ed.). Coulson, U.K.: International Thomson.

Voorhees, J. W., Bucher, N. M., Gossett, T., & Haworth, L. A. (1989). The Crew Station Research and Development Facility. *Army Research, Development, & Acquisition Bulletin*, **2**, 22-25.