Design Principles for AI-Assisted Attention Aware Systems in Human-in-the-Loop Safety Critical Applications

Max Nicosia(✉) and Per Ola Kristensson

Department of Engineering, University of Cambridge, Cambridge CB2 1PZ, UK
{lmn27,pok21}@cam.ac.uk

Abstract. AI-assisted, attentiona-ware systems support operators in detecting and managing targets present in visual scenes. Such a system typically attempts to automatically identify targets of interest and increase the probability that an operator can detect them by, for example, modifying their visual saliency in the visual scene. Applications of AI-assisted attention awareness include air-traffic control, submarine demining and armored vehicle situational awareness. This chapter explains the key human-machine challenges intrinsic in this design problem and distills six design principles based on a functional design of a general AI-assisted attention-aware system for target identification.

Keywords: Attention awareness · Situational awareness · System design

1 Introduction

Application complexity is of particular pertinence in safety critical applications, where additional information can potentially assist the operator in making better decisions in difficult situations. A typical example of an application is an air-traffic control system, where the application displays a variety of information linked to multiple tasks. This application complexity means that operators have to maintain a high level of situational awareness, and therefore, may suffer from cognitive overload.

Traditionally, safety critical applications strive to mitigate this complexity with safety features that are intended to ensure operators are informed of specific noteworthy events through warnings or alarms. Operators of these applications go through extensive training to ensure that they have internalized the application's operational features, are familiar with the locations of various critical information data points, and have sufficient experience to preempt or mitigate various possible operational situational outcomes. However, such training is costly and time consuming, and does not guarantee optimal performance in practice. As a result,

W. F. Lawless et al. (Eds.): Engineering Artificially Intelligent Systems, LNCS 13000, pp. 230–246, 2021.
https://doi.org/10.1007/978-3-030-89385-9_14

in some particular complex domains, operators are required to go through several years of work experience before they have achieved the required level of proficiency.

AI-assisted target identification applications aim to overcome this problem by providing some level of assistance to the operator during operation. At the most basic level an AI-assisted target application seeks to manage the attention of the operator so they are aware of important or relevant events and pieces of information relevant for the current task (targets) while minimizing distracting information (distractors). To successfully achieve this, the AI-assisted application needs to make use of sensor data to infer and track the operator's focus of attention, detect important targets within the application, and deploy various subtle visualization techniques to draw the operator's attention towards relevant information at an optimal time without jeopardizing overall task performance.

Designing AI-assisted applications is not a straightforward task. Their success hinges on them being able to both accurately track the operator's focus of attention, their capacity to accurately detect targets, and their ability to preempt the operator's intentions or actions. Typically, they adjust the saliency of relevant information on a display using a mechanism that amplifies the operator's decision making capabilities while avoiding to inadvertently generate further distractors or disruptions. For an AI-assisted application to achieve balance, it needs to be designed specifically for the application's domain and it needs to dynamically adjust any operator intervention strategies based on current operator performance and actions, and in response to the application's current data stream.

The realization of AI-assisted applications is riddled with pitfalls. In this chapter we seek to assist system designers in avoiding those pitfalls by describing six system design principles necessary for the development of such a system. These principles are based on the functional design of a general AI-assisted target identification application, which we also introduce in this chapter.

In Sect. 2 we review prior work on similar applications and the challenges of presenting information to the operator in an optimal way. In Sect. 3 we discuss the human-machine challenges, our approach for tackling situational awareness, and the challenges of relying on binary classifiers for separating targets from distractors. In Sect. 4 we present a functional architecture for a generic AI-assisted target identification system. In Sect. 5 we use this functional description to distill six system design principles for AI-assisted target identification systems. In Sect. 6 we discuss the challenges and limitations of the design, and in Sect. 7 we conclude.

The main contribution in this chapter is the functional architecture of an AI-assisted target identification system and the six system design principles we distill from this functional description to address the human-machine challenges raised by a safety critical application.

2 Related Work

The human factors literature has extensively studied the potential benefits of adaptive guidance in human-machine systems (e.g. [27,31,32,37]). The efficacy of these complex systems is determined by the system managing the information presented to the user in such a way that it maximizes task performance. The literature has primarily focused on studying interruptions and information presentations.

By focusing on interruptions and information presentation, human factors researchers have attempted to balance attention allocation between bottom-up and top-down processes [33]. Bottom-up processes are said to be automatic and capture the user's attention without the user consciously acknowledging them; for example, becoming aware of a stimulus. Top-down processes require the user's conscious action and effort; for example, reading. Attention is understood to be a finite resource. However, there is still much debate about how attention is allocated and consumed across various stimuli, such as which stimuli will enter working memory and consciousness, and which stimuli will be acted on and with what response. This definition of attention as a finite resource that is subdivided among processes has become known as attention by selection [1,11,23,30,34].

Determining the rules that dictate which stimuli will capture the attention of a bottom-up or top-down process has proven difficult in some cases. For example, very salient stimuli can remain completely unnoticed under certain circumstances, such as when the user is engaged in another task demanding attention to other stimuli at the same time [38]. Other stimuli, such as alarms [42] and moving or looming objects [16], capture attention consistently. Somewhat contradictory to intuition, increasing the mental load of the operator has been shown to reduce distractor interference and increase the capacity for stimuli perception [22]. Desimone et al. [7] observed that stimuli relevant to the current user task tended to be favored for processing and entering consciousness.

Another reason for the literature to focus on interruptions is that interruptions affect working memory by causing retrospective and/or prospective memory failures. Prospective memory is what allows users to function—remembering to remember what we need to do or be aware of in the future. Retrospective memory is what is normally referred to as simply recalling something from the past. Einstein et al. [12] suggest that cues in the environment can trigger automatic-associative memory and lead to activating actions associated with the appropriate stimulus. In light of these results, the human factors literature has focused on manipulating the presentation of information to reduce the negative effects of interruptions.

The effects of interruptions on task performance have been studied extensively both in general (e.g., [5,21,26]) and in task-specific fields, such as air-traffic control research (e.g., [20,41]). Avoiding interrupting the user during the task and/or delaying the interruption until the latter half of the task has been observed to be effective [5]. More advanced strategies include context-sensing and/or using the contents of the message (notification) to infer an optimal time to interrupt the operator [18,26]. In office settings, listening for voices or noises has been shown to be effective for determining an appropriate time to interrupt users [15].

Saliency changes can be used to minimize change blindness [39] and inattentional blindness [38]. However, saliency changes are difficult to use effectively as there are many factors that affect how they are perceived. For example, Healey et al. [19] found that hue and target orientation work best for numerical estimations, while colors are discriminated by their distance, category and linear separation. Textures are discriminated by their size and density. In the case of multi-dimensional data, Healey et al. [19] found that the best strategy was to reduce feature interference as much as possible. Simple motions have been observed to be easily detectable both in the near and far field of view, perform well in peripheral visualizations, and interfere less with color and hue features [3].

Despite all of the benefits of saliency manipulation, complex changes in saliency can easily cause negative effects. A study focusing on air-traffic controllers demonstrated that some visualizations resulted in adverse effects on secondary task performance while only simple and subtle visualizations, such as pulsating objects, were relatively effective [20].

Prior work has used sensors to estimate the position and presence of users to expose them to proximity-aware applications [9,25]. Further research used multi-sensing to expose contextual information that allowed the development of more complex strategies to avoid interruptions. Gellersen et al. [18] used various external sensors to feed contextual information into mobile phones to reduce interruptions. Lopez-Tovar et al. [24] used a Bayesian network to enable smartphones to infer the correct notification preference for each user based on previous user choices and contextually sensed information during meetings.

The idea of extending system design and development with sensors to create attention aware systems has been advocated in the literature [2,36]. Such solutions aim at providing general purpose solutions that reduce interruptions and improve cognitive abilities of operators. Toolkits to manage attention by pushing notifications towards peripheral displays have been explored [36]. Other prior work investigated strategies, such as reducing interruptions, managing context switches, and tagging actively used objects, to improve task resumption [8,35]. Task-assisted resumption was shown to be effective in a learning system that managed interruptions and attention changes during learning sessions [35]. Tagging or marking the current position on the current task during notification interruptions was shown by Cutrell et al. [5] to be ineffective as improvement was only noticeable when no interruptions were present. In an earlier study, Czerwinski et al. [6] demonstrated that interruptions during the early stages of the task, that is, before the user entered the planning, execution or evaluation stage, reduced overall task performance time. However, users experienced significant disruptions when the interruptions occurred later in the task and required the use of reminders to resume the current task [5]. Bailey and Konstan [2] suggested using a two-level hierarchy within tasks to separate coarse events from fine events. They observed that interruptions between coarse events yielded less disruption.

Dostal et al. [10] used RGB cameras to develop an inattention aware multi-display system that detected if an operator was attending to a particular display. It used this information to derive subtle visualizations in unattended displays to reduce distractions while allowing the operator to perceive changes in peripheral vision. Garrido et al. [17] extended this system into a graphical user interface (GUI) toolkit.

Other recent work has used eye-tracking to increase the saliency of unobserved changes in a radar task [40]. However, the results were mixed as the system did not have the knowledge of several application-dependent factors, such as the task, the situational context, and the current workload of the operator. Nicosia and Kristensson. [28] developed an inattention management middleware to incorporate several of the techniques previously discussed in multi-display setups. The middleware provided a software layer that allowed distributed applications to query the status of the operator and trigger specific visualization logic as a result of operator action, attention and task performance. The idea of a dynamic system was based on allowing the operator to further their understanding about their current task and performance, as established in the situational awareness literature [13]. The system was later extended to support a consumer-producer competitive task setup in which the system dynamically adjusted the saliency of each target based on operator performance against externally supplied expected performance metrics [29]. The system effectively demonstrated improvement in performance for some of the task metrics, but it also highlighted the complexity of balancing dynamic strategies for multiple conflicting tasks.

3 Human-Machine Challenges

There are four challenges that need to be addressed in the design of an AI-assisted target identification system.

First, the system has to be able to separate targets from distractors. A *target* is a data point of any number of dimensions that is relevant to the operator at a particular point in time. A *distractor* is any data point that detracts, prevents, delays or confuses the operator in carrying out a task. Additionally, the system has to provide flexibility to manage incorrectly classified targets (false positives). We discuss this later in more detail in Subsect. 3.2.

Second, the system has to be able to account for target and task priorities. This accounting allows the system to apply specific strategies to ensure that the operator can distinguish the importance of specific targets and their relation to the current task.

Third, the system has to be capable of conveying all of the previously discussed information to the operator in such a way that it does not reduce task performance or overload the operator.

Fourth, the system has to be able to identify if the operator has failed to notice a specific data point that has been deemed important or relevant for a specific task (a target), and to be able to construct a strategy that brings such information to the operator's attention without compromising overall task performance.

3.1 Attention and Situational Awareness

There are several models that explain Situational Awareness (SA). We will focus on Endsley's model [13] and Situated Situational Awareness [4] as they are the most suitable models for addressing the complexity of AI-assisted systems.

Endsley's model [13] describes the process of building SA by identifying three stages of the operator. These stages are: 1) perception of the elements in the environment; 2) comprehension of the current situation; and 3) projection of future states. Depending on how the operator approaches the task and what their current goal is, the operator may progress through these stages in order, jump between them, or iterate through them to arrive at a particular SA level.

These stages do not guarantee that the operator will build a correct SA model—the stages merely describe the operator's cognitive activities. For example, if the operator's focus of attention are not on important information at that point in time, the operator will not perceive it and, subsequently, will build an incorrect or incomplete understanding of the current situation and its potential future states. Similarly, if the information is presented in a confusing or unclear manner, the operator may perceive it, but the operator may still build an incorrect understanding of its importance and its future potential states. Finally, an operator could still arrive at an incorrect SA model even if all of the information has been presented in an optimal manner due to external factors outside of the system, such as, for example, a failure to recall critical information, stress, fatigue, or poor decision-making skills.

In the Situated Situational Awareness model [4], the operator builds the SA by repeatedly sampling the environment for limited amounts of information based on its relevance. This model emphasizes SA building on the working memory of the operator and the capacity to maintain and update it. Endsley [14] criticized this approach, suggesting that expert operators will normally build their SA using their understanding of the possible projection scenarios from information stored in their long-term memory to reduce the necessity to continuously sample and notice relevant information. Further, Endsley [14] argued that operators will seek to validate their understanding or complete their projections actively, as maintaining full SA in working memory is suboptimal for experts and in most cases almost impossible.

The approach taken in this chapter is that the solution for an AI-assisted target identification system lies somewhere in between. We agree that expert operators will draw from their experience in long-term memory. However, it is also important that the system leverages the potential capacity for associative memory and automaticity of action that can arise from manipulating the relevance of information at a specific point in time.

An AI-assisted system needs to ensure that the information is presented in a clear manner to the operator, and that the operator's focus of attention is on the relevant data point at the correct time. As such, managing attention is instrumental to increasing the probability of the operator building a successful SA model and to reducing any adverse influence.

3.2 Classifier Interaction

A target identification system can be fundamentally viewed as a binary classifier that attempts to identify *targets* among *distractors*. In a binary classifier there are four possible outcomes: 1) a true positive (TP)—a target is correctly classified as a target; 2) a false positive (FP)—a distractor is incorrectly classified as a target; 3) a true negative (TN)—a distractor is correctly classified as a distractor; and 4) a false negative (FN)—a target is incorrectly classified as a distractor.

These four outcomes give rise to a set of metrics for understanding the binary classification performance of the system. A Receiver Operating Characteristic (ROC) curve is used to analyze the operating envelope of a binary classifier. A perfect classifier has 100% TP and TN rates. A random classifier, similar to a coin toss, has an even distribution of TP, FP, TN and FN. Classifiers that are worse than random classifiers can be reversed to increase their *positive* predictive power. The area under the ROC curve, also called the *c-statistic*, can be used to calculate the predictive power of the classifier as this area will reflect the probability that the classifier will score a randomly chosen *positive* outcome higher than a randomly chosen *negative* outcome. The higher this probability, the better the classifier.

Fundamentally, an AI-assisted target identification relies on an AI module to perform binary classification on information presented to the operator. This information may be either correctly classified as relevant (a true positive target), correctly classified as irrelevant (a true positive distractor), incorrectly classified as relevant (a false positive target), or incorrectly classified as a distractor (a false positive distractor). While great efforts are spent maximimzing classification performance, it is important to realize and accept that incorrect classifications are unavoidable in practice. Therefore, any system solution must assume the presence of both incorrectly classified targets (which now serve as dangerous distactors to the operator) and incorrectly classified distractors (which may potentially obscure critical information).

4 Functional Architecture

We here present six system design principles for AI-assisted target identification systems. We introduce a generic high-level functional architecture for this purpose in the form of a function-structure model and use this model to distill design principles.

The function-structure model of the joint human-machine system is shown in Fig. 1. Dashed arrows indicate signal flows. The overall function is `Detect and Process Target`. The overall function outputs a signal *Processed Targets* in response to two *Sensor Data* signals, which may be identical, or be received from an identical signal source, however, this is not a requirement. The overall function is decomposed into nine key subfunctions. We separate functions into two categories: 1) AI-functions, which are carried out by a technical embodiment of the system; and 2) operator functions, which are carried out by a human operator.

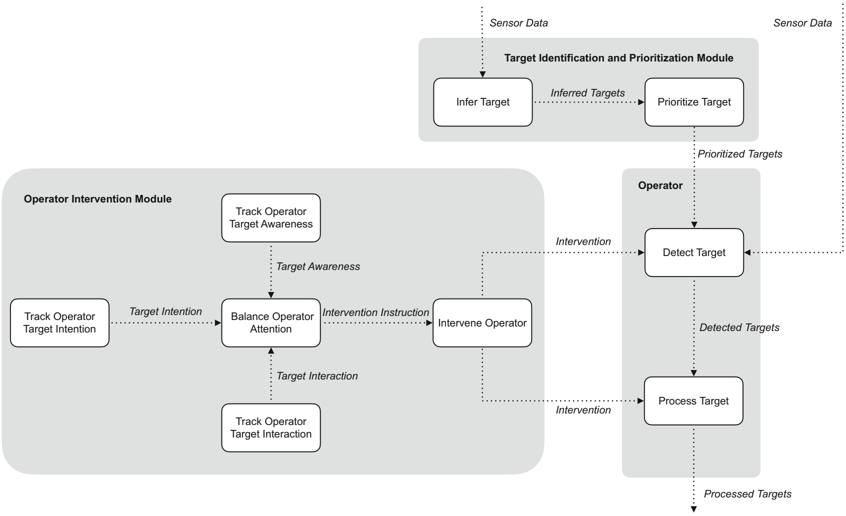

Fig. 1. A function-structure representation of an AI-assisted target identification system with operator inattention support.

Fundamentally, the functional architecture in Fig. 1 identifies three modules: 1) a human *operator*; 2) an AI-based *target identification and prioritization* module; and 3) an AI-based *operator intervention* module.

The AI-function `Infer Target` receives a *Sensor Data* signal and infers any targets in the sensor stream. The output of this function results in an *Inferred Targets* signal transmitted to the AI-function `Prioritize Target`, which assesses the importance of each target and determines its priority for an operator's attention. Collectively, the two AI-functions, `Infer Target` and `Prioritize Target`, represent a machine learning module that identifies and prioritizes targets in a scene.

The first operator-function, `Detect Target`, detects targets, which can be both detected directly by the operator in a raw *Sensor Data* signal, or presented to the operator as a *Prioritized Targets* signal. The second operator-function `Process Target` receives the signal *Detected Targets* and performs actions in response to them. This last step results in `Process Target` finally outputting a *Processed Targets* signal from the system. Collectively, the two operator-functions, `Detect Target` and `Process Target`, represents a human operator tasked with identifying and processing targets on a display assisted by an automatic target identification system.

However, merely assisting the operator by automatically identifying and prioritizing targets may not be sufficient. First, the AI-functions `Infer Target` and `Prioritize Target` are likely to occasionally generate erroneous results, such as failing to identify targets or incorrectly identifying targets (distrators). The presence of such uncertainty around prioritized targets increases the cognitive load on an operator; Second, the cognitive load on the operator may also increase further, for example, as a result of operator overload, fatigue, stress, etc.

Therefore, a critical component of a well-functioning AI-assisted target identification system is an operator intervention module. This module uses three AI-functions to collectively track the operator's target awareness, target intention and target interaction (`Track Operator Target Awareness`, `Track Operator Target Intention` and `Track Operator Target Interaction`). For clarity, the input signals to these functions are not shown in Fig. 1.

These three AI-functions generate corresponding signals to the AI-function `Balance Operator Attention`, which is tasked with managing the operator's current ability to identify and process targets. If the operator is overloaded, `Balance Operator Attention` sends an *Intervention Instruction* signal to the AI-function `Intervene Operator`, which can intervene with the two operator-functions `Detect Target` and `Process Target` by sending them an *Intervention* signal.

5 System Design Principles

The AI-functions `Infer Target` and `Prioritize Target` are domain-specific applications of machine learning infrastructure. We are therefore here concerned with distilling six system design principles for improving the operator functions `Detect Target` and `Process Target` by realizing effective operator interventions.

5.1 Determine Operator Focus of Attention

Establishing the focus of attention is the first necessary step in order to track operator awareness. Determining the operator's focus of attention requires reconciling target locations with the sensor stream or detecting targets from application data events. The specific realization of this function will be application specific, as different domains will have different constraints or requirements on how data is represented, as well as the number of dimensions that are represented. Data dimensions can include, for example, color changes, location changes, and value changes. Data may also consist of video or audio streams, still images and visualized sensor data (such as sonar data).

Regardless of these factors, a typical implementation will use eye-tracking data to convert fixations into operator attention focus points. A fixated target, or area, can then be used by the system to determine that a data point has been recognized by the operator. Further program logic can be constructed based on the duration of the fixation and the proximity of the fixation to the target data point. However, such modifications will be domain dependant.

The primary concern in the realization of this function is in the reconciliation of the sensor data and the application's data points. The first challenge arises due to how the data is presented in the application. Applications that display data points in clusters or in close proximity to each other will cause the reconciliation of events and data points to be non-deterministic as there will inevitably be uncertainty around the correctly determined operator's focus of attention. In such cases, the system will need to operate under the assumption that the operator fixated on all nearby data points and apply some further heuristics based on the likelihood of specific data points being noticed during the fixation time, or use a corroboration mechanism by highlighting (or moving) subsets of data points, that reveal to the user what the system has recognized.

The second challenge arises due to the system misclassifying an operator's focus of attention due to the operator zoning out, possibly due to sleep deprivation, information overload, stress, or trauma, and the system incorrectly detecting this behavior as an eye-tracking fixation and hence an operator's focus of attention. Since the operator's mind is unobservable, the system will need to filter such instances out. Possible solutions include: 1) require additional data from the primary attention-tracking channel (for example, an eye-tracker) to confirm the focus of attention, such as increased fixation duration, or a series of fixations; 2) require additional data from a secondary attention-tracking channel (such as a head tracker) to confirm the focus of attention (for example, the operator rotating its head in the direction of the inferred focus of attention); and/or 3) track the operator user interface activity to confirm focus of attention by, for example, analyzing the mouse cursor movement in the vicinity of the hypothesized point of the operator's focus of attention.

5.2 Track Operator Target Awareness

The ability to determine operator awareness separates a dynamic *intelligent* system from a system that can only react using fixed predetermined output for each configured situation. The system can only produce a tailored output signal that addresses the current operator state if the system incorporates information about the operator's target awareness.

A system able to establish the operator's awareness of each target on the display, by reconciling the operator's focus of attention with data points on the display, can infer which targets the operator may be unaware of. This inference permits the system to predict what the operator has perceived and thus allows the system to estimate the level of SA of the operator. The system can then transmit this signal, together with other output signals, to the `Balance Operator Attention` function.

Realizing this function requires the system to reconcile data point changes, data point priorities, and associated operator events. The implementation of this function is typically relatively straight-forward, assuming the `Determine Operator Focus of Attention` function and the `Infer Target` and `Prioritize Target` functions are reliable, that is, they exhibit low false-positives rates.

5.3 Track Operator Target Intention

Once the system has acquired knowledge about which targets the operator has perceived, the system can use this information to infer what the operator comprehended of the current application state and attempt to use this inference to project what the operator's intentions are towards any targets on the display. The previous operator interaction, or logged traces of previous operator performance, can be used to guide the system's inference. Having the system generating hypotheses of operator intentions can be used to assess current operator performance and incorporate error prevention strategies to preempt future operator errors.

For example, if the operator has recently fixated on a set of specific targets, it is probable that the operator will carry out actions related to those targets in the near future. Correspondingly, the operator is less likely to follow-up on actions for targets that have not been fixated on recently. Similar logic can be applied for tasks within the application. The system can infer if a specific task is being neglected based on which targets are being fixated upon or interacted with by the operator the most. The output of this function will be used by the `Balance Operator Attention` function to estimate future operator actions and whether they will be in line with expected actions and baseline performance.

The primary challenge in realizing this function is accurately inferring the operator's intention within the application's domain for specific circumstances and actions. One approach to implement this function is to have the system use a task template matching mechanism to predict the likelihood of specific follow-up actions as a result of target fixations and then match these predictions with actual operator outcomes. Since intention is closely related to task execution, the system needs to account for the specifics of each task within the application and its domain.

A related challenge that can arise in some applications is that an operator's actions may be ambiguous and thus match multiple task templates at once. In such cases, the system can use two strategies: 1) the system can use past history or other data feeds about the overall state of the system to predict operator intentions; and 2) the system can operate under the assumption that all of the operator's intended actions take place at the same time and allow the `Balance Operator Attention` function, which has access to both *Target Interaction* and *Target Awareness* signals, to perform more accurate inferences of the operator's intention.

5.4 Track Operator Target Interaction

The operator's interaction with the system is a rich source for the system to infer the current awareness state of the operator and possible future state projections. The system can also use the information associated with an action to estimate its impact on the overall future application state. The degree to which the system can estimate this information depends on the application's domain.

Nevertheless, the system can gain important operator insights by tracking the operator's interactions.

The operator's interactions can be classified into two types: 1) definite actions; and 2) precursors to definite actions. The first type of interaction can inform the system of the culmination of the awareness of the operator. The second type of interaction can inform the system about potential definite actions that the operator may take. For example, when the operator selects a group of targets, the operator performs an interaction of type 2. When the operator executes an action for selection, the operator performs an interaction of type 1.

Both of these types of interaction provide information on the operator's focus of attention. If the user has interacted with a particular target, or a set of targets, the operator is definitely aware of them. Conversely, this information also informs the system about what the operator is *not* aware of (inattention). A system having knowledge about the current operator's awareness is a system capable of both estimating future system states and the level of comprehension of the operator of any targets and potential follow-up actions. For example, the system is able to detect if an operator is acting on low priority targets instead of high priority targets, or whether an operator is neglecting one task in favor of another task.

The system can also estimate delays and the costs of actions by examining the timing associated with task execution and target fixations. Such statistics can then be used to detect anomalies, for example, fatigue or lapses in individual operator performance. These statistics can then be propagated to the `Balance Operator Attention` function to allow the system to execute an appropriate operator intervention tactic. Alternatively, or in parallel, these statistics can be used to evaluate the effectiveness of prior operator intervention tactics.

5.5 Balance Operator Attention

Balancing operator attention is carried out by the system reconciling all information from the `Track Operator Target Awareness`, `Track Operator Target Intention` and `Track Operator Target Interaction` functions in order to decide if an *Intervention Instruction* signal should be sent to the `Intervene Operator` function in the system.

Balancing operator attention is necessary to manage the operator's attention, which is a limited resource as explained in the Related Work Sect. 2 previously in this chapter. Balancing operator attention means distributing operator attention across tasks and targets to ensure the operator chooses the most relevant actions at any given time.

To balance the operator's attention, it is necessary for the system to have knowledge about the operator's awareness of any targets, the operator's intentions regarding targets, and the operator's interaction with any targets. Using this information, the system can balance the operator's attention in several ways.

First, the system can balance operator attention by assessing the operator's performance in comparison to an expected benchmark. For example, the system can estimate changes in target priorities as a result of possible operator actions and compare such changes against previous actions to assess if the operator's actions are resulting in expected performance. If the observed operator performance is below a threshold, the system can intervene with the operator; for example, by drawing the operator's attention to higher priority targets.

Second, the system can balance operator attention by distributing attention across tasks. Successfully achieving this distribution depends critically on the system's ability to infer the operator's target awareness, target intention and target interaction.

A challenge in balancing operator attention in an optimal fashion is the fact that the system is dynamic—the system changes states both due to new incoming data and to operator actions, which may or may not be influenced by feedback from the system itself (such as an operator intervention). One way to tackle this challenge is to associate specific operator actions with potential application state outcomes. System applications that have clear definite actions are able to evaluate outcomes with higher a accuracy compared to system applications where outcomes are not easily attributed.

Hence, balancing operator attention is challenging. To avoid the main pitfalls, as discussed in the Related Work Sect. 2 previously in this chapter, it is advisable to: 1) avoid unnecessary interruptions; 2) ensure that relevant information is clearly shown without confounding the operator; and 3) prevent adverse effects, such as inducing additional confusion or stress.

5.6 Intervene Operator

When the system intervenes with the operator the system performs an action in response to an *Intervention Instruction* signal from `Balance Operator Attention`.

Operator interventions can be carried out in a number of ways. The central idea is to modify operator behavior to respond to targets or tasks in a domain-appropriate way. One straightforward example is for the system to increase the visual saliency of targets the system believes the operator is unaware of.

Realizing such a function is fundamentally application specific as these operator interventions must be carefully designed to not intrude on established workflows. Poorly designed operator interventions can confuse the operator or add further cognitive overload.

A typical way to achieve an operator intervention, as previously mentioned, is modifying visual saliency. Visual target attributes, such as hue, color and movement, can be modified to attract an operator's attention to specific targets or individual data attributes. This method necessitates careful design work to ensure the visual attribute modifications are compatible with existing workflows and the visual grammars used in the domain.

The primary challenge in the realization of a needed function is in ensuring any operator intervention does not detract the operator from other critical targets or tasks and that any operator intervention has a clear purpose and does not result in unexpected operator behavior. This requirement necessitates that interventions are coordinated to ensure that they do not clash, overload or confuse the operator.

6 Discussion

A particular challenge for any AI-assisted target identification system is realizing the `Balance Operator Attention` and `Intervene Operator` functions. As previously discussed, these two functions are critical in enhancing operator performance. Fundamentally, this means ensuring that the operator's attention is optimally allocated to allow the operator access to the highest quality information at any given point in time.

The efficacy of the `Balance Operator Attention` function depends on two factors: 1) the quality of logged traces available for predicting expected operator actions based on the estimated states of the operator and the application; and 2) the capacity for establishing a baseline that allows the system to evaluate the current application's state to determine when to intervene the operator.

The `Balance Operator Attention` function is one of the distinguishing features that separates an AI-assisted target identification system from static systems that rely on simplistic tactics, such as highlighting important information regardless of the current state of the operator or application. This capacity to balance operator attention for the joint human-machine system to adapt to the operator's awareness and the current application state is critical for the successful operation of challenging tasks in this area. On the other hand, this very same system adaptability is what the literature has identified as being a source of potential pitfalls and challenges. Therefore, any `Balance Operator Attention` function must be realized with great care.

In general, an AI-assisted target identification system needs to ensure that it does not cause detrimental performance. This case requires a careful consideration of many design parameters and functions, e.g. as determining when to intervene, how to intervene, and how to verify and validate that performance does not degrade under certain task conditions, such as during an unusually high complexity period of operation.

A benefit of the functional architecture introduced in Fig. 1 is that by turning off the `Intervene Operator` function, the output of the `Balance Operator Attention` function can be used to assess operator performance live. The data can then be used to identify any common operational challenges that are experienced under particular operational situations, as well as any weaknesses in the design of the system application. This data also serves to establish a baseline, which can be used to assess the improvements induced by an AI-assisted target identification system that re-couples these two functions. The logged data can also be used to gain further insights into operator behavior, or be used to support operator training activities.

7 Conclusions

In this chapter we have introduced a high-level functional architecture for AI-assisted target identification systems. From this functional description, we have distilled six system design principles required for the optimal operation of such a system. While successful implementation of operator intervention is challenging, we believe the incorporation of these AI-assisted functions is critical for successful operation in safety critical domains, in particular when the task complexity is difficult to predict and occasionally very high.

References

1. Baddeley, A.D., Weiskrantz, L.E.: Attention: Selection, Awareness, and Control: A Tribute to Donald Broadbent. Clarendon Press/Oxford University Press, New York (1993)
2. Bailey, B.P., Konstan, J.A.: On the need for attention-aware systems: measuring effects of interruption on task performance, error rate, and affective state. Comput. Hum. Behav. **22**(4), 685–708 (2006)
3. Bartram, L., Ware, C., Calvert, T.: Moticons: detection, distraction and task. Int. J. Hum. Comput. Stud. **58**(5), 515–545 (2003)
4. Chiappe, D.L., Strybel, T.Z., Vu, K.P.L.: Mechanisms for the acquisition of situation awareness in situated agents. Theoret. Iss. Ergon. Sci. **13**(6), 625–647 (2012)
5. Cutrell, E., Czerwinski, M., Horvitz, E.: Notification, disruption, and memory: Effects of messaging interruptions on memory and performance. In: Proceedings of Interact. pp. 263–269 (2001)
6. Czerwinski, M., Cutrell, E., Horvitz, E.: Instant messaging: effects of relevance and timing. In: People and Computers XIV: Proceedings of HCI 2000, vol. 2, pp. 71–76 (2000)
7. Desimone, R., Duncan, J.: Neural mechanisms of selective visual attention. Ann. Rev. Neurosci. **18**(1), 193–222 (1995)
8. D'Mello, S., Olney, A., Williams, C., Hays, P.: Gaze tutor: a gaze-reactive intelligent tutoring system. Int. J. Hum-Comput. Stud. **70**(5), 377–398 (2012)
9. Dostal, J., Hinrichs, U., Kristensson, P.O., Quigley, A.: Spidereyes: designing attention-and proximity-aware collaborative interfaces for wall-sized displays. In: Proceedings of the 19th International Conference on Intelligent User Interfaces, pp. 143–152 (2014)
10. Dostal, J., Kristensson, P.O., Quigley, A.: Subtle gaze-dependent techniques for visualising display changes in multi-display environments. In: Proceedings of the International Conference on Intelligent User Interfaces, pp. 137–148 (2013)
11. Driver, J.: A selective review of selective attention research from the past century. Br. J. Psychol. **92**(1), 53–78 (2001)
12. Einstein, G.O., McDaniel, M.A.: Retrieval processes in prospective memory: theoretical approaches and some new empirical findings. In: Brandimonte, M., Einstein, G.O., McDaniel, M.A. (eds.) Prospective Memory: Theory and Applications, pp. 115–141. Lawrence Erlbaum Associates Publishers (1996)
13. Endsley, M.R.: Measurement of situation awareness in dynamic systems. Hum. Fact. **37**(1), 65–84 (1995)
14. Endsley, M.R.: Situation awareness misconceptions and misunderstandings. J. Cogn. Eng. Decis. Making **9**(1), 4–32 (2015)

15. Fogarty, J., Hudson, S.E., Atkeson, C.G., Avrahami, D., Forlizzi, J., Kiesler, S., Lee, J.C., Yang, J.: Predicting human interruptibility with sensors. ACM Trans. Comput.-Hum. Inter. **12**(1), 119–146 (2005)
16. Franconeri, S.L., Simons, D.J.: Moving and looming stimuli capture attention. Percep. Psychophys. **65**(7), 999–1010 (2003)
17. Garrido, J.E., Penichet, V.M., Lozano, M.D., Quigley, A., Kristensson, P.O.: Awtoolkit: attention-aware user interface widgets. In: Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces, pp. 9–16 (2014)
18. Gellersen, H.W., Schmidt, A., Beigl, M.: Multi-sensor context-awareness in mobile devices and smart artifacts. Mobile Netw. Appl. **7**(5), 341–351 (2002)
19. Healey, C.G., Booth, K.S., Enns, J.T.: High-speed visual estimation using preattentive processing. ACM Trans. Comput.-Hum. Inter. **3**(2), 107–135 (1996)
20. Imbert, J.P., Hodgetts, H.M., Parise, R., Vachon, F., Dehais, F., Tremblay, S.: Attentional costs and failures in air traffic control notifications. Ergonomics **57**(12), 1817–1832 (2014)
21. Iqbal, S.T., Bailey, B.P.: Effects of intelligent notification management on users and their tasks. In: Proceedings of the ACM Conference on Human Factors in Computing Systems, pp. 93–102 (2008)
22. Lavie, N., Hirst, A., De Fockert, J.W., Viding, E.: Load theory of selective attention and cognitive control. J. Experiment. Psychol. Gen.l **133**(3), 339 (2004)
23. Lavie, N., Tsal, Y.: Perceptual load as a major determinant of the locus of selection in visual attention. Percep. Psychophys. **56**(2), 183–197 (1994)
24. Lopez-Tovar, H., Charalambous, A., Dowell, J.: Managing smartphone interruptions through adaptive modes and modulation of notifications. In: Proceedings of the 20th International Conference on Intelligent User Interfaces, pp. 296–299 (2015)
25. Marquardt, N., Diaz-Marino, R., Boring, S., Greenberg, S.: The proximity toolkit: prototyping proxemic interactions in ubiquitous computing ecologies. In: Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, pp. 315–326 (2011)
26. Mehrotra, A., Musolesi, M., Hendley, R., Pejovic, V.: Designing content-driven intelligent notification mechanisms for mobile applications. In: Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pp. 813–824 (2015)
27. Morris, N.M., Rouse, W.B., Frey, P.R.: Adaptive aiding for symbiotic human-computer control: conceptual model and experimental approach. Tech. rep., Search Technology Inc., Norcross (1985)
28. Nicosia, L., Kristensson, P.: Inattention-management middleware for human-in-the-loop multi-display applications. In: Proceedings of the IEEE Workshop on Human-Centered Computational Sensing (2018)
29. Nicosia, M., Kristensson, P.O.: A conceptual design of an inattention management middleware with adaptive target saliency. In: 2020 IEEE Aerospace Conference, pp. 1–11 (2020)
30. Parasuraman, R., Davies, D.R.: Varieties of attention. Academic Pr (1984)
31. Parasuraman, R., Molloy, R., Singh, I.L.: Performance consequences of automation-induced'complacency'. Int. J. Aviat. Psychol. **3**(1), 1–23 (1993)
32. Parasuraman, R., Riley, V.: Humans and automation: use, misuse, disuse, abuse. Hum. fact. **39**(2), 230–253 (1997)
33. Posner, M.I.: Orienting of attention. Q. J. Experiment. Psychol. **32**(1), 3–25 (1980)
34. Posner, M.I.: Cumulative development of attentional theory. Am. Psychol. **37**(2), 168 (1982)

35. Roda, C., Nabeth, T.: Supporting attention in learning environments: attention support services, and information management. In: Creating New Learning Experiences on a Global Scale, pp. 277–291 (2007)
36. Roda, C., Thomas, J.: Attention aware systems: theories, applications, and research agenda. Comput. Hum. Behav. **22**(4), 557–587 (2006)
37. Rouse, W.B.: Adaptive aiding for human/computer control. Hum. Fact. **30**(4), 431–443 (1988)
38. Simons, D.J., Chabris, C.F.: Gorillas in our midst: sustained inattentional blindness for dynamic events. Perception **28**(9), 1059–1074 (1999)
39. Simons, D.J., Rensink, R.A.: Change blindness: past, present, and future. Trends Cogn. Sci. **9**(1), 16–20 (2005)
40. Vallières, B.R., Hodgetts, H.M., Vachon, F., Tremblay, S.: Supporting dynamic change detection: using the right tool for the task. Cogn. Res. Princ. Implicat. **1**(1), 1–20 (2016). https://doi.org/10.1186/s41235-016-0033-4
41. Wilson, M.D., Farrell, S., Visser, T.A., Loft, S.: Remembering to execute deferred tasks in simulated air traffic control: the impact of interruptions. J. Exp. Psychol. Appl. **24**(3), 360 (2018)
42. Yantis, S.: Goal-directed and stimulus-driven determinants of attentional control. Attent. Perform. **18**, 73–103 (2000)