The Impact of Increasing Autonomy on Training Requirements in a UAV Supervisory Control Task

Mary Cummings[1], Lixiao Huang[2], Haibei Zhu[1], Daniel Finkelstein[3], Ran Wei[4]

[1] Duke University, Durham, NC, USA

[2]Arizona State University, Mesa, AZ, USA

[3]Georgia Institute of Technology, Atlanta, GA, USA

[4]Texas A&M University, College Station, TX, USA

Corresponding author: Mary Cummings, 304 Research Drive, Box 103957, Durham, NC 27708; email: m.cummings@duke.edu

ABSTRACT

A common assumption across many industries is that inserting advanced autonomy can often replace humans for low-level tasks, with cost reduction benefits. However, humans are often only partially replaced and moved into a supervisory capacity with reduced training. It is not clear how this shift from human to automation control and subsequent training reduction influences human performance, errors, and a tendency towards automation bias. To this end, a study was conducted to determine whether adding autonomy and skipping skill-based training could influence performance in a supervisory control task. In the human-in-the-loop experiment, operators performed unmanned aerial vehicle (UAV) search tasks with varying degrees of autonomy and training. At the lowest level of autonomy, operators searched images and at the highest level, an automated target recognition algorithm presented its best estimate of a possible target, occasionally incorrectly. Results were mixed, with search time not affected by skill-based training. However, novices with skill-based training and automated target search misclassified more targets, suggesting a propensity towards automation bias. More experienced operators had significantly fewer misclassifications when the autonomy erred. A descriptive machine learning model in the form of a Hidden Markov Model also provided new insights for improved training protocols and interventional technologies.

INTRODUCTION

Human supervisory control is a common control scheme for unmanned aerial vehicles (UAVs), during which human operators supervise high-level tasks while autonomous systems are responsible for local control of the UAVs. One common assumption is that advanced autonomy requires less training to master associated tasks, as functionalities are shifted from human to automation control. Indeed, one of the most popular selling points for increasing autonomy across a number of commercial industries is that such systems require less training since autonomy is doing more of the low-level control work.

The military is increasingly attempting to reduce training time and costs, and increased autonomy is one possible way to do this. Because of the advanced automation onboard UAVs, particularly their automatic landing and takeoff abilities, the Air Force has ushered in a new program that allows UAV operators to skip flight training and enter UAV training with significantly less experience than their counterparts that attend flight school. Because of the onboard automation, Air Force UAV training has been reduced from 2 years to 9 months for these new non-pilot operators (Blacke, 2009).

While this reduction in training time is important for minimizing costs and potentially getting personnel to theaters of conflict faster, it is not clear how this shift could influence human performance and increased error rates. More specifically, if aspects of training are deleted due to increased vehicle autonomy, like the elimination of learning to take off and land a plane because these functions are automated in a UAV, is there a potential cost in terms of UAV operator performance? If human training is dropped for low-level tasks that are taken over by autonomy, is there a link to inappropriate trust which could lead to complacency and associated errors? An operator's trust in an autonomous system influences reliance on that system (Dzindolet, Peterson,

Pomranky, Pierce, & Beck, 2003; Moray, Inagaki, & Itoh, 2000), and so it is critical that the influence of increasing autonomy on inappropriate trust is also examined (Lee & See, 2004; Mittu, Sofge, Wagner, & Lawless, 2016).

Deskilling has long been suspected to be a negative outcome of increased automation and subsequent reduced training (Bainbridge, 1983; Strauch, 2018). More recent research has demonstrated that there is complex interplay between inserting automation and different training tasks, exacerbated by operators' aptitudes (Clegg, Heggestad, & Blalock, 2010), and the use of automation to augment part-task training has been called into question (Gutzwiller, Clegg, & Blitch, 2013). In general, it is unknown how training should be redesigned to successfully accommodate new task requirements in the presence of increased autonomy such that task performance and situation awareness are not decreased.

This problem of determining training requirements in the presence of advanced autonomy is related to the function allocation conundrum, i.e., what functions should be exclusively given to autonomy versus exclusively humans, or possibly shared between the two. The SRKE (skill, rule, knowledge, expertise) model provides a framework to shed light on this problem (Cummings, 2014; Cummings, 2018) and is illustrated in Figure 1.

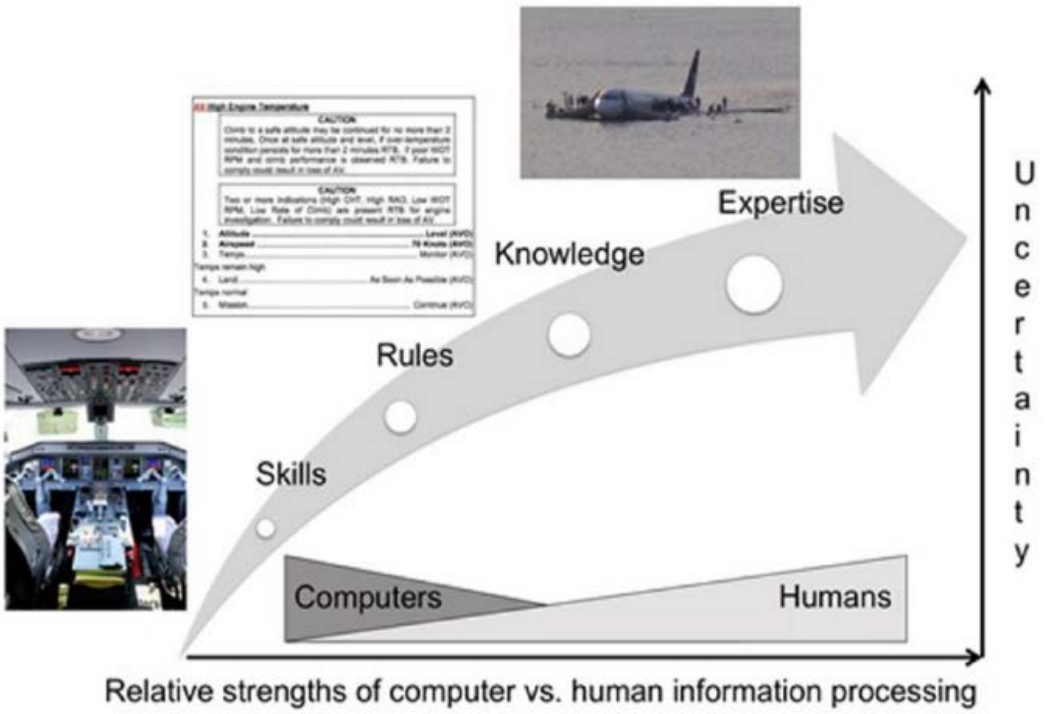

***Figure 1*. Role allocation for increasingly complex behaviors and the relationship to uncertainty (Cummings, 2014)**

In the SRKE model in Figure 1, skill-based behaviors are highly practiced and automatic responses to stimuli, such as a pilot's ability to keep the plane in stable flight. Once these are mastered, this frees the operator to concentrate on attending to rule-based behaviors that have clear action criteria associated with signals from the world. For example, pilots use procedures as instructions in response to emerging situations, such as the steps to follow when a cabin depressurization alert occurs. Once skills and rules are mastered, operators are then able to turn more cognitive resources to knowledge-based behaviors, which occur in novel situations where conceptual understanding of the environment is required. Expertise-based responses occur under the highest degrees of uncertainty, such as when a plane loses both engines and the pilot must determine how to safely land the airplane. Typically, only highly experienced people can become true experts since it takes time to be presented with such uncertain scenarios over the course of a

career. As depicted in Figure 1, computers currently can automate many skill- and rule-based behaviors in settings with adequate sensors, such as autopilot and path planning, but knowledge- and expert-based behaviors require judgments under uncertainty that are outside the scope of current systems enabled with any form of artificial intelligence.

The SRKE framework falls in line with other research calling for a capabilities-based approach to function allocation, as compared to a levels-of-automation approach (see (Kaber, 2018) for debates on this point). For example, Feigh and Pritchett (2014) lay out requirements for such an approach including "Each agent must be Allocated Functions That It Is Capable of Performing," and "The Function Allocation Must Support the Dynamics of the Work," which are similar to the discussions about humans and computer capabilities in the SRKE model. Other researchers have used the skill, rule, and knowledge-based behavior approach to determine function allocation in the design of cockpit automation (Idris, Enea, & Lewis, 2016), as well as for intelligent driving applications (Wang, Hou, Tan, & Bubb, 2010).

What is unclear in any SRKE paradigm is how much one level of successful reasoning relies upon adequately learning those behaviors from a lower level, especially in human supervisory control. For example, how much do takeoff and landing skills learned by Air Force pilots who fly UAVs actually benefit them when the UAV experiences a problem while landing? It is widely recognized that deskilling is a significant concern for operators of semi-autonomous systems where they still have to take over manual control at times (Ferris, Sarter, & Wickens, 2010; Parasuraman, Sheridan, & Wickens, 2000; Wickens & Hollands, 2000). However, what is less known is how removing the learning of these skills due to increasing autonomy potentially influences possible performance degradation and potential errors in judgment. It has been well-established that an increase in automation in supervisory control settings can lead to automation

bias, which is the propensity to over trust automated recommendations and not seek any disconfirming evidence (Cummings, 2004; Mosier, Skitka, Heers, & Burdick, 1998). It is possible that the insertion of automation and the reduction of training could lead to more cases of automation bias.

In addition to the influence of increasing UAV autonomy on training objectives, the role of video game experience in these settings is also important to consider as it has been shown to lead to better learning (Schenk, Lech, & Suchan, 2017) and improved performance in such settings (Lin, Wohleber, Matthews, & Funke, 2015). In addition, trust is also a significant consideration as previous research has shown that people with a lack of experience can overly trust such supervisory control systems (Cummings, Bertucelli, Macbeth, & Surana, 2014). However, distrust can also be a factor in such settings (Parasuraman, Sheridan, & Wickens, 2008), even though the automation performs satisfactorily (Cummings, Buchin, Carrigan, & Donmez, 2010). Thus, a study of operator training and autonomy should consider these influences as well.

To this end, we wanted to explore in a simplified UAV control environment, whether the presence or absence of a trained skill could dramatically influence performance under two different levels of autonomy and reliability. We wanted to determine whether there were consequences of bypassing skill-based training in a UAV supervisory control task, and having people start with just rule-based training. To this end, we developed a test environment that allowed for training of operators with and without automated target search. By manipulating the type of training participants received in the presence (or absence) of advanced autonomy, we aimed to examine the criticality of skill-based training that focused on manipulating an unfamiliar input device, particularly in the presence of advanced autonomy. In addition, because

experience can influence not only trust in automated systems, but also the ability to deal with uncertainty in systems, we wanted to examine how novices differed from experts in this regard. Our hypothesis was that skipping skill-based training, as represented in the SRKE model in Figure 1, would have a negative effect on performance in a human supervisory control task of supervising multiple UAVs, which would be more pronounced in novices than experts.

METHOD

Experiment Testbed

An experiment was designed and conducted utilizing a modified version of the Research Environment for Supervisory Control of Heterogeneous Unmanned Vehicles (RESCHU) experiment platform (Nehme, 2009). RESCHU is a Java-based discrete event simulation platform, which provides for testing of single operator supervision of multi-UAVs in multi-tasking supervisory control scenarios that include navigational and imagery searching tasks. The interface of the RESCHU platform is shown in Figure 2. The interface features four main components: the payload camera view, control panel, mission timeline and map area.

The primary purpose of the camera view displays is to conduct imagery searching tasks through the camera once a UAV reaches a target. The control panel provides the UAV damage level, which is caused by UAVs intersecting with the yellow hazard areas. A message box provides directives from a virtual supervisor for searching tasks (i.e., a supervisor would tell the operator that when he or she reaches target A, search for a red truck in the parking lot of the chemical plant). The timeline in Figure 2 shows the estimated remaining time of all UAV arrivals at waypoints and assigned targets. The map displays the area of surveillance with real-time locations of all UAVs, hazard areas and targets.

One significant addition to this version of RESCHU was the insertion of an automated target recognition (ATR) system to assist in imagery search tasks. Such a system represents advanced autonomy that is meant to both reduce operator workload as well as training time. In theory, ATR systems use computer vision and machine learning techniques to automatically identify a target of interest, thus reducing the search time of a human operator. However, in practice such systems have high false alarm rates and are often problematic in actual operations (Ratches, 2011), which can later lead to issues with trust.

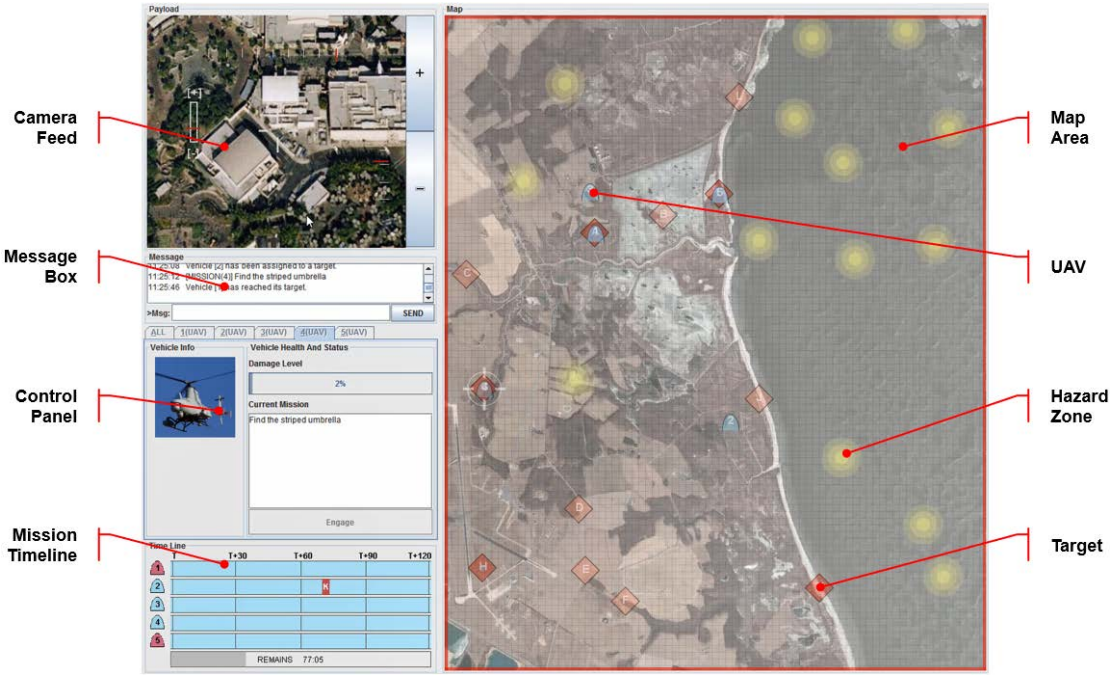

Figure 2. **RESCHU experiment platform operator interface.**

In RESCHU, participants with ATR assistance see the computer's proposed target in the camera video feed window without any need to search for the target through panning and zooming. However, the ATR system was only 70% correct in its identification to simulate real-world problems with reliability and also since previous research has indicated this is a critical threshold for human trust (Wickens & Dixon, 2007). For participants without ATR assistance, they had to search for the target through panning and zooming interactions, which were also

available to those with ATR if they decided they did not agree with the automated recommendation, which is similar to actual systems.

In order to test the interactions between increasing autonomy and skill-based training, we needed a scenario where skills would be required under the lower autonomy case of no ATR, which would not be needed in scenarios where operators had access to ATR. To this end, we elected to use a new, unfamiliar input device that required dedicated training to master, especially in the search task, but could still be used with some on-the-job training.

Thus, the Kensington Expert Trackball Mouse® was used as the new unfamiliar input device, which required operators to learn a new manual control skill vis-a-vis trackball manipulation. This device is much faster than a traditional mouse, and participants had to both adjust to the speed of movement and relearn which buttons mapped to various functions. Such an equipment change is reflective of real-world decisions to add hardware to a new system that requires a skill set which takes time to learn, but is often introduced with the expectation that operators will learn it on their own.

Participants who received trackball training were given ~20 minutes of dedicated training in an abstract training task. None had ever used such a device prior to the experiment. An abstract training task was needed to allow participants to become experienced in using the trackball without giving them significant experience inside the RESCHU environment. Thus, we developed a Fitts's Law training environment (FLTE) (Figure 3), which embodies the well-known Fitts's Law relationship that the movement time (MT) required to rapidly move to a target area is a function of the distance and the size of the target (Fitts, 1954; MacKenzie, 1995).

$$Movement\ Time = a + b*ID \quad ID = \log_2 (D / W_e + 1)$$ (Equations 1 & 2)

In these equations, ID is the index of difficulty, D is the distance to the target for selection and W_e is the effective width of the target. Coefficients a and b are the slope and intercept coefficients, determined through empirical tests conducted in a pilot study. In the training environment in Figure 3, participants conducted 6 blocks of 75 clicks each in FLTE with 30s breaks between blocks. This protocol has been shown to effectively train people learning a new mouse input device to relatively stable levels of performance (MacKenzie, Kauppinen, & Silfverberg, 2001).

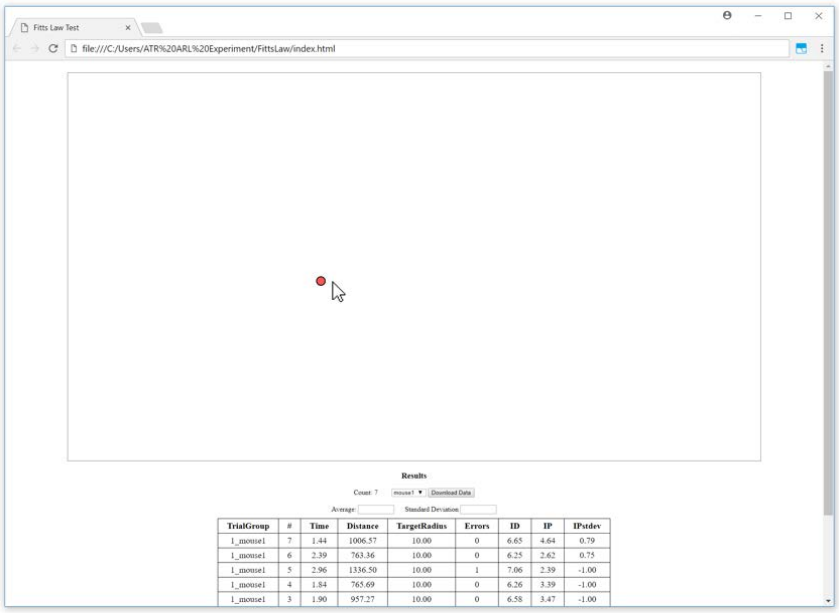

Figure 3. **Fitts's Law Training Environment, the 10 pixel target dot moves randomly across the screen.**

Experiment Objectives and Participants

The primary objectives for participants in this study were to direct multiple UAVs to: (a) search images for specific objects when UAVs reached their assigned targets as directed in the message box (e.g., a red car in the parking lot), (b) ensure UAVs do not encounter hazard zones marked in yellow in Figure 2, and (c) optimize the routing as needed. UAVs were automatically

assigned to a new target once a target search concluded but the assignment was often intentionally sub-optimal, with the underlying algorithm randomly assigning UAVs to targets, so participants could improve the search process.

Participants included one group of 15 students from a US southeastern university (9 male, 6 female, age *mean* = 24.1 years, *SD* = 4.0 years) and another group of 15 FAA Part 107 commercial UAV pilots (13 male, 2 female, age *mean* = 35.1 years, *SD* = 10.8 years). The participants were all over the age of 18, with 20/20 or corrected to normal vision, no neurological disorders, or any physical impairments that would prevent them from using conventional computer input devices. The study recruited participants from campus mailing lists and flyers with a $25 gift card for the 1.25-hour experiment and an additional $100 gift card for the best performer.

Among the participants, thirteen had little video game experience, eight participants had monthly gaming experience, four participants played video game several times a week, another three participants had weekly gaming experience, and only two participants had daily gaming experience, ranked on a Likert scale from 1-5. The Part 107 commercial UAV pilots in this study were very familiar with looking through actual UAV cameras but had no experience supervising multiple UAVs.

Experiment Procedures

The experiment procedure consisted of three phases. The first phase provided participants with different types of training, detailed in the next section. The second phase was a 15-minute practice session to allow participants to get familiar with the experiment interface, including the steps needed to engage the camera, the procedures needed to successfully search and identify

each target assigned, and then how to submit a final answer. This training also included the other elements of RESCHU including directing UAVs out of hostile areas and how to improve their path planning. At the end of this phase, there was a dedicated test session during which an evaluator ensured that each participant could successfully identify and search for a target without any assistance and could successfully navigate the UAVs between targets with no exploration for the correct interface elements. The third phase was data collection with two different 20-minute test scenarios per person with no change in experimental conditions.

Participants were split into three training groups for the first phase. In the first training group, participants were provided with the additional trackball training described previously. The FLTE practice was considered as skill-based training because it was designed to decrease participants' input movement speeds while searching in the camera window and thus reducing their overall time in the search task, which would allow them to more rapidly switch between targets.

In terms of the level of autonomy, this first group labeled "*skill-based with ATR*" was given the ATR so they, on average, should have the fastest target identification times. However, given that their ATR was 70% reliable, which they were told, it was possible that they could become complacent and not recognize the automation provided a poor recommendation. Given that each person was presented with ten targets, in the case of the unreliable ATR, three of the ten were not correct identifications but the correct target was somewhere in the image. Thus, the presence of ATR could lead to incorrect identifications. If participants did catch the anomaly, given that this group had significant training in using the trackball, we expected that their search task times would be somewhat slower than when the ATR was correct, but still fairly quick.

The second group also had the same training in terms of the interface and the trackball but were not given the ATR assistance, so it was expected that this group would have longer times for target identification than those with ATR. Participants with no ATR would have to pan and zoom to find the target much more than people who had ATR. We also expected this group to have the fewest number of incorrect identifications, as the chance for automation bias was removed. This group was labeled "*skill-based without ATR*".

The last training group also had the ATR assistance but had minimal training and was labeled the "*ATR without skill-based training*" group. This group received no special trackball training but spent 15 minutes in the practice session acclimating to the new device. Because they did not have specific skill-based trackball training, we expected their search times to be the longest of all three groups when the ATR provided a poor recommendation, since they would be inexperienced at controlling the mouse while performing their two missions. However, their search times when the ATR was correct should be the same as those with skill-based training and ATR. Because of the lack of foundational training in manipulating the trackball, we expected this group to exhibit more instances of automation bias.

The two testing populations, undergraduate and graduate university students versus people with formal UAV commercial licenses, represent novices vs. experienced operators. Given their commercial certification, the experienced operators, in theory, possessed significantly more knowledge about actual UAV operations than people who had never before operated a UAV. Given their time looking through UAV cameras, we expected the experienced operators to more easily identify targets but also detect when the automation was not correctly performing. In order to take this experiment to the commercial UAV operators, who were geographically dispersed, a mobile command center van was used (Figure 4). This van is

equipped with a test station and wireless communication capabilities in which the training and testing took place. All experiments took place in this van for both novices and experienced operators.

a. Mobile command center exterior

b. Mobile command center interior

Figure 4. **Mobile van exterior and interior**

In summary, these different experimental factors resulted in a 3 (skill-based training with ATR, skill-based training without ATR, ATR without skill-based training) x 2 (15 UAV operators, 15 novices) mixed-subject design with 2 test sessions per person that each lasted 20 minutes (Table 1). ATR assignment and experience were between factors with a repeated measure on each test session. Two sessions were included to increase statistical power. The training lasted ~45 minutes for the two skill-based groups but only lasted ~30 minutes for the group that did not receive the skill-based training. Every test group supervised four UAVs, and were presented with ten possible targets, spaced approximately 90s apart. All participants experienced all 10 targets and no vehicles were destroyed in the hazard zones.

Table 1

Experiment Design

Conditions	Experience	
	Novices	Operators
Skill-based w/ ATR	5	5
Skill-based w/o ATR	5	5
ATR w/o skill-based	5	5

RESULTS

A multivariate repeated-measure 3x2 ANOVA model with a repeat on the test session and a significance level of 0.05 was used to analyze experiment data. Video game experience was used as a covariate, but age was not since it was not correlated with any of the performance metrics. There were two mission performance metrics, 1) overall success rates of finding targets and 2) the amount of time UAVs spent in hazard zones. The overall success rate was defined as the percentage of targets correctly identified in a test session. The amount of time in a hazard zone was the average amount UAVs spent transiting a hazard area. There were two other search-related metrics, which were time spent searching for each of the ten targets in the camera window and the average time other UAVs waited at their targets as they needed the operator's attention to search their target window.

When looking at performance across the three experimental conditions (*skill-based with ATR* (1), *skill-based without ATR* (2), *ATR without skill-based training* (3)), there were no statistical differences in overall success rates of finding targets and the amount of time UAVs spent in hazard zones, which were the two primary performance metrics. The video game covariate was also not significant. While not statistically different, participants in the *ATR without skill-based training* condition had the overall lowest error rates of 5%, while those in the *skill-based without ATR* condition were at 9% and *skill-based with ATR* participants were at 11%. As expected, the

commercial Part 107 operators had a third fewer misclassifications than the students (21 vs. 30). When considering all the trials, this meant that the experienced operators had an error rate of 7%, whereas the novices were at 10%. The highest number of misclassifications occurred in the *skill-based with ATR* condition with student novices as seen in Figure 5.

In terms of time spent searching for targets in the camera window, different combinations of training and autonomy had a significant effect on the average time expended in imagery searching tasks ($F(2, 23) = 10.901$, $p < .001$, $\eta^2 = .320$) and the average UAV waiting time ($F(2, 23) = 7.539$, $p = .003$, $\eta^2 = .338$). This waiting time indicated how long UAVs were waiting at other targets for the operator to finish. Thus, it is a measure of operator efficiency in managing his or her attention. Table 2 details the means and standard deviations of these metrics for each group, including the different search times for the reliable and unreliable ATR events. While the video game covariate was not significant for wait time, it was for search time ($p = .043$), meaning the more experience an operator had, the less time was spent on the search task. Overall, for both metrics people with the ATR were faster in searching and left other UAVs waiting less time than those without the ATR.

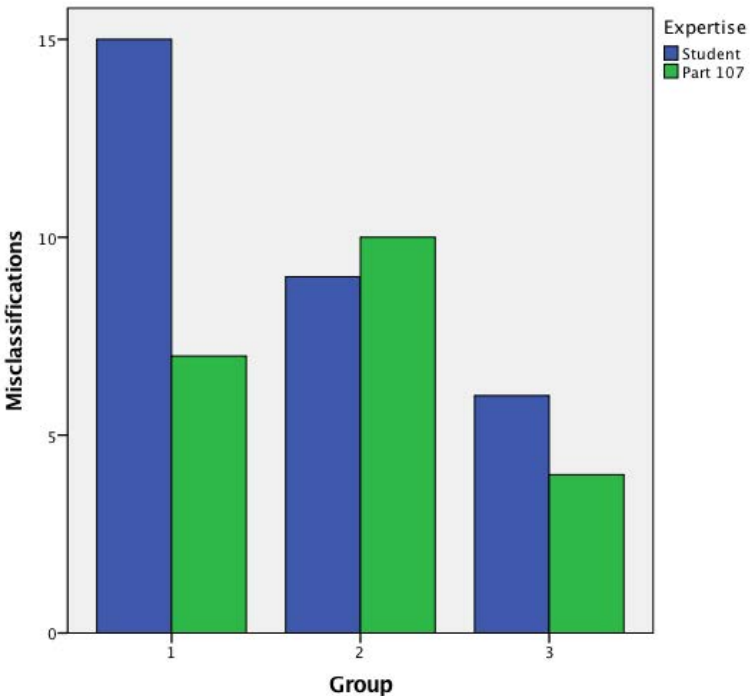

Figure 5. **Number of misclassifications for students and Part 107 certified pilots, as a function of their training group: 1. Skill-based with ATR, 2. Skill-based without ATR, 3. ATR without skill-based training**

When looking at Tukey pairwise comparisons of the average search time, the *skill-based without ATR* group was statistically significantly slower than the other two groups ($p_{12} = .027$, $p_{23} < .001$), which were statistically not different. Thus, on average and as expected, people in the *skill-based without ATR* group took approximately 10s or longer to search for a target than the other groups with ATR. This relationship also held true for the average task waiting time ($p_{12} = .006$, $p_{23} = .015$), with the lack of ATR causing a 15-20s wait time, on average, for images to be searched. There was one single significant interaction ($F(2,23) = 3.37$, $p = .040$, Figure 6) in that the commercial pilots' actions led to the longest wait times, on average 23s longer, but only for those commercial operators without ATR. Thus, the Part 107 experienced operators took much longer to identify a target when they had no ATR, even with significant trackball training. However, as mentioned previously, this same group also had the fewest misclassifications.

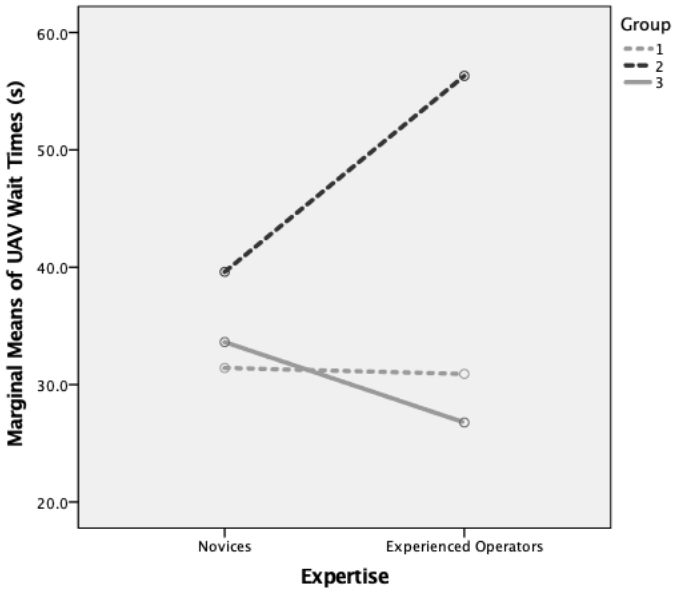

Figure 6. **The significant interaction (F(2,23)=3.37, p=.040) between expertise and different training groups, where 1 = skill-based with ATR, 2 = skill-based without ATR, and 3 = ATR without skill-based training.**

It is also interesting to note the difference between those people in the *skill-based with ATR* condition when it failed (30% of the time) and those in Condition 2 who never had ATR, especially since they had the exact same training. This large difference could be attributed to a fatigue effect since participants without any ATR assistance had to manually search each image, whereas those with ATR had less work to do for 70% of the time. There was no statistical difference in ages across the groups, so age was not an influencing factor.

Table 2

Efficiency metrics of average image search time and average UAV waiting time

Conditions	Average search time, s (*M/SD)*		Average UAV waiting time, s (*M/SD*) novice vs. experienced operators	
	reliable ATR	ATR not working		
Skill-based w/ ATR (1)	14.1/11.9	21.1/15.8	31.1/8.9	30.4/11.2
Skill-based w/o ATR (2)		33.6/4.7	39.1/3.3	62.1/10.0
ATR w/o skill-based (3)	12.9/10.9	19.2/13.1	42.4/11.1	31.1/13.2

When examining the raw counts of correct versus incorrect classification for each image searched, for participants with ATR (Conditions 1 and 3), when the ATR was correct, there was a 96% likelihood that the participants correctly identified the target. However, when the ATR was wrong, 30% of the time, the correct identification rate dropped to 79%, demonstrating a clear propensity towards automation bias ($\chi^2 = 29.3$, $p < .001$). As illustrated in Figure 5, these mistakes were made primarily by students in the *skill-based with ATR* condition who simply agreed with the automation and did not look for evidence to the contrary. Interestingly, there were several instances (4%) where the ATR was correct, but participants disagreed and found another incorrect target. Only five people made this mistake, and all but one was in the *Skill-based with ATR* student group.

In summary, the statistical analysis results revealed that regardless of the presence of advanced autonomy in the form of ATR or the fidelity of the training, participants' performance results were no different in terms of overall mission performance. However, experienced operators made fewer errors, as expected, but when they were not assisted by ATR, they also took the longest to search for targets, causing the longest system delays.

These results provide significant insight into the overall behavior trends of participants across the experimental factors but do not provide any information on how or why people took specific actions or made mistakes. What is needed is an analytic method to accompany traditional inferential statistics that not only bridges this gap but also gives clear design guidance either for new technologies or training interventions. We believe the hidden Markov modeling technique can provide such benefits, described in more detail in the next section.

Evaluating Supervisory Control Performance Through A Machine Learning Approach

Given that experiments with realistic testbeds often do not show clear performance advantages through traditional inferential testing, it is worth exploring if there are other analytic techniques that can provide additional insight. Previous research has indicated that the application of hidden Markov models, a machine learning modeling approach, can provide greater insight into operator strategies, particularly in terms of supervisory control environments like RESCHU. Indeed, a previous hidden Markov model analysis of a similar RESCHU environment was able to determine that despite having three different types of vehicles under their control, operators tended to cluster the vehicles into two categories, thus reducing their cognitive complexity and workload (Boussemart, Cummings, Las Fargeas, & Roy, 2011).

A Hidden Markov Model (HMM) is a stochastic model that describes a Markov process with some states and variables that are not observable (Rabiner & Juang, 1986). While system states and state transitions are observable for Markov models, in a Hidden Markov Model, system states are not directly observable (thus are 'hidden'), and the only observable variables are emission probabilities determined by hidden system states. HMMs have been used previously to develop human operator behavior models (Boussemart & Cummings, 2011; Pentland & Liu, 1999; Suzuki et al., 2005), but none of these previous efforts attempted to investigate the impact of different training and autonomy paradigms on operator image searching.

An HMM can represent both higher-level human operator behavioral states and lower-level operator interactions with human supervisory control systems. The observable emissions are determined by clusters of lower-level interactions between operators and an interface, for example when people click on an interface button. The clustering of related behaviors forms the

hidden states of an HMM, which then suggest operator strategies. For example, an operator generates observable states by clicking on a button to add a UAV waypoint and then another button to assign a new target, but the combination of these button actions indicates that an operator is executing the higher-level action of navigating, which is the hidden state.

Given that the presence of ATR affected the participants in this study in terms of search and wait times as well as misclassifications, we developed an HMM that represented operator interactions in the search camera window to determine if such an HMM could provide any useful insight into how operators searched the image for the correct target. There were 10,203 observations used to build this model, with an average of 165 observations per each test session. To this end, Figure 7 demonstrates the HMM of operator interactions in the search window, which were consistent regardless of the training group or level of expertise.

In Figure 7, operators generally exhibited two observed and three hidden states in their search efforts. The initial state in this task was the observed state Payload Engaged. This state was fully observed since operators had to right-click on a UAV–target pairing to select the Payload Engage function. Thus, this state indicates with no uncertainty when people began a search task. Once this button was selected, then the HMM model in Figure 7 reveals 3 distinct clusters of states with the relevant transition properties. The final state, Payload Finished, was also fully observed when people submitted their answer.

The three hidden states can be generally described as the Up state where participants predominantly searched up, the Lateral state where operators generally panned left and right, and the Zoom state people predominantly zoomed in and out on the target. These clusters are interesting since, for example, Up was its own state. People tended not to scan down, revealing a possible gap in training and understanding. Moreover, people tended to group lateral left and

right behaviors together and the large self-transition of 77% suggests that people had very inefficient search strategies. Self-transitions in HMMs indicate repetitive activity, and operators tended to repeat left and right camera movements, suggesting that this could be an area where either more training or better decision support could be helpful.

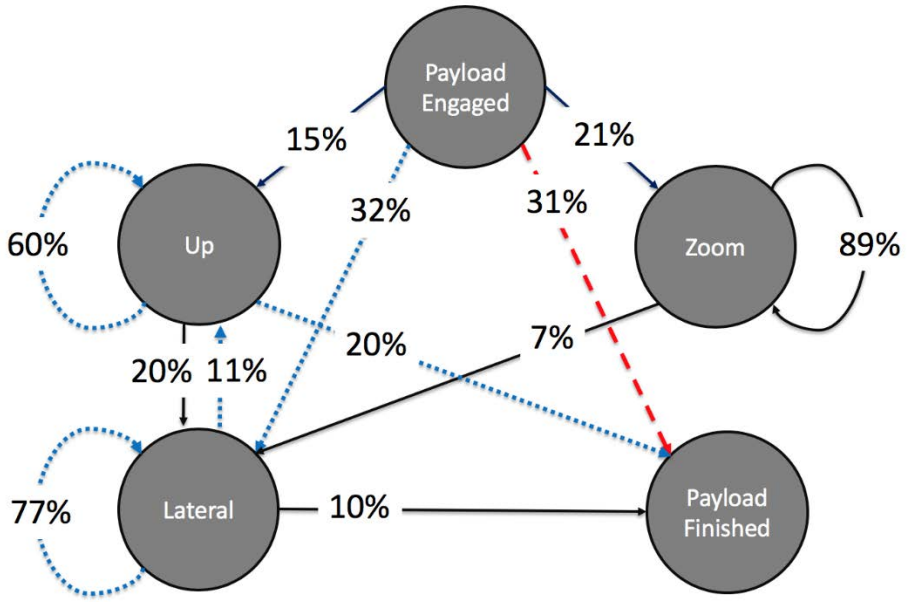

Figure 7. **RESCHU search task Hidden Markov Model. The red line indicates the dominant strategy for misclassifications and the blue dotted lines indicate the dominant strategy for correct classification.**

While the overall structure of the HMM is instructional in terms of how people clustered behaviors and where attentional inefficiencies likely reside, such a representation can also help diagnose problem behaviors. In Figure 7, the red dashed line shows the dominant strategy of people who made target misclassifications and the blue dotted line illustrates the dominant strategy of those people who correctly classified targets. The difference is quite stark. In agreement with the statistical analysis in terms of errors, people who made errors had a strong propensity to only look at the ATR's recommendation and then jump straight to the answer submission. They did not investigate any alternatives, as indicated by the move from the payload engage to finish observable state. In contrast, those people who correctly identified the targets,

even when they were assisted by ATR, still explored their region to ensure the answer was correct.

This is an important finding because while the inferential statistical analysis indicated a propensity towards automation bias, particularly for novices, the HMM analysis demonstrated how such a bias was manifested in actual behaviors. Armed with such results, designers now have a clear intervention in that ATR recommendations could provide a general area of a target instead of a specific point to reduce this bias. Another option would be to train operators to never simply accept a recommendation without first exploring the solution space, and also to specifically remember to search down in an image. Previous research has shown that HMMs could be used to develop automated training aids (Gombolay, Jensen, & Son, 2017), so we leave these areas for future work but believe these results demonstrate the utility of adding such an HMM analysis to any complex of human-computer study in a supervisory control setting.

DISCUSSION

Given that autonomous systems often remove the need for humans to perform low-level skill-based tasks and offer potentially lower training costs, there is a need to better understand how the removal of training for these tasks can potentially influence overall joint human-system performance. To this end, our goal in this study was to determine whether skipping skill-based training, as represented in the SRKE model, could have a demonstrable negative effect on performance in a human supervisory control task.

Using both traditional inferential statistics and a machine learning analysis in the form of a Hidden Markov Model, operators supervising multiple UAVs who were given focused skill-based training for an unfamiliar trackball input device did not have different search times in

panning and zooming a simulated camera, as compared to operators with no special training when the autonomy was correctly working. Thus, for this specific instance, the lack of skill-based training did not affect people's ability to search for a target in terms of time.

However, one concern with increased autonomy and reduced training is that this combination could lead to more mistakes, a loss of situation awareness and a tendency towards automation bias. This result was partially seen in this study, in that students with the most training and highest autonomy had the most misclassifications. So, while search times were not dramatically affected by differences in training and autonomy, error rates were, influenced but not as expected. Those people with the least training (both students and Part 107 operators) also had the least misclassifications, which is completely counter to our expectations. This result could have occurred because the additional skill-based training could have made people more complacent in the search task since it was highly automatic for them. When augmented with the HMM analysis, the lack of investigation of alternate targets was a clear contributing factor. The participants in the *ATR without skill-based training* condition were still learning to control the trackball while executing their searches, so this could have made them more vigilant, leading to the reduced misclassifications.

One other result worth highlighting is that overall as a group, the Part 107 commercial UAV operators were slower at searching for targets when they had no autonomy assistance, however, they also had the lowest misclassification rates, especially when the autonomy made an error. This statistically significant search time delay potentially suggests that the experienced operators were more cautious in this mode, even though they received the same training as the novices. This also may reflect their experience with looking through UAV cameras, which is difficult and has been compared to looking through a soda straw (Endsley & Jones, 2004).

The caution of the experienced operators is in sharp contrast to the novices who, with the most automated assistance and the most training, exhibited a tendency towards automation bias in that they had the highest error rates, especially when the automation made an erroneous suggestion. This result suggests that novices in the ATR without skill-based training group were very focused on their environment, likely because they were still developing their trackball manipulation skill during the experiment, which made them more careful in their selections.

So, these results are counter to our expectation that participants in the *ATR without skill-based training* condition would make more mistakes. Indeed, both novices and experienced operators had their lowest errors in this condition suggesting that their higher workload was not a problem. The additional trackball skill training for the novices in the *Skill-based with ATR* condition potentially made them more complacent. However, this was not the case for the experienced operators.

In part, this validates the SRKE model such that knowledge- and expert-based behaviors have distinct benefits and provide some level of protection against degraded system operation and unexpected uncertainties. However, significant work is needed to determine how much and what kind of training is needed to achieve these levels as well as how expertise does (or does not) transfer across domains.

There are many limitations to this work. First, these conclusions are based only on the RESCHU application and cannot be generalized across all multiple UAV testbeds. Moreover, while there were 30 subjects, a larger sample size could have led to different trends. For example, the 10 people in the *ATR without skill-based training* condition could have just been very good at adapting to an unfamiliar input device, and so another group may struggle with adapting to a new device with no training. This limitation also applies to the HMM. Such models

work best with significant amounts of data, but it is also important to note that they also indicate aggregate observed behavioral trends so cannot specifically predict any individual behavior.

Despite these limitations, there are many important lessons learned in this study which combined inferential statistics and a machine learning descriptive model. First, knowledge gained through significant experience in a similar but not exactly same domain may provide some protection against growing uncertainty in the form of degraded automation. More specifically, operators with real-world UAV experience appeared to recognize and exercise more caution when the automation was degraded in this multiple UAV simulation. On a related note, another critical result is that when the autonomy was unreliable, so were the humans that were novices, even despite additional skill-based training. Such propensity toward over trust and automation bias in autonomous systems could be extremely dangerous in safety-critical settings, and the reduction of training could lead to increased inappropriate trust and more cases of automation bias.

Lastly, through using a descriptive machine learning model in the form of a Hidden Markov Model, we gained new insights into the problems with extended search time and misclassifications that could form the basis for new training protocols and interventional technologies. While more data with more people would be needed on actual systems to determine more prescriptive interventions, the ability to acquire interface interaction data is relatively straightforward with supervisory control systems in both commercial and military systems, such as all types of UAVs, air traffic control, self-driving cars, and cockpits of airplanes. The use of machine learning analytic approaches has increased significantly in the past few years with the explosion of “big data,” but most of these efforts look at replacing human judgment as opposed to aiding in diagnosing human actions and judgments. What is markedly

missing is a focus on what patterns could emerge from the vast amount of training data generated during computer-based training sessions, which represent a large part of military and commercial training programs.

ACKNOWLEDGEMENTS

This research was sponsored in part by the US Army Aberdeen Training Ground and the Office of Naval Research Science of Autonomy program. We thank Ted Zhu, Varun Aggarwal, Andrew Hutchins, the Duke Marine Laboratory, the NCSU Institute for Transportation Research and Education (ITRE), and Wings of Carolina Flying Club for their help with the experiment.

AUTHORS' BIOGRAPHIES

Mary L. Cummings received her Ph.D. in Systems Engineering from the University of Virginia in 2004. She is currently a Professor at the Duke University Department of Electrical and Computer Engineering and also in the Department of Computer Science. She is the director of the Duke Humans and Autonomy Laboratory.

Lixiao Huang received her Ph.D. in Human Factors and Applied Cognition from North Carolina State University Department of Psychology in 2016. She worked as a postdoctoral associate at the Duke University Humans and Autonomy Laboratory from 2016–2018. Currently she is an assistant research scientist at the Center for Human, Artificial Intelligence, and Robot Teaming at Arizona State University.

Haibei Zhu received his B.S. degree in Electrical Engineering from Rensselaer Polytechnic Institute, NY, in 2015. He is a Ph.D. candidate in the Department of Electrical and Computer Engineering (ECE) at Duke University. He is currently a research assistant in the

Duke Humans and Autonomy Lab. His research interests include human computer interaction, data mining, and operator strategy prediction.

Daniel Finkelstein is an undergraduate student in Computer Science at Georgia Tech. He has worked in the Humans and Autonomy Lab as an undergraduate researcher.

Ran Wei received his B.S. degree in Industrial and Systems Engineering from Rutgers University in 2019. He is currently a graduate student in the Department of Industrial and Systems Engineering at Texas A&M University.

REFERENCES

Bainbridge, L. (1983). Ironies of Automation. Increasing levels of automation can increase, rather than decrease, the problems of supporting the human operator. *Automatica, 19*, 775-779.

Blacke, K. (2009). Tinker Airman graduates in first class of UAV operators, accessed Novermber 18, 2018.

Boussemart, Y., & Cummings, M. L. (2011). Predictive models of human supervisory control behavior using hidden semi-Markov models. *Engineering Applications of Artificial Intelligence, 24*(7), 1252-1262.

Boussemart, Y., Cummings, M. L., Las Fargeas, J., & Roy, N. (2011). Supervised vs. Unsupervised Learning for Operator State Modeling in Unmanned Vehicle Settings. *Journal of Aerospace Computing, Information, and Communication, 8*(11), 71-85.

Clegg, B. A., Heggestad, E. D., & Blalock, L. D. (2010). *The Influences of Automation and Trainee Aptitude on Training Effectiveness.* Paper presented at the Human Factors and Ergonomics Society 54th Annual Meeting, San Francisco.

Cummings, M. L. (2004). *Automation Bias in Intelligent Time Critical Decision Support Systems.* Paper presented at the AIAA 3rd Intelligent Systems Conference, Chicago.

Cummings, M. L. (2014). Man vs. Machine or Man + Machine? *IEEE Intelligent Systems, 29*(5), 62-69.

Cummings, M. L. (2018). Informing Autonomous System Design Through the Lens of Skill-, Rule, and Knowledge-Based Behaviors. *Journal of Cognitive Engineering and Decision Making, 12*(1), 58–61.

Cummings, M. L., Bertucelli, L., Macbeth, J., & Surana, A. (2014). Task versus vehicle-based control paradigms in multiple unmanned vehicle supervision by a single operator. *IEEE Transactions on Human-Machine Systems, 44*(3), 353-361.

Cummings, M. L., Buchin, M., Carrigan, G., & Donmez, B. (2010). Supporting Intelligent and Trustworthy Maritime Path Planning Decisions. *International Journal of Human Computer Studies, 68*(10), 616-626.

Dzindolet, M. T., Peterson, S. A., Pomranky, R. A., Pierce, L. G., & Beck, H. P. (2003). The role of trust in automation reliance. *International Journal Human-Computer Studies 58*, 697-718.

Endsley, M. R., & Jones, D. G. (2004). *Designing for situation awareness: an approach to user-centered design* (2nd ed.). Boca Raton, FL: CRC Press, Taylor & Francis Group.

Feigh, K. M., & Pritchett, A. R. (2014). Requirements for Effective Function Allocation: A Critical Review. *Journal of Cognitive Engineering and Decision Making, 8*(1), 23–32. doi:10.1177/1555343413490945

Ferris, T., Sarter, N., & Wickens, C. (2010). Cockpit automation: still struggling to keep up *Human Factors in Automation* (2nd ed.). Amsterdam NL. : Elsevier.

Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology, 47*, 381-391.

Gombolay, M. C., Jensen, R., & Son, S.-H. (2017). Machine learning techniques for analyzing training behavior in serious gaming. *IEEE Transactions on Computational Intelligence and AI in Games.* doi:10.1109/TCIAIG.2017.2754375

Gutzwiller, R., Clegg, B., & Blitch, J. (2013). Part-Task Training in the Context of Automation: Current and Future Directions. *The American Journal of Psychology, 126*(4), 417-432.

Idris, H., Enea, G., & Lewis, T. A. (2016). *Function Allocation between Automation and Human Pilot for Airborne Separation Assurance.* Paper presented at the International Federation of Automatic Control.

Kaber, D. B. (2018). Issues in Human–Automation Interaction Modeling: Presumptive Aspects of Frameworks of Types and Levels of Automation. *Journal of Cognitive Engineering and Decision Making, 12*(1), 7–24. doi:10.1177/1555343417737203

Lee, J. D., & See, K. A. (2004). Trust in technology: Designing for Appropriate Reliance. *Human Factors, 46*(1), 50-80.

Lin, J., Wohleber, R., Matthews, G., & Funke, G. J. (2015). *Video Game Experience and Gender as Predictors of Performance and Stress During Supervisory Control of Multiple Unmanned Aerial Vehicles.* Paper presented at the 59th International Annual Meeting of the Human Factors and Ergonomics Society, Los Angeles, CA.

MacKenzie, I. S. (1995). Movement time prediction in human-computer interfaces. In R. M. Baecker, W. Buxton, J. Grudin, & S. Greenberg (Eds.), *Readings in human-computer interaction* (2nd ed., pp. 483-493). San Francisco: Kaufmann.

MacKenzie, I. S., Kauppinen, T., & Silfverberg, M. (2001). *Accuracy measures for evaluating computer pointing devices.* Paper presented at the ACM Conference on Human Factors in Computing Systems - CHI 2001, New York.

Mittu, R., Sofge, D., Wagner, A., & Lawless, W. F. (Eds.). (2016). *Robust Intelligence and Trust in Autonomous Systems.* New York: Springer.

Moray, N., Inagaki, T., & Itoh, M. (2000). Adaptive Automation, Trust, and Self-Confidence in Fault Management of Time-Critical Tasks. *Journal of Experimental Psychology: Applied, 6*(1), 44-58.

Mosier, K., Skitka, L. J., Heers, S., & Burdick, M. D. (1998). Automation bias: decision making and performance in high-tech cockpits. *International Journal of Aviation Psychology, 8*(1), 47 - 63.

Nehme, C. E. (2009). *Modeling human supervisory control in heterogeneous unmanned vehicle systems.* (Doctor of Philosophy), Massachusetts Institute of Technology, Cambridge, MA.

Parasuraman, R., Sheridan, T. B., & Wickens, C. (2008). [Situation Awareness, Mental Workload, and Trust in Automation: Viable, Empirically Supported Cognitive Engineering Constructs].

Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (2000). A Model for Types and Levels of Human Interaction with Automation. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 30*(3), 286-297.

Pentland, A., & Liu, A. (1999). Modeling and prediction of human behavior. *Neural Computation, 11*, 229-242.

Rabiner, L., & Juang, B. (1986). An introduction to hidden Markov models. *ASSP Magazine, IEEE [see also IEEE Signal Processing Magazine], 3*(1), 4-16.

Ratches, J. A. (2011). Review of current aided/automatic target acquisition technology for military target acquisition tasks. *Optical Engineering, 50*(7), 072001. doi:https://doi.org/10.1117/1.3601879

Schenk, S., Lech, R. K., & Suchan, B. (2017). Games people play: How video games improve probabilistic learning. *Behavioural Brain Research, 335*, 208-214.

Strauch, B. (2018). Ironies of Automation: Still Unresolved After All These Years. *IEEE Transactions on Human-Machine Systems, 48*(5), 419-433. doi:10.1109/THMS.2017.2732506

Suzuki, T., Sekizawa, S., Inagaki, S., Hayakawa, S., Tsuchida, N., Tsuda, T., & Fujinami, H. (2005). *Modeling and Recognition of Human Driving Behavior based on Stochastic Switched ARX model.* Paper presented at the 44th IEEE Conference onDecision and Control CDC-ECC '05. , Seville, Spain.

Wang, W., Hou, F., Tan, H., & Bubb, H. (2010). A Framework for Function Allocations in Intelligent Driver Interface Design for Comfort and Safety. *International Journal of Computational Intelligence Systems, 3*(5), 531-541.

Wickens, C. D., & Dixon, S. R. (2007). The benefits of imperfect diagnostic automation: a synthesis of the literature. *Theoretical Issues in Ergonomics Science, 8*(3).

Wickens, C. D., & Hollands, J. G. (2000). *Engineering Psychology and Human Performance* (3rd ed.). Upper Saddle River, N.J.: Prentice Hall.