



UPPSALA
UNIVERSITET

IT 18 032

Examensarbete 30 hp
Juni 2018

Visualising the unpredictable

Effective communication of process anomalies
requiring human intervention

Darren Lehane



UPPSALA
UNIVERSITET

**Teknisk- naturvetenskaplig fakultet
UTH-enheten**

Besöksadress:
Ångströmlaboratoriet
Lägerhyddsvägen 1
Hus 4, Plan 0

Postadress:
Box 536
751 21 Uppsala

Telefon:
018 – 471 30 03

Telefax:
018 – 471 30 00

Hemsida:
<http://www.teknat.uu.se/student>

Abstract

Visualising the unpredictable: Effective communication of process anomalies requiring human intervention

Darren Lehane

The role of data is of critical importance to our modern societies. As the total quantity of data being created is increasing exponentially, it is more important than ever to ensure that the optimal data visualisation methods are used to represent this data. One scenario in which this is particularly pertinent is for the identification of anomalous system states which require human intervention, such as in power plants and aviation control. As part of this study, a configurable and extendable software testbed for methodically evaluating the performance of different data visualisation types was developed. These tools were then used to investigate the performance of three different display types representing the same data set: a bar graph display, a configural coordinate display (CCD) designed using the principles of ecological interface design, and a configural integral display (CID) designed following the principles of poietic design. Both the total simultaneous quantity and the dynamicity of displays were examined. The bar graph display was observed to perform significantly worse than the other display types as the number of simultaneously visible displays increased. The CCD performed best overall, while the CID was not far behind in all cases except for those with the largest quantity of displays, indicating that both of these display types afford a certain level of gist perception. This research provides a basis for future studies aiming to evaluate the performance of alternative display types or redesigns to those evaluated herein and suggests some potential properties which could be of benefit to such redesigns.

Handledare: Mikael Laaksoharju
Ämnesgranskare: Mats Lind
Examinator: Anders Arweström Jansson
IT 18 032
Tryckt av: Reprocentralen ITC

Acknowledgements

I wish to express my gratitude to all the people who have contributed to the completion of this thesis. To Mikael Laaksoharju for the plentiful and insightful discussions, feedback, and advice. To Mats Lind for reviewing the thesis content, providing feedback, and helping with statistical analysis of the results. To Jeżyna Domańska for providing feedback on thesis content, dynamic displays, and helping to administer the experiment. To everyone who took the time to participate in the experiment and answer questions. And finally, to my family for their unwavering support and encouragement.

Darren Lehane

Table of contents

| | | |
|----------|---|-----------|
| 1 | Introduction..... | 1 |
| 1.1 | Research questions..... | 2 |
| 1.2 | Disposition..... | 2 |
| 2 | Background | 3 |
| 3 | Theory..... | 7 |
| 3.1 | Ecological interface design — configural coordinate display | 7 |
| 3.1.1 | <i>Direct perception</i> | <i>7</i> |
| 3.2 | Poietic design — configural integral display..... | 7 |
| 4 | Display types | 8 |
| 4.1 | Bar graph display | 8 |
| 4.2 | Configural coordinate display | 9 |
| 4.3 | Configural integral display | 10 |
| 5 | User study..... | 11 |
| 5.1 | Software testbed | 11 |
| 5.1.1 | <i>Technology.....</i> | <i>11</i> |
| 5.1.2 | <i>Data generation tool.....</i> | <i>11</i> |
| 5.1.3 | <i>Core library.....</i> | <i>12</i> |
| 5.1.4 | <i>Display type rendering add-ons.....</i> | <i>12</i> |
| 5.2 | Method | 12 |
| 5.2.1 | <i>Overall design decisions</i> | <i>12</i> |
| 5.2.2 | <i>Pilot tests.....</i> | <i>15</i> |
| 5.2.3 | <i>Definition of variable ranges.....</i> | <i>16</i> |
| 5.2.4 | <i>Test participants</i> | <i>17</i> |
| 5.2.5 | <i>Test location and equipment</i> | <i>18</i> |
| 5.2.6 | <i>Test design.....</i> | <i>18</i> |
| 5.2.7 | <i>Test procedure.....</i> | <i>19</i> |
| 5.2.8 | <i>Hypotheses.....</i> | <i>22</i> |
| 5.2.9 | <i>Real experiments.....</i> | <i>22</i> |
| 6 | Results | 23 |
| 6.1 | Participant feedback | 23 |
| 6.2 | Demographic data | 23 |
| 6.3 | Data analysis | 23 |
| 6.3.1 | <i>Exclusion of data points</i> | <i>23</i> |
| 6.4 | Accuracy | 24 |
| 6.5 | Response times | 24 |
| 7 | Discussion and conclusion | 27 |
| 7.1 | First research question..... | 27 |
| 7.2 | Second research question..... | 27 |
| 7.3 | Third research question..... | 27 |
| 7.4 | Fatigue | 28 |
| 8 | Future work..... | 29 |
| 8.1 | Potential redesigns | 29 |
| 8.2 | Dynamicity | 30 |
| 8.3 | Availability of software testbed and data | 30 |
| 9 | References | 31 |

List of figures

| | |
|--|----|
| Figure 1: Bar graph displays (states 1–4) | 8 |
| Figure 2: Configural coordinate displays (states 1–4) | 9 |
| Figure 3: Configural integral displays (states 1–4) | 10 |
| Figure 4: Decolourised bar graph displays | 13 |
| Figure 5: Decolourised configural coordinate displays | 13 |
| Figure 6: Decolourised configural integral displays | 13 |
| Figure 7: CCD showing two valid transitions (green) and two invalid transitions (red) | 14 |
| Figure 8: Calculating potential endpoint areas (green) for transition paths that will avoid quadrant 3 ... | 15 |
| Figure 9: Average response time by display type and number of displays | 26 |
| Figure 10: CIDs with inverted opacity of bars and bounding boxes | 29 |
| Figure 11: CIDs with gravitational corner quadrants illuminated in state-specific colours | 30 |
| Figure 12: CIDs with all corner quadrants illuminated in state-specific colours | 30 |

List of tables

| | |
|---|----|
| Table 1: Range of values for system variables from Coury et al. (1989) | 4 |
| Table 2: System variable mappings to states | 17 |
| Table 3: Overall test structure | 19 |
| Table 4: Known Unique State Performance Check — Accuracy by display type, layout, and dynamicity | 24 |
| Table 5: Known Unique State Performance Check — Response time in seconds by display type, layout, and dynamicity | 24 |
| Table 6: Repeated measures analysis of variance (ANOVA) | 25 |

I Introduction

The effective recognition of unique values and anomalies is a key use case for data visualisations. Take for example the task of identifying anomalies within a large set of integers. Such a task could prove to be practically impossible for the human brain beyond a certain quantity threshold. However, anomalies within the same set of integers visualised using an appropriate data visualisation method could become immediately identifiable by a human operator. Given this, the question then becomes: which data visualisation methods are most effective for which types of data sets?

In a world with ever-increasing sources and quantities of data, there is a constant need for novel and improved ways to visualise data in order to make it useful. After all, it is entirely irrelevant how much data one has available if it cannot be made interpretable and actionable in a reasonable manner.

Choosing the correct form of data visualisation is a critical component of designing information systems used to interact with large data sets. The choice of visualisation technique can be the difference between preventing a catastrophe and being just a moment too late.

While this is naturally the case for monitoring data generated by processes within computer systems, it is even more pertinent for the monitoring of real-world processes such as in weather systems, power plants, and air traffic control systems. As the edge case values for real-world processes often cannot be entirely predicted at the time of system design, this must be taken into consideration when designing systems to interact with such data. Furthermore, the specific values a human operator may need to identify can change from minute-to-minute, based on a plethora of external factors which may also be unforeseeable at the time of system design. As stated by Laaksoharju, Lind, & Jansson (2017), *“One critical scenario in which it is important to keep humans in the loop is when it is not possible to fully automate a process; when there is a risk that a decision needs to be made that requires knowledge – contextual, situated, synthetic, or empirical – that is not available at the time of system design”*.

Vicente & Rasmussen (1992) provide a way to classify events in complex systems based on the degree of their novelty to the system operators and system designers. Broadly defined, these classes are:

- **Familiar events:** Experienced frequently by system operators and anticipated by system designers. Due to training and experience, system operators are assumed to have the necessary skills to deal with these events sufficiently.
- **Unfamiliar but anticipated events:** Events which occur infrequently but are still accounted for by system designers. System operators will not be as experienced in dealing with such events, and therefore some responsibility will be delegated to system features designed to cope with these events, such as decision support systems or automation.
- **Unfamiliar and unanticipated events:** As with the previous class of events, system operators will not be experienced in dealing with these events. However, unlike the previous classes of events, these events will not have been anticipated by the system designers, and therefore no system-level features can be relied upon for delegation of responsibility.

While in an ideal world events which could be classified as both unfamiliar and unanticipated would never occur, this is beyond the realm of possibility when dealing with systems interacting with real-world processes. In Lipsett, Olmstead, & Stevens' (1989) paper, "*Balancing the roles of humans and machines in power plant control*", the authors state that "No matter how well designed and operated a plant may be, there will always be a remote chance that an obscure event will occur or that the plant will be damaged by equipment failure. For these largely unpredictable circumstances we must rely completely on human capabilities for problem identification and action".

This thesis aims to investigate and evaluate the efficiency of three different data visualisation display types for the purpose of variable anomaly detection by human system operators. The display types presented in this study are: a standard bar graph display, a 'configural coordinate display' (CCD) designed using the principles of ecological interface design (EID), and a custom box-like display type referred to as a 'configural integral display' (CID) designed using the principles of poietic design.

1.1 Research questions

This thesis aims to address the following research questions:

1. How does the performance of each display type compare for the purpose of anomaly detection after training?
2. Do any of the display types afford gist perception, and if so, under what circumstances?
3. What are the effects of dynamicity in data on the performance of the display types?

1.2 Disposition

Following this *Introduction* chapter, the *Background* chapter provides an overview of previously conducted research in this field and the motivation for this study.

The *Theory* chapter then describes two design philosophies which have dictated the design of two of the display types evaluated — ecological interface design and poietic design.

The next chapter, *Display types*, gives an overview of the three different display types which were evaluated in this study.

The following *User study* chapter consists of two subchapters. The *Software testbed* subchapter provides an overview of the bespoke testbed which was developed to facilitate the execution of our user testing experiments and represented a significant portion of the overall study. This is followed by a *Method* subchapter, which chronicles the extensive user testing methods and procedures.

The *Results* chapter then presents the results of the study in a summarised format.

In the *Discussion and conclusion* chapter which follows, the results are interpreted in relation to the research questions and hypotheses.

The thesis finishes with a *Future work* chapter which provides direction for other researchers wishing to replicate or extend this study. Additional hypothetical design ideas are presented for consideration.

2 Background

Data has become one of the most important elements of our modern societies. Data is used to understand ourselves and the world around us, and to direct decision-making in ways never before possible. Increasing quantities of technologies and systems are becoming networked and interconnected, and thus, more data than ever is being created with systems being interoperated at an accelerating pace. A 2013 study found that 90% of the world's data had been generated in the previous two years alone (SINTEF, 2013).

While there is virtually no limit to the number of data visualisation use cases that could be examined in this study, its primary focus will be on the representation of data for use by system operators monitoring real-world processes.

We begin by turning to Rasmussen (1986) who discusses the design of control and safety systems, and in particular, those used by system operators at industrial process plants. As stated by Rasmussen, and reiterated by Coury, Boulette, & Smith (1989), a system operator's task is composed of two fundamental components: *"identifying the state of the system"* and *"selecting the appropriate course of action"*. Referencing another paper (Coury & Pietras, 1989) published in the same year, Coury et al. (1989) found that *"the information necessary for supervising a simulated fluid-processing plant operating in a normal mode was significantly different from the information required to control a failing process plant"* and *"the way in which operators were presented information and data had a significant effect on their ability to control the system and accommodate the disruptive effect of failures"*. Coury et al. (1989) go on to state that operators were observed to employ significantly different information-gathering strategies when *"data and information were displayed in suboptimal form"*.

It is often critical that human operators understand the relationships between visual data representations shown on their monitors and the underlying system variables which can be subjected to various levels of abstraction in an effort to provide clarity (de Kleer & Brown, 1983; Rasmussen, 1986).

With the rapid advancement and general availability of artificial intelligence, automated control systems will delegate an increasing amount of decision-assisting and decision-making tasks to algorithms rather than to human operators. As Nickerson (1969) aptly predicted, *"the need for the future is not so much computer oriented people as for people oriented computers"*. This statement is truer than ever today, at a time when many people are no longer being assisted by computers to perform primarily human jobs, but rather, people are now assisting computers to perform primarily computational jobs.

Henneman & Rouse (1987) discuss how the rise in automation will fundamentally change the way in which human operators interact with systems. No longer will humans be in continuous control of the system at all times, but instead, control will only be transferred to human operators during abnormal events like system failures. Hence, the authors state that the skills required of human operators are shifting from psychomotor to problem solving (Wickens & Hollands, 1984).

In a paper titled *"Effect of Uncertainty and Diagnosticity on Classification of Multidimensional Data with Integral and Separable Displays of System Status"*, Coury

et al. (1989) investigated the role which uncertainty plays in the classification of system states. The authors conducted an experiment to examine the efficiency of three different data display types for representing numerical data points, for the purpose of system state classification. The chosen display types were a configural display (polar graph), an alphanumeric display, and a bar graph display. The experiment focused on the classification of individual displays into specific system states based on the relative comparison of four independent integer variables. The evaluation focused on both the participants' speed of classification, as well as the accuracy of their decisions. The results of this study demonstrated that the bar graph display was both the fastest and the most accurate method of data visualisation for tasks concerned with the integration of distinct variables.

| System State | System Variable | | | |
|--------------|-----------------|-------|--------|--------|
| | Q | M | B | H |
| 1 | 25–51 | 49–75 | 0–26 | 74–100 |
| 2 | 25–51 | 49–75 | 74–100 | 0–26 |
| 3 | 49–75 | 25–51 | 0–26 | 74–100 |
| 4 | 49–75 | 25–51 | 74–100 | 0–26 |

Table 1: Range of values for system variables from Coury et al. (1989)

In 2015, Holt, Bennett, & Flach (2015) replicated Coury et al.'s (1989) study, adding a new display type which they designed specifically for the given classification task, named the 'configural coordinate display' (CCD). This new display type was designed following the principles of ecological interface design (EID), which is described in the *Theory* chapter below. Their results indicated that the CCD was both faster and provided greater accuracy than all other display types for the specified classification task. Given that this new display type was designed specifically for the classification task featured in this experiment, this outcome would not be entirely unexpected.

As part of a 2016 thesis, Löfvenberg (2016) devised a replication study which evaluated the most performant display types from the previous two studies — the bar graph display and the CCD — along with a newly introduced 'configural integral display' (CID) designed by Mikael Laaksoharju using the principles of poietic design. The aim of this poietic redesign was *"to combine the configural properties of the CCD while retaining generalizability by adding integral (object-forming) properties that correspond in a more unbiased way to the data"* (Laaksoharju et al., 2017). The study found that while the CCD performed very well in the task for which it was specifically designed, it did not perform as well in more generalised use cases. In contrast, the poietic display was found to work well across a number of independent tasks, although never matching the performance of the CCD for the originally specified classification task.

While the aforementioned previous studies focused on the identification of a specific system state represented by a single display, the research carried out for this thesis acts as a continuation and focuses on the performant identification of single anomalous system states within larger sets of displays. Hence, a key aspect of this study is that it presents the human operator with a set of similar displays

simultaneously and evaluates how efficiently the operator is able to identify the single display representing a system state *unique* to all the other displays.

While much research has been carried out in the area of state determination for decision problems (Coury et al., 1989), the area of anomaly detection within combined sets of such system states has not seen as much prior research.

For this study, a system state is defined as a single integer between 1 and 4. It is important to note, however, that this system state integer is never directly communicated to the system operator. The system state is ultimately determined by the relative differences within two pairs of integer variables (four variables in total), all ranging from 0 to 100. These four variables are the only values directly communicated to the system operator. The variable ranges used were those first put forth by Coury et al. (1989) (shown in Table 1), modified slightly to better fit our use case of unique anomaly detection. The reasoning behind the adjustment of these ranges is discussed further in the *Method* chapter below.

As previously mentioned, it is critical that the system operator has persistent visibility into the underlying four variables constituting each display, even if the chosen display type substantially augments or enhances the variables in order to create the final visual data representation.

Coury et al. (1989) states that *“there is substantial evidence to suggest that object displays are superior to alphanumeric displays in many applications in which identification of system state requires integrating data from a number of information sources”*. Object displays, as defined by Wickens & Hollands (1984), are displays in which *“multiple information sources are encoded as the stimulus dimensions of a single object”*. This is relevant in the case of our own study, as data from up to four different information sources could be present for each individual display, since each display represents four distinct variables. However, Coury et al. go on to state that *“the superiority of an object display may diminish when the state of the system is uncertain and identification of system state requires that the display be decomposed into its individual system variable values”*.

In Coury et al.’s configural coordinate display type, the values of the four variables create lines which act together to compose a four-sided polygon shape. The assumption is that the operator is then able to identify which system state the display is in based on the specific shape of this polygon, without needing to individually inspect all four variables. Coury et al. claim that this *“allows the operator to process the data as a spatial code”*. This is a manifestation of the ‘emergent features’ concept presented by Pomerantz, Pristach, & Carson (1989). As described by Holt et al. (2015), emergent features are *“higher order visual properties, often hierarchically nested, that are produced by the interaction of lower level graphical elements”*.

Salience and perceptual cues also play a major role in the visual recognition of system states represented by object displays. While all display types examined in our study could be said to feature a certain level of salience, some are objectively more salient than others.

The aim of the study presented in this thesis was not to examine state identification of single displays in isolation, but rather, to examine larger sets of displays and the use of serial and parallel visual search patterns for the purpose of anomaly identification.

Treisman and others (Nakayama & Silverman, 1986; Narbutas, Lin, Kristan, & Heinke, 2017; Treisman & Gelade, 1980) have conducted much research in the field of serial versus parallel visual search, and this can be used as a solid basis for designing new display types intended for use in anomaly identification. Treisman & Gelade (1980) introduced the ‘feature-integration’ theory of attention in their paper, which found that when distinguishing individual displays in a set of equivalent display types, those requiring conjunction of more than a single separable feature to determine state required the operator to conduct a serial visual search across all displays. Consequently, the total visual search time increased linearly as the total number of displays increased. Conversely, display types whose state could be determined from a single separable feature enabled the operator to conduct a visual search of all displays in parallel, and thus the total visual search time remained stable, regardless of the number of distractor displays.

Visual attention can be considered analogous to a ‘spotlight’ or ‘zoom lens’, in that it can be applied with great detail to a localised focal area, or with diminishing detail to a larger diffuse area (Eriksen & Hoffman, 1972; Treisman & Gelade, 1980). Goodhew, Shen, & Edwards (2016) further elaborated on this concept of attentional spotlight size and examined its effects on both spatial acuity (the ability to distinguish fine details in a visual scene) and temporal acuity (the ability to distinguish visual events in time). They found that spatial acuity was enhanced in the case of more focalised attentional spotlights, but that this same enhancement was not observed in regard to temporal acuity.

The second factor that we aimed to investigate was how adding dynamicity to the data set would affect classification performance and results, particularly when compared to the static displays. While there are endless forms of dynamicity that could theoretically be applied to the display types examined, it was decided to limit the scope of this evaluation to interval-based, involatile transitions which exhibited mostly uniform transition speed patterns.

As dynamicity is likely to be a key element of real-world displays used to monitor interactive processes, we strived to incorporate the evaluation of dynamicity into this study, although there is not as much prior research for dynamic displays as for static displays. One related study is that by Foley & McMath (1986) which discusses the development of a process visualisation system intended to enable non-programmers to design productivity tools to monitor dynamic real-world processes such as those in manufacturing plants, power plants, and refineries.

In a paper focusing on human problem solving in dynamic environments, Henneman & Rouse (1987) point out that *“if the main role of the human operator in a large dynamic system is to diagnose failures, an important issue is whether or not humans can change their control strategies to adapt to changes in the quality (or reliability) of individual system components”*.

3 Theory

As two of the display types examined in this study — the CCD and the CID — have originated from specific design methodologies, it is essential to provide an overview of those methodologies so the rationales for the designs can be understood. The third display type — the bar graph display — is perhaps one of the most well-known display types, and is included because of its high performance demonstrated in Coury et al.'s (1989) study and because it serves as a standard baseline visualisation type for a fixed set of integers, to which other display types can be compared.

3.1 Ecological interface design — configural coordinate display

Ecological interface design (EID) is a design framework first introduced by Rasmussen & Vicente (1989). The primary focus of EID is on the work environment, rather than on the users, their tasks, or their knowledge. The EID framework has been tested in many critically important and complex work domains such as in chemical process plants, power plants, and aviation control (Anokhin, Ivkin, & Dorokhov, 2018; Burns et al., 2008; Ho & Burns, 2003; Jamieson, 2007). As technological systems become more advanced, the likelihood of unanticipated events occurring in such work domains increases significantly (Vicente, 1999). EID does not intend to produce simple and trivialised displays (Borst, Flach, & Ellerbroek, 2015), but rather, its intention is to provide system operators with an effective method of understanding complex system values and constraints, allowing them to effectively deal with both anticipated and unanticipated events. EID asserts that to understand human behaviour, we must first understand the environment in which people are working.

3.1.1 Direct perception

Direct perception is a key principle of ecological interface design. Holt et al. state that when dealing with analogical visual displays, the principle of direct perception refers to the quality of two specific mappings. The quality of the first mapping, referred to as 'content mapping', can be established by asking *"have all properties of the work domain that are necessary for effective control been incorporated into the set of visual displays?"* (Holt et al., 2015). The quality of the second mapping, referred to as 'form mapping', can be established by asking *"have the perceptual properties of these displays been designed so that the practitioner can readily pick up the possibilities for action through consistent spatio-temporal patterns in the displays?"* (Holt et al., 2015).

3.2 Poietic design — configural integral display

Poietic design is a design philosophy originally proposed by Mikael Laaksoharju. Löfvenberg (2016) states that the purpose of poietic design is to *"make an artefact's features perceptible in order to allow its users to develop new intuitions about its functions and its possible effects on the world"*. Essentially, poietic design advocates design that does not entirely abstract the inner workings of a system away from the user, but instead enables the user to gain an understanding of the underlying system processes naturally and intuitively.

4 Display types

4.1 Bar graph display

The first of the three display types in our study, the bar graph display, was chosen in order to provide a baseline to which other more intricate and refined display types could be compared.

Upon first glance, the bar graph could be considered an entirely separable display type. Each of the bars represents just one variable, and each of them is spatially separated from the others. Due to this variable separability, bar graphs have been advocated for use as separable displays by many researchers (Carswell & Wickens, 1987).

Coury et al. (1989) assert that the use of verbal coding can be applied to bar graph displays, giving the example of “*high-low-low-high*” to represent a bar graph in which the outer two bars are taller than the inner two bars. Such verbal coding strategies proved highly relevant to our study and could in fact be further enhanced to match our use case. Coury et al. go on to address the possibility that the bar graph display “*can possess objectlike properties*” and “*the heights of the bars produce contours unique to a system state*”. Indeed, Sanderson, Flach, Buttigieg, & Casey (1989) observed that the emergent feature of inferred linearity, based on the relative height difference between the bars in a bar graph, played a role in the interpretation of these graphs. Two of the other studies mentioned also reference the configural potential of the bar graph display (Buttigieg, Sanderson, & Flach, 1988; Coury & Purcell, 1988).

The potential use of verbal coding strategies like those mentioned by Coury et al. (1989) was of particular interest to our own study for the purpose of providing participants with useful visual search strategy suggestions. Furthermore, as the state classification process in our study was based on the relative comparison of two pairs of variable values, and not the individual values themselves, verbal coding could be simplified to just a single phrase for each pair. Thus, Coury et al.’s example of ‘high-low-low-high’ could be rewritten as ‘slope down, slope up’ for the purpose of our system classification task. Indeed, to simplify this further, the examples of ‘slope down, slope up’ and ‘slope up, slope down’ could instead be rewritten as the higher order emergent features of a ‘valley’ and a ‘peak’ respectively.

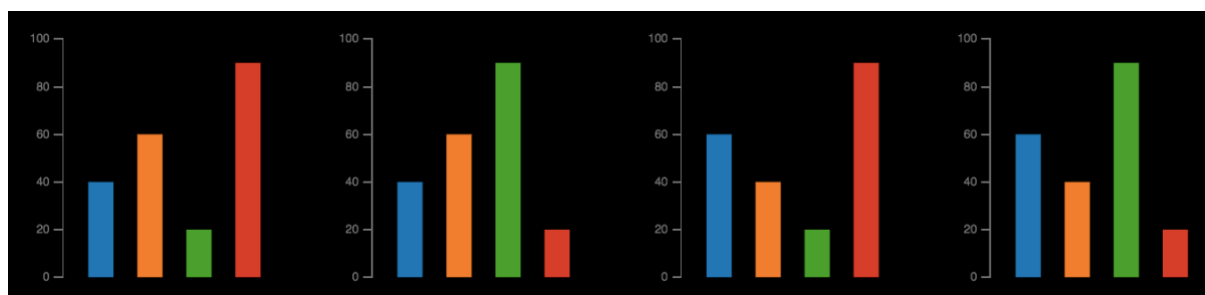


Figure 1: Bar graph displays (states 1–4)

4.2 Configural coordinate display

The second display type evaluated was the configural coordinate display (CCD), originally designed by Holt et al. (2015) following the framework of ecological interface design.

This display type was designed to allow the operator to perceive the system state more directly. Instead of presenting the operator with the four system variables individually, the CCD presents a single calculated point based on the relative value differences within the two pairs of system variables. The x-axis value for this point is determined by subtracting the H variable from the B variable, while the y-axis value is determined by subtracting the Q variable from the M variable. The actual system state can then be easily determined by identifying which quadrant this calculated point is located in. As stated by Holt et al., *“the spatial location of a single point in the coordinate grid provides a salient emergent feature that accurately reflects the rules for state identification and requires little or no mental calculations, nor any sort of fine perceptual discriminations”*.

A limitation highlighted by Holt et al. in their original study was the fact that the ‘content mapping’ aspect of direct perception was not effectively realised with this display type — i.e. the actual values of the four individual variables are made unintelligible, as only the calculated differences between the values are presented to the operator. Holt et al. proposed a redesign of this display which would provide visibility into the underlying four system variables. In essence, their redesign added a two-value bar graph for each pair of variables outside the relevant axis for each pair. As described by Holt et al., *“The locations of the axes for both sets of bar graphs are dynamic. The tops of the H and Q bar graphs are always aligned with the origin of the appropriate axis in the coordinate display. The values of these variables therefore determine where the origin of each bar graph axis will be located”*. This redesigned version of the display was chosen for evaluation in the study by Löfvenberg (2016) and was also chosen for evaluation in our study due to its similar level of separable features to the CCD and bar graph display.

There is no doubt that the CCD performed exceptionally well in the task for which it was originally designed, however, we became curious if it would perform as well in a task largely dependent on gist perception.

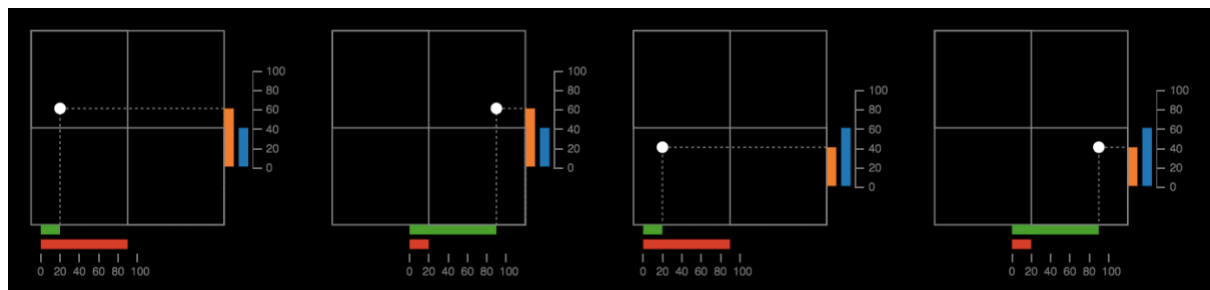


Figure 2: Configural coordinate displays (states 1–4)

4.3 Configural integral display

The third display type was the box-like configural integral display (CID) originally designed by Mikael Laaksoharju and presented by Löfvenberg (2016). This display was a poietic redesign of the CCD described above. It was chosen as we wished to further explore the possible use cases for poietic design and it was hypothesised that it may perform better than the other display types for establishing gestalt and facilitating gist perception. As stated by Löfvenberg, “*poietic design not does presuppose a goal, but rather strives after bringing forth the essence of the artefact*”.

This display consists of four bars representing the system variable values, just as in the other display types. In this design, all four variables begin from the same centre origin and extend outwards along the right side of their respective axis lines. A translucent white bounding rectangle is created around the extremities of the four bars. The system state can then be determined by judging the quadrant in which this rectangle's centre of gravity resides.

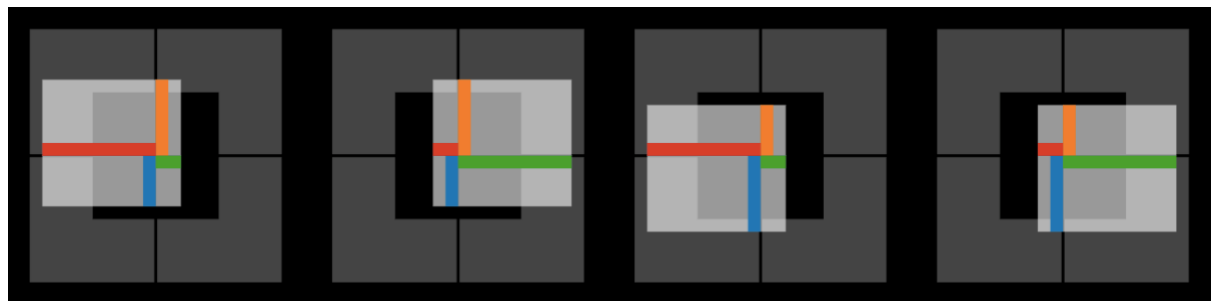


Figure 3: Configural integral displays (states 1–4)

5 User study

5.1 Software testbed

To test the hypotheses and to carry out user testing experiments, we developed a software testbed for evaluating the performance of different data visualisation display types for a shared data set in multi-user testing environments. This design and engineering effort represented a significant portion of the thesis and its product is one of the key contributions provided.

We investigated existing solutions aimed to serve a similar purpose but felt that none of them fully satisfied the needs of this study. Furthermore, our technical skill set and experience with the relevant web technologies meant that we could iterate quickly on new versions of our bespoke testbed, and comfortably tailor it to fit our specific needs.

The testbed itself and a practical guide will be available at thesis.darrenlehane.com.

5.1.1 Technology

The software developed for the user testing sessions was a custom web application written in JavaScript, utilising the D3.js data visualisation library for display rendering. The software was hosted remotely and accessed by all participants via a web browser. Participants' actions were monitored and recorded to a PostgreSQL database on the remote server as they completed each screen of the experiment.

While the application is designed to work in any web browser, we found that Google Chrome had the best SVG (Scalable Vector Graphics) rendering support for our particular use cases.

5.1.2 Data generation tool

The data generation tool is a script capable of recursively generating data sets consisting of both static and dynamic data points as JSON (JavaScript Object Notation) payloads. It was originally designed to be used via a browser-based configuration dashboard, however, as the complexity and intricacy of potential data sets increased, this became cumbersome to use and slow to iterate upon. The new version is interacted with via a JavaScript configuration file in which users define their desired data set configuration. The testbed provides the capability to generate informational screens, sequential display screens, and unique state display screens. Additional parameters are also available when generating display screens. These include the ability to generate static or dynamic display data, varying quantities of displays per screen, specific quantities of displays in each system state on each screen, and the ability to generate multiple screens from a single definition, reducing the amount of configuration lines required for generating larger data sets.

Another key element of the data generation tool is the ability to define 'shared data' which is intended to be randomly shuffled and reused in multiple phases of the same test. This shuffling can be defined on two levels. The first level is that the order of all displays within a given screen can be randomly shuffled while keeping all other variables constant. The second level of shuffling applies to the screens themselves. Screens can be grouped into shuffle-ready sets, and their order will be randomly reassigned for each test phase. The combination of these two random shuffle functions means that test participants can be shown the same shared data set multiple times

without being able to discern that they have already seen the same displays before. There are multiple benefits to this shuffling practice.

The first benefit is that a single shared data set can be defined just once and reused for multiple test phases, drastically reducing the amount of display configuration definitions required.

The second benefit is that it ensures all phases are of approximately equal difficulty for participants. No phases will have easier or more difficult unique state determinations, as they will all be comprised of the exact same displays.

The third benefit is that it allows researchers to compare participants' performance when identifying the exact same unique state display from the same set of displays multiple times. This can be used to measure training effects (whether a user becomes more performant at identifying the same unique display after practice) and also to measure whether the spatial placement of the displays has any bearing on performance.

5.1.3 Core library

The core library is responsible for fetching and parsing the JSON file generated by the data generation tool, invoking the appropriate display rendering add-on, handling participants' test answers and other on-screen events, managing dynamic display rendering loops, transmitting the recorded answer data to the server, and more associated functions.

Upon completing a screen, participants pressed the *Enter* key to proceed. Participants could only proceed to the next screen when their answers for the current screen had been successfully saved to the remote database. As this was always only a matter of milliseconds, it had no bearing on the participants' interactions with the software. This restriction was simply enforced to eliminate the possibility of any data loss due to unexpected network latency or connectivity issues.

5.1.4 Display type rendering add-ons

Three individual display rendering add-ons were developed, one for each of the display types evaluated in this study. This add-on-based system was implemented to add extensibility to the testbed, allowing other researchers to develop their own add-ons for any new display types they may wish to evaluate.

5.2 Method

A user testing experiment was carried out to investigate how each display type performed under otherwise identical conditions. A randomised data set was generated and stored in a single JSON file. This JSON file served as the data source for every participant's testing session, regardless of which display type they were assigned to.

5.2.1 Overall design decisions

Colour

The original display prototypes created for our study all featured uniquely coloured variable representation bars, similar to those presented by Löfvenberg (2016).

Following preliminary testing sessions, it became apparent that the colouration of the variables bars could potentially unfairly favour display types in which these bars were

larger. In our case this applied to the bar graph display, as the variables bars for this display type occupy approximately four times more spatial area than for the CCD or the CID.

As our primary objectives were to evaluate anomaly identification and gist perception within larger sets of displays, we believed that colour was too significant of a separable dimension to be included. To remove this potential biasing, and to maximise visual contrast, all variable bar colours were substituted for pure white.

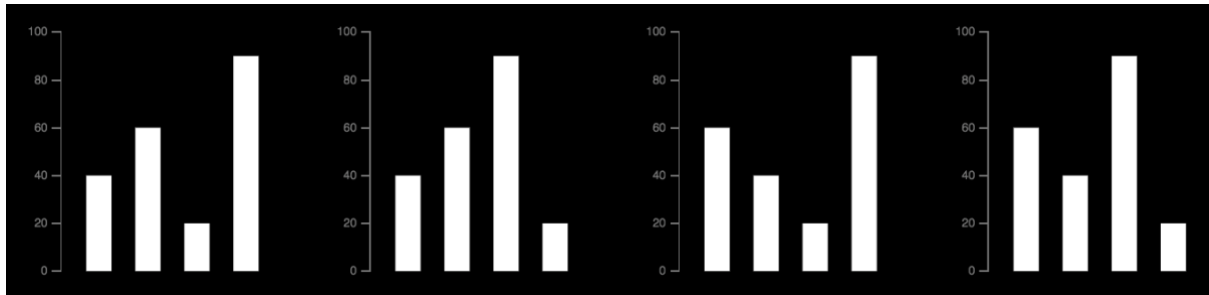


Figure 4: Decolourised bar graph displays

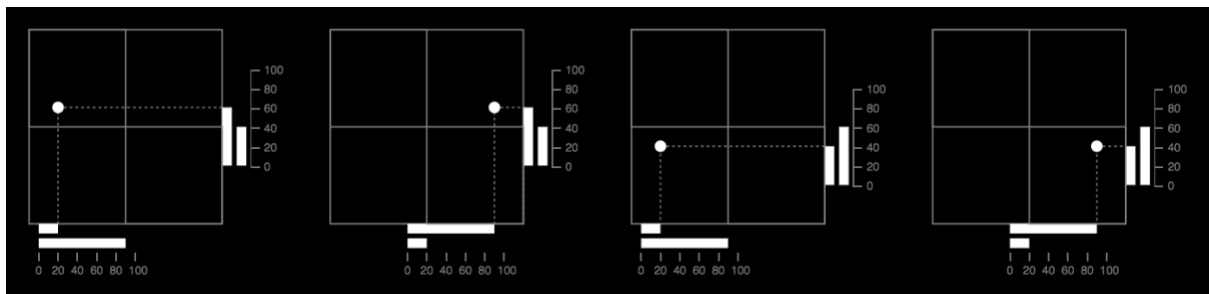


Figure 5: Decolourised configural coordinate displays

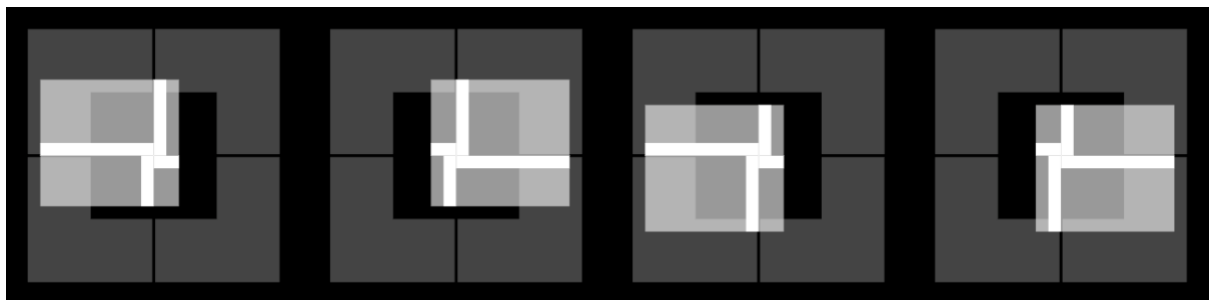


Figure 6: Decolourised configural integral displays

Dynamicity

The element of dynamicity played a significant role in our study. While all previous studies focused on the evaluation of static displays, we were intrigued to investigate the potential implications dynamicity may have on these display types.

In the bar graph display, only the bars themselves exhibited movement and thus did not cause any knock-on visual effects. In the case of the CID, the resizing of the bars caused the bounding rectangle to reposition and resize simultaneously. However, the greatest knock-on effect was exhibited by the CCD — as the bars in this display transitioned between their different values, it caused not only the dashed axis lines and calculated point to reposition, but it also caused the outer bar graph axes to

dynamically reposition themselves accordingly, contributing to a significant amount of visual momentum.

Preventing accidental unique states

A significant consideration that needed to be made when designing the dynamic display transition patterns was that *accidental* unique states could temporarily appear depending on the transition patterns of other displays. The easiest way to visualise this potential problem is with the use of a CCD.

Take for example a situation in which a participant is required to identify a dynamic display in state 3. Before the intended unique state display enters state 3, all displays will be randomly transitioning between states 1, 2, and 4. In other words, the calculated dot will be transitioning between quadrants 1, 2, and 4. While the potential transition paths between quadrants 1 and 2 present no possible problems, the potential transition paths between quadrants 1 and 4 are not as problem-free. A dot transitioning between quadrants 1 and 4 will either pass through the bottom-left corner of quadrant 2, or the top-right corner of quadrant 3, depending on the specific randomly-generated variable values. The problem arises when the dot passes through the corner of quadrant 3, as this could cause the participant to erroneously believe that this is the state 3 display they are required to identify, resulting in them choosing this display as their answer. Even though the participant would in fact be correct that this chart momentarily appeared in state 3, this would still be considered an incorrect answer and disrupt our data analysis procedures.

The aforementioned problem is illustrated in the figure below. The two transition paths shown in green are valid in this case as they do not pass through quadrant 3, while the two transitions paths shown in red are invalid in this case as they pass through quadrant 3.

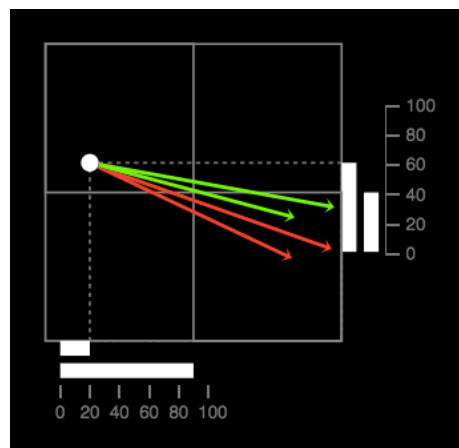


Figure 7: CCD showing two valid transitions (green) and two invalid transitions (red)

This edge case was prevented by developing a much more advanced dynamic data generation algorithm. To provide a simplified explanation of the algorithm, consider the state 1 to state 4 CCD transition mentioned above. To ensure this randomly-defined linear transition does not accidentally pass through quadrant 3 on its way to an endpoint in quadrant 4, we use a linear function to calculate the path of the dot through the origin (0,0) and into quadrant 4. This line is then used to further delimit the potential area in which a random state 4 point can be placed so that the dot will

not pass through quadrant 3. Without this delimitation, the potential area in which a state 4 point can be placed based on the system variable ranges is bounded by (50,0), (100,0), (50,-50), and (100,-50) (the yellow boxes shown in the figure below). In effect, this algorithm is slicing off the part of the yellow box which the dot cannot linearly transition into without passing through the third quadrant. In this particular case, as the dot's starting position becomes lower (closer to $y=0$) and more leftward (closer to $x=-100$), the potential area it can transition into (shown in green in the figure below) in quadrant 4 becomes more limited.

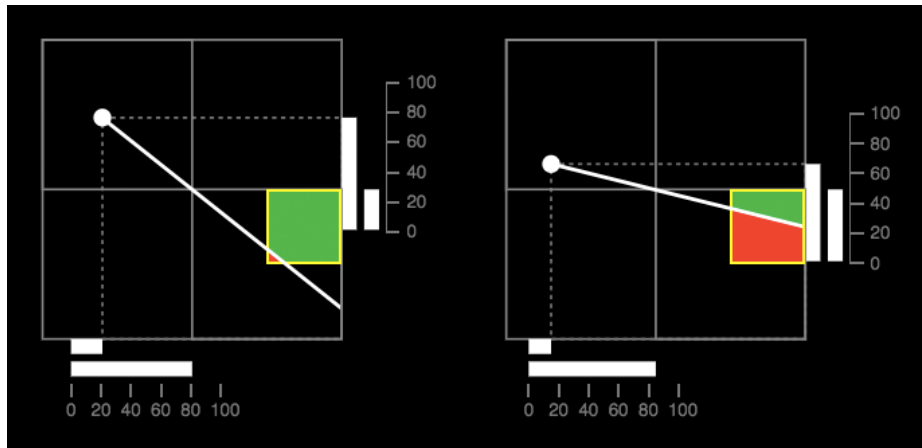


Figure 8: Calculating potential endpoint areas (green) for transition paths that will avoid quadrant 3

5.2.2 Pilot tests

Two pilot testing sessions were conducted with three participants in total prior to the real experiment. These pilot tests provided immensely valuable insights about how the application would function in a real test environment and highlighted key enhancements which could be made to the software testbed.

First pilot test

The first pilot testing session was conducted with one test participant who completed the test with the CID. The main takeaways from this first pilot test were regarding the animation sequences of the dynamic display screens, and how the screen configurations influenced the total test duration.

Prior to this pilot test, on dynamic display screens, all of the non-unique displays transitioned between different points within a single non-unique state which was assigned to that display. Therefore, the eventual unique display was the only one which would transition from one state to another. The result of this was that the accelerated movement present in the eventual unique display during transition to the new state could be used as an obvious indicator this display was transitioning into a different, and therefore unique, state. As our primary goal was not to measure how well visual movement highlighted the unique displays, we sought to alleviate this issue, either by making the movement of a transition into a unique state less noticeable, or by increasing the overall visual movement in the non-unique displays. It was decided to choose the latter option, as we believed that this best reflected how the variables may behave in a real-world environment. In the next iteration of the application, the non-unique displays randomly transitioned between the three non-unique states.

The takeaway regarding total test duration was that the entire test could be completed by a first-time user much faster than we anticipated. As a result, for the subsequent testing sessions, we added additional screen configurations, eventually totalling 76 screens which contained displays (outlined in Table 3).

Second pilot tests

The second pilot testing session was conducted with two participants — one completing the test using the bar graph display, and the other completing the test using the CCD.

The primary goal of this second pilot testing session was to test the changes which were implemented based on what we learned from the first pilot testing session, as well as to compare the differences in total test durations between all display types.

A key takeaway from the second pilot testing sessions related to which keys the participants should make use of while performing the test. Originally, the number of input methods was limited as much as practically possible. The interaction paradigm was as follows: The user would press the space bar when they had identified the unique display. They would then click on the overlay of the display they had identified. Finally, they would press the space bar once again to proceed to the next page when ready. We believed that this had the added benefit of ergonomically preparing the user for the next screen before it even loaded, as their hand was already prepared to press the space bar again when identifying the next unique display.

However, throughout the pilot testing sessions, we observed a number of accidental space bar presses occurring immediately after a new screen loaded. These accidental presses primarily occurred when the participant pressed the space bar to continue to the next screen, but subsequently rested their hand too heavily on the space bar in an effort to be prepared to press it after identifying the next unique display.

To counter the possibility of this occurring in the real user testing experiment, a decision was made to designate the space bar as a single function key for identifying a unique display. The key to be pressed to proceed to the next screen was changed to *Enter*, as it is acceptably distant from the space bar, and as it works conceptually as a 'next' button due to its placement on the right side of the keyboard.

5.2.3 Definition of variable ranges

A principle motivation for Coury et al.'s (1989) variable definition ranges was the concept of 'uncertainty'. That is, in some edge cases, the operator viewing a display representing a set of four variables could not be certain about which state a display was in fact in, due to slight overlaps in the variable ranges.

The first pilot tests for this study were conducted using the exact variable ranges defined by Coury et al. When analysing the results from the pilot tests, certain screens were found to exhibit an abnormally high error rate. Upon investigation, we discovered that these screens contained displays which were in one of these aforementioned 'uncertain' edge cases.

After much reflection and discussion, our belief was that these 'uncertain' edge cases in the variable ranges are not in fact uncertain per se, but rather can be more accurately described as 'ambiguous'. The operator is actually entirely certain about the

state of the display, and the fact is that the display is in two distinct states simultaneously, therefore creating ambiguity.

While this configuration worked as intended for Coury et al.'s study which focused on the classification of individual displays, it presented fundamental problems for both our testing sessions and statistical analysis.

Take for example the following set of system variable values: $Q: 51, M: 49, B: 20, H: 85$. By Coury et al.'s definition, such a set of variables would constitute a system state of both 1 and 3 simultaneously. Coury et al. describe such uncertain sets of variables as having low 'diagnostic value'.

An issue arose in our pilot testing sessions when the unique state was randomly assigned to be system state 1. All the other displays' system states were randomly generated to be in one of the non-unique states (2, 3, or 4), and there were at least two occurrences of each, to ensure that they were all in fact not unique. Two displays were assigned a system state of 3, however, one of these sets of system variables was identical to the one mentioned in the previous paragraph which exhibits an edge case scenario. In practice, this meant that some operators would perceive two of the displays as being in state 1 (and therefore state 3 would be the inferred unique state), while other operators would perceive two of the displays as being in state 3 (and therefore state 1 would be inferred as the unique state). This presented problems for data analysis as it was necessary for there to be only one definitive unique state in order to compare all participants' performances accurately.

To remedy this, we adjusted the variable ranges slightly to reduce the possibility of overlap edge cases in our study. The aim was to modify the original variables as little as possible to maintain parity with the previous study. Our new variables ranges were as follows:

| System State | System Variable | | | |
|--------------|-----------------|-------|--------|--------|
| | Q | M | B | H |
| 1 | 24–50 | 50–76 | 0–26 | 74–100 |
| 2 | 24–50 | 50–76 | 74–100 | 0–26 |
| 3 | 50–76 | 24–50 | 0–26 | 74–100 |
| 4 | 50–76 | 24–50 | 74–100 | 0–26 |

Table 2: System variable mappings to states

5.2.4 Test participants

Recruitment

It was believed that having 10 participants testing each of the three display types would provide substantial statistical reliability for the purpose of data analysis. To provide a safety buffer in case of participants dropping out or in case of other problems, we decided to increase this target to 11 people per display type. Exactly 33 participants in total were recruited in the days prior to the testing sessions. Approximately half of these participants were recruited online via groups of Uppsala

University students, and the other half were recruited in person at Campus Polacksbacken of Uppsala University. Participants were informed that they would receive cinema ticket vouchers as a reward for their voluntary participation.

Participants signed up for the testing sessions via an online form where they provided their name, email address, and preferred time slot. 15 people participated in the first time slot, 17 people participated in the second time slot, and one person was unable to attend.

Demographics

Various demographic information was collected anonymously from test participants, including their genders, ages, countries, and study programmes. The gender breakdown was 19 male and 13 female. The youngest test participant was 19 years old, the oldest was 34 years old, and the average participant age was 28 years old. Participants from nine different nationalities were present. 16 participants had 11 or more years of formal English language education, 15 participants had between 6 and 10 years, and 1 participant had 5 years or fewer. 20 participants came from a science, technology, engineering, or mathematics (STEM) academic background, while 12 participants came from a social sciences or interdisciplinary academic background.

5.2.5 Test location and equipment

User testing sessions were conducted in a computer laboratory at the Information Technology Centre of Campus Polacksbacken at Uppsala University. The laboratory consisted of 20 computers in total — six along both side walls, and four along each side of a centre row.

All testing sessions were carried out using a Google Chrome kiosk extension on the Microsoft Windows 10 operating system. HP EliteDisplay E272q 27-inch monitors were used at native Quad HD (2560x1440) resolution and 60Hz refresh rate. All monitors were set to 100% contrast, 100% brightness, and a neutral colour temperature.

5.2.6 Test design

Three different display types were evaluated during this study — a bar graph display, a configural coordinate display, and configural integral display. Both static and dynamic versions of each display type were evaluated, as well as three different display layouts: an 8-display layout (2 rows, 4 columns), a 32-display layout (4 rows, 8 columns), and a 72-display layout (6 rows, 12 columns). Participants completed an equal number of static and dynamic screens for their assigned display type.

The experiment can be described as a 3x2x3 factorial experiment. These factors were the display type (bar graph display, CCD, CID) which was a between-groups factor, the dynamicity (static, dynamic) which was a within-subjects factor, and the number of displays (8, 32, 72) which was also a within-subjects factor.

Overall test structure

| Phase | No. Displays | Dynamicity | No. Screens |
|--|--------------|------------|-------------|
| Sequential Display Training | 8 | Static | 1 |
| | | Dynamic | 1 |
| | 32 | Static | 1 |
| | | Dynamic | 1 |
| Known Unique State Training | 8 | Static | 4 |
| | | Dynamic | 4 |
| | 32 | Static | 4 |
| | | Dynamic | 4 |
| | 72 | Static | 4 |
| | | Dynamic | 4 |
| Known Unique State Performance Check | 8 | Static | 4 |
| | | Dynamic | 4 |
| | 32 | Static | 4 |
| | | Dynamic | 4 |
| | 72 | Static | 4 |
| | | Dynamic | 4 |
| Unknown Unique State Performance Check (not part of main study) | 8 | Static | 4 |
| | | Dynamic | 4 |
| | 32 | Static | 4 |
| | | Dynamic | 4 |
| | 72 | Static | 4 |
| | | Dynamic | 4 |

Table 3: Overall test structure

5.2.7 Test procedure

The test was divided into four distinct phases. The first two phases (Sequential Display Training and Known Unique State Training) were intended to train participants in all skills necessary for the following ‘Performance Check’ phases. The third phase (Known Unique State Performance Check) was the focal point of the test, and the phase on which our primary data analysis was conducted. The fourth and final phase of the test (Unknown Unique State Performance Check) was an experimental redesign of the third phase which served as a pilot test for a potential future study.

In the first phase of the test, participants were required to identify the state of each display shown on the screen in sequential order.

In the latter three 'Unique State' phases of the test, participants were required to identify a single unique display which was in a different state than all the others. For the static display screens, this unique display existed from the beginning. For the dynamic screens, one display would enter the unique state after a random delay, which could take up to 45 seconds. These latter three phases consisted of an identical set of screen configurations and utilised a shared data set, which was randomly shuffled for each performance phase.

It is important to reiterate that all randomness mentioned in this thesis is deterministic. That is, while the values were in fact randomly generated, this occurred only once when the master data set was originally generated. Each test participant received the exact same master data set with randomly generated values — be that for shuffling the order of displays, shuffling the order of screens, or for deciding the duration after which a dynamic display should transition into a unique state.

Computer assignment

All computers were numbered and randomly preassigned to a specific display type. At the beginning of the test, participants were asked to select a random computer assignment card from a stack. They then sat at the computer corresponding to the number on the card they chose.

Informed consent form

All participants read and signed an informed consent form before choosing to participate in the study. This informed participants about the purpose of the research study, their tasks, confidentiality of data, and their right to withdraw.

Pre-test survey

All participants completed a pre-test survey (Appendix B) intended to gather basic demographic information such as their gender, their age, the country in which they lived for the longest time, for how many years they studied English, if they were currently studying, and if so, their programme of study.

Starting instructions

The test began by presenting the participants with a set of general instructions applicable to all phases of the test (Appendix A, Figure 1; Appendix A, Figure 2; Appendix A, Figure 3). These instructions explained the four different systems states, how the participant could identify each state for their assigned display type, and provided visual examples of displays in each of the four possible states. The participants were also told that it was important to be as accurate as possible when answering, while completing the tasks without unnecessary delays.

Sequential Display Training

Participants began the test with the Sequential Display Training phase. They received a brief set of instructions specific to this particular phase (Appendix A, Figure 4), describing how they should indicate their choice of state for each display.

The Sequential Display Training phase consisted of four screens in which the user was required to identify the state of each display shown in sequential order.

Participants input their selection for the state of each display using the numeric row on their keyboards. In this phase, participants were immediately told whether their chosen answer was correct or incorrect. If their chosen answer was correct, a label in the format “YES: *N*” was shown below the display, where *N* was the state which they had chosen. If their chosen answer was incorrect, a label in the format “NO. Answer: *N*” was shown below the display, where *N* was the actual state of the display (Appendix A, Figure 9). This type of feedback was chosen as a form of reinforcement learning for the participants.

Following this phase, participants were then to proceed through each of the ‘Unique State’ test phases.

Known Unique State Training

Before the first of these phases, Known Unique State Training, participants received detailed instructions on how these ‘Unique State’ phases would work. They were provided with a visual example of a set of eight displays in which one display was in a unique state. They also received instructions regarding how they were required to identify a unique display, press the space bar, and then use the mouse to make their selection (Appendix A, Figure 11; Appendix A, Figure 12; Appendix A, Figure 13).

In the Known Unique State Training phase, participants were told the specific state of the unique display which they were required to identify (Appendix A, Figure 17). When the participant had made their choice, they were told whether or not their answer was correct. If their answer was correct, the label “Correct” was displayed beneath their chosen display (Appendix A, Figure 24). If their answer was incorrect, the label “Incorrect” was displayed beneath their chosen display, and the actual unique display was clearly outlined so that it could be used as a method of training (Appendix A, Figure 25).

Known Unique State Performance Check

Before the next phase, Known Unique State Performance Check, participants received a brief set of instructions informing them that they would be told the state of the unique display which they were required to identify, but that they would not receive feedback on their answers (Appendix A, Figure 26).

In the Known Unique State Performance Check phase, participants were once again told the specific state of the unique display which they were required to identify (Appendix A, Figure 27). However, in this phase, participants were not told whether or not their answer was correct. Regardless of their choice, only the label “Answer recorded” was shown beneath their chosen display (Appendix A, Figure 28).

Unknown Unique State Performance Check

Before the final phase, Unknown Unique State Performance Check, participants received a brief set of instructions informing them that they would not be told the state of the unique display which they were required to identify (Appendix A, Figure 30).

In the Unknown Unique State Performance Check phase, the instruction text simply read “Press the space bar when you find the display in a unique state” (Appendix A, Figure 31). As in the previous phase, participants were not told whether or not their

answer was correct, and the only feedback given was an “*Answer recorded*” label beneath their chosen display (Appendix A, Figure 32).

Unique State screen interactions

For these ‘Unique State’ screens in which participants needed to distinguish a single unique display from a set of other displays, they first indicated that they believed they had identified the unique display by pressing the space bar. All of the displays on the screen were then obscured by solid square overlays, to prevent the participant from further analysing the displays after already pressing the space bar. This was important as the response time calculation was based on when the space bar was pressed. The participant then used the mouse to click on the square overlay now obscuring the unique display that they had identified. Following this, the overlays were hidden once again to allow the participant to analyse their answer and receive feedback on their decision. This keypress-then-click approach was employed to increase the accuracy of measuring the participants’ response times. Simply asking the user to click on the unique display they identified without first pressing the space bar would introduce a confounding variable based on how quickly and accurately each participant could move their mouse cursor, as well as on where exactly their mouse cursor was located before they identified the unique display.

Post-test questionnaire

Upon finishing the test, all participants completed a short post-test questionnaire (Appendix C) consisting of two questions. The first question asked how confident they were that they had identified the unique display on the last four screens. It was decided to ask about the last four screens only as these would be the most recent in the participants’ memories, and these were deemed to be the most difficult screens of the entire test. The second question asked the participants about their method of finding the unique display on the last four screens.

5.2.8 Hypotheses

Based on the research questions, the following three hypotheses were formulated:

1. The bar graph display is less efficient than the CCD and the CID.
2. The CID becomes more efficient than the bar graph display and the CCD for 32-display and 72-display layouts.
3. Dynamicity leads to a greater efficiency decrease for the bar graph display than for the CCD or the CID.

5.2.9 Real experiments

The real user testing experiments took place in two different sessions. Both testing sessions took place on the same day, in the same location, with a one-hour gap in between.

The testing sessions consisted of 32 participants in total, 19 male and 13 female, ranging in ages from 19 to 34, with an average age of 28. 10 participants were assigned to the bar graph display type, 11 participants were assigned to the configural coordinate display type, and 11 participants were assigned to the configural integral display type. The shortest test time was 36 minutes and 55 seconds, while the longest test time was 72 minutes and 24 seconds. The first testing session consisted of 15 participants, and the second consisted of 17 participants.

6 Results

6.1 Participant feedback

We gathered extensive first-person accounts of how participants searched for the unique display on the final screens and found that many reported visually scanning through the displays either row-by-row or column-by-column. Some participants reported scanning sequentially and others reported scanning in random patterns, but most reported trying to pay individual attention to one display at a time in general.

6.2 Demographic data

We analysed the data gathered about participants' genders, ages, academic disciplines, and countries. We cross-checked this data and did not find any noticeable impact of participant demographics on test results. The data we are making available includes this demographic data in a further anonymised format, to protect the privacy of study participants. Exact ages have been converted to age ranges, and study programmes have been converted to broader study programme classifications. An overview of the demographic breakdown was provided in the *Demographics* section of the *Method* subchapter above.

6.3 Data analysis

Only the data from the main phase of the experiment, Known Unique State Performance Check, will be presented here. The specific instructions given to all participants regarding accuracy and speed were as follows: *"It is important that you try to be as accurate as possible while completing the tasks without unnecessary delays"*. Hence, participants were supposed to place greater importance on accuracy than on speed.

Therefore, the first analysis conducted was to investigate the participants' accuracy levels. Given that the results of this first investigation showed that the participant followed the instructions well (i.e. exhibited a high level of accuracy), their response data was then to be used in the further analysis of response times.

6.3.1 Exclusion of data points

It was decided to exclude from analysis certain subsets of the data in order to maximise the integrity of the results. This was done by establishing a set of inclusion criteria which participants' answers were required to meet in order for their data to be considered valid. The first criterion was that the participant must complete the entire test, and the second criterion was that the participant's overall accuracy must exceed a minimum threshold of 50%.

One test participant was excluded due to leaving the experiment with ten screens remaining and therefore not satisfying the first criterion. Three more participants were excluded due to exceptionally low accuracy rates, therefore not satisfying the second criterion, and indicating that they either did not fully understand the instructions or had simply given up at some point.

The final phase, Unknown Unique State Performance Check, was also excluded from detailed data analysis as it was to be considered a pilot for a future study.

6.4 Accuracy

Accuracy for static displays was measured as whether or not a participant correctly identified the single unique state display. Accuracy for dynamic displays was measured as whether or not a participant correctly identified the single unique state display, given that it had already begun or completed its transition into the unique state.

| | 8-display | | 32-display | | 72-display | |
|------------|-----------|---------|------------|---------|------------|---------|
| | Static | Dynamic | Static | Dynamic | Static | Dynamic |
| Bar | 100% | 100% | 100% | 96.9% | 100% | 68.8% |
| CCD | 97.5% | 97.5% | 100% | 100% | 100% | 100% |
| CID | 100% | 95.0% | 95.0% | 95.0% | 87.5% | 85.0% |

Table 4: Known Unique State Performance Check — Accuracy by display type, layout, and dynamicity

Due to the lower than desirable accuracy exhibited by the 72-display layout for both the bar graph display and the CID, it was decided to exclude this display layout from further analysis.

6.5 Response times

Response times for static displays were measured as the time from initial screen load until the time at which the participant pressed the space bar. Response times for dynamic displays were measured as the time from when the eventual unique state display began its transition into the unique state until the time at which the participant pressed the space bar. Only the times for the last screen in each block of four identical screen configurations (shown in Table 3) are evaluated below, in order to give participants as much training time with each configuration as possible. All times shown below are calculated average means.

| | 8-display | | 32-display | |
|------------|-----------|---------|------------|---------|
| | Static | Dynamic | Static | Dynamic |
| Bar | 8.850s | 7.172s | 22.324s | 28.219s |
| CCD | 3.788s | 9.479s | 10.086s | 9.721s |
| CID | 4.483s | 9.320s | 11.945s | 11.208s |

Table 5: Known Unique State Performance Check — Response time in seconds by display type, layout, and dynamicity

The response times were analysed through a split-plot ANOVA with Display Type (Bar, CCD, CID) as the between-groups factor and Dynamicity (Static, Dynamic) and Number of Displays (8, 32) as the within-subject factors. The results are shown in the table below. A significance level of 5% was chosen.

| Effect | SS | df | MS | F | P |
|--|--------------|----|--------------|---------|----------|
| Display Type | 1.440851E+09 | 2 | 7.204253E+08 | 5.5064 | 0.010436 |
| Error | 3.270859E+09 | 25 | 1.308343E+08 | | |
| Dynamicity | 1.432017E+08 | 1 | 1.432017E+08 | 2.4273 | 0.131808 |
| Dynamicity × Display Type | 2.227620E+06 | 2 | 1.113810E+06 | 0.0189 | 0.981312 |
| Error | 1.474899E+09 | 25 | 5.899596E+07 | | |
| Number of Displays | 1.954840E+09 | 1 | 1.954840E+09 | 17.3150 | 0.000327 |
| Number of Displays × Display Type | 1.018959E+09 | 2 | 5.094794E+08 | 4.5127 | 0.021218 |
| Error | 2.822459E+09 | 25 | 1.128984E+08 | | |
| Dynamicity × Number of Displays | 1.266673E+07 | 1 | 1.266673E+07 | 0.2302 | 0.635515 |
| Dynamicity × Number of Displays × Display Type | 2.563267E+08 | 2 | 1.281633E+08 | 2.3296 | 0.118099 |
| Error | 1.375361E+09 | 25 | 5.501446E+07 | | |

Table 6: Repeated measures analysis of variance (ANOVA)

The first hypothesis receives support as there is a significant effect of Display Type ($F(2,25)=5.5064$, $p<0.0105$). To further investigate this main effect, a post-hoc analysis was performed using Scheffé's procedure. This showed that the bar graph display was significantly different from both the CCD and the CID, and the latter two were not significantly different from one another.

There was a main effect of Number of Displays ($F(1,25)=17.3150$, $p<0.0004$). The mean response time was 7.182 seconds for the 8-display condition and 15.584 seconds for the 32-display condition.

There was also a significant interaction effect between Number of Displays and Display Type ($F(2,25)=4.5127$, $p<0.0213$). A post-hoc test, also using Scheffé's procedure, revealed that the bar graph display produced longer response times compared to the other two display types and that this difference became significantly larger at 32 displays compared to at 8 displays. There were no significant differences between the CCD and CID at 8 displays nor at 32 displays. Thus, the second

hypothesis does not receive support, although it is interesting to note that the bar graph display performs increasingly worse with an increasing number of displays.

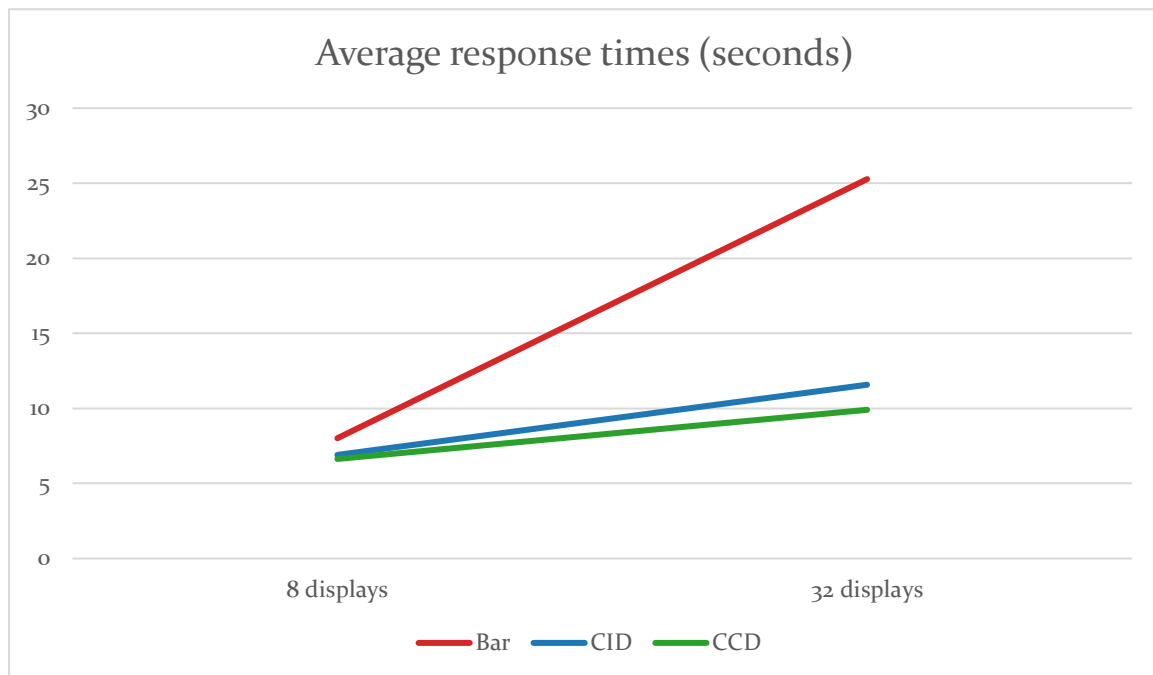


Figure 9: Average response time by display type and number of displays

There was no main effect of Dynamicity nor any interaction effects. Therefore, the third hypothesis does not receive any support.

7 Discussion and conclusion

7.1 First research question

Regarding the first research question, *“How does the performance of each display type compare for the purpose of anomaly detection after training?”*, we observed that the bar graph display performed substantially worse than the CCD and CID. This supports the first hypothesis that *“The bar graph display is less efficient than the CCD and the CID”*.

Although the bar graph display can be interpreted in a somewhat integral manner as mentioned earlier in this thesis (i.e. ‘slope up, slope down’), it is still an inherently separable display type and thus more individual attention is required to ascertain the specific state of each display. This is much different to the CCD and CID which have far more integral properties, and thus allow the operator to perform a visual search at a much faster pace.

Additionally, while the CCD performed slightly better than the CID for anomaly detection regarding system states, it is possible that the CID could perform better for anomaly detection concerning specific variables, as it presents the variable bars in a more integral manner.

7.2 Second research question

The second research question we aimed to explore was *“Do any of the display types afford gist perception, and if so, under what circumstances?”*. As can be seen in the ‘Average response times’ figure above, all display types exhibited a similar average response time for the 8-display layout.

However, at the 32-display level, the latency for the bar graph display became much greater. The increase in latency for the bar graph display was approximately 215%, while the increase in the number of displays was 300% (8 to 32). This would indicate that the majority of visual searches for this bar graph layout were performed in some type of serial manner.

Conversely, the overall increase in latency between the 8-display and 32-display layout for the CCD was approximately 49% and for the CID was approximately 68%, despite the number of displays increasing by 300%. These much lower relative increases in latency would indicate that some form of parallel visual search was taking place with these display types for the larger number of displays.

Despite this, the second hypothesis that *“The CID becomes more efficient than the bar graph display and the CCD for 32-display and 72-display layouts”* was not supported, as the performance of the CCD was never surpassed by the CID.

7.3 Third research question

The third and final research question investigated was *“What are the effects of dynamicity in data on the performance of the display types?”*.

This was investigated through the specific hypothesis that *“Dynamicity leads to a greater efficiency decrease for the bar graph display than for the CCD or the CID”*. It was observed that dynamicity in the data did not have a statistically significant impact on the total response times overall, and therefore this hypothesis was not supported.

It is interesting to note, however, that for the 8-display layout, dynamicity had little effect on the bar graph display response times and had a greater effect on the CCD and CID response times. For the 32-display layout, this effect was inverted — dynamicity had a greater effect on the bar graph display response times and had a much smaller effect on the response times for the CCD and CID.

It is also worth noting the difference in response times caused by the increase in the number of displays, and how this differs depending on whether the displays are static or dynamic. For static bar graph displays, the increase from 8 to 32 displays caused a latency increase of approximately 152%, while for dynamic bar graph displays this latency increase was approximately 293%. For static CCDs and CIDs, the latency increases were both approximately 166%. Most interestingly, for dynamic CCDs and CIDs, the increase in the number of displays caused approximate latency increases of just 3% and 20% respectively.

While there was no statistically significant effect of dynamicity on response times, the large differences in mean values highlighted in the preceding two paragraphs indicates that the potential effects of dynamicity could use further attention.

It is possible that this statistically insignificant overall effect of dynamicity was due to the specific manifestations of dynamicity that were chosen, as well as the general movement patterns portrayed. While there is an abundance of potential dynamic movement patterns that could be applied to our display types, an effort was made to keep the patterns in this experiment relatively simple, as this was the first study integrating dynamicity into this specific set of displays.

7.4 Fatigue

Löfvenberg (2016) stated that the total duration of their experiment was longer than expected and some participants exhibited signs of fatigue by the end. This could also be a contributing factor to the decreased accuracy towards the end of our own experiment, particularly for the bar graph display and the CID, as participants assigned to these display types had the longest total test durations on average.

8 Future work

While the CCD and the CID performed quite well in our study, it remains an open question whether there is a display type that allows for highly efficient anomaly identification and parallel processing, especially when it comes to dealing with larger data sets. A display type that allows users to recognise unique or anomalous states would likely also support increased gist perception. A good starting point is to consider the properties that such a display may exhibit.

As our study has shown, the CCD identifies the display state very clearly, and could therefore provide inspiration for future designs. The best element of the CCD for efficient state identification is the fact that it portrays a non-ambiguous state classification. Users can determine the state of almost any CCD with absolute certainty after brief training.

A question to be asked regarding the CID is what can be done to make state identification clearer than when just using the ‘centre of gravity’ method. For instance, it is likely that the use of contrast and colour would play a major role in such a display. It is also likely that the more visually abrupt a state change is, the better it would perform in anomaly recognition tasks, as observed in our first pilot test.

The aim of a future study should be to identify a display type that supports the following:

- **Discoverability:** It should be easy to determine that a single display is unique or anomalous within a larger set.
- **Classification:** It should be easy to determine the exact system state represented by the display.
- **Analysis:** The system operator should be able to effectively view and compare the underlying system variable values.

8.1 Potential redesigns

Potential starting points for redesigns of the CID are offered below.

The first potential redesign (Figure 10) inverts the opacity of the variable bars and the bounding box. In the design chosen for our experiment, the variable bars were rendered in solid white, while the bounding box was rendered at 60% opacity. This redesign puts less visual emphasis on the variable bars by rendering them at 60% opacity and puts more visual emphasis on the bounding box by rendering it in solid white.

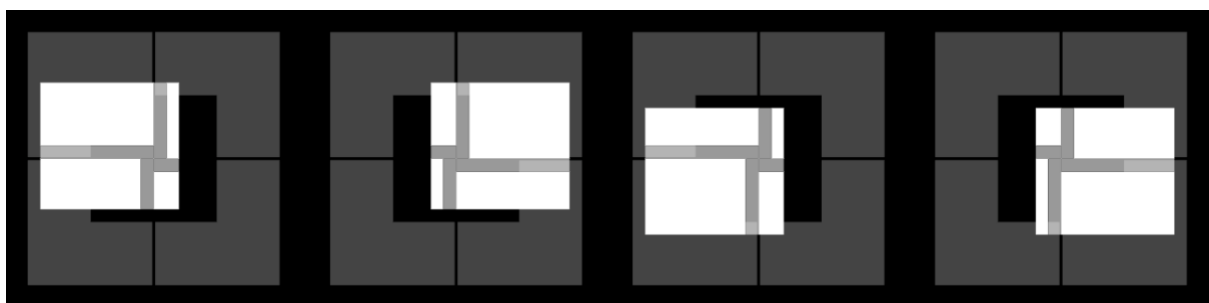


Figure 10: CIDs with inverted opacity of bars and bounding boxes

The next two redesigns (Figure 11 and Figure 12) make heavy use of state-specific coloured regions as this is a highly separable feature proven to be beneficial in gist perception.

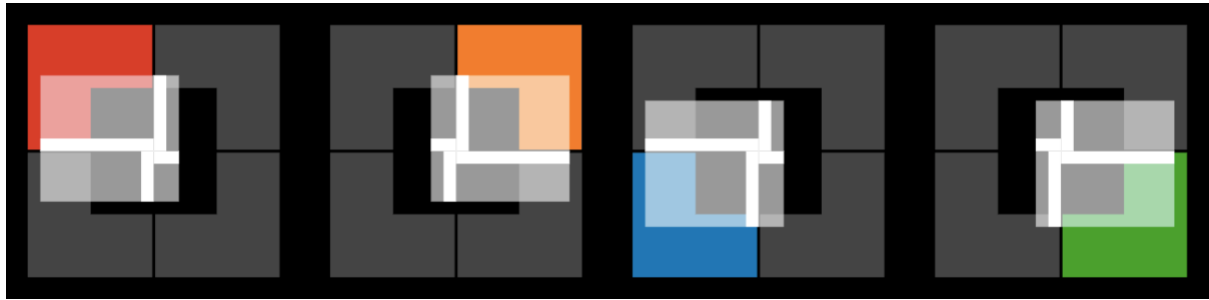


Figure 11: CIDs with gravitational corner quadrants illuminated in state-specific colours



Figure 12: CIDs with all corner quadrants illuminated in state-specific colours

8.2 Dynamicity

There is a wide breadth of possible dynamicity which could be added to these display types to better emulate real-world movement patterns. As mentioned earlier, we decided to limit the scope of dynamicity in this study to interval-based, involatile transitions which exhibited mostly uniform transition speed patterns. However, it is likely that many real-world environments would exhibit more random, erratic, and non-linear transition patterns. In fact, it is entirely plausible that different display types would perform better based on the specific properties and behaviours of the dynamic data they are representing, and therefore this could justify an entire research effort in and of itself.

8.3 Availability of software testbed and data

The software testbed is made available for other researchers to use as a basis to conduct their own studies and investigate their own theories regarding these or other display types. The test results data is also being made available for further analysis by interested parties. The software testbed and test results data will be available at thesis.darrenlehane.com.

9 References

- Anokhin, A., Ivkin, A., & Dorokhov, S. (2018). Application of ecological interface design in nuclear power plant (NPP) operator support system. *Nuclear Engineering and Technology*, 50(4), 619–626.
<http://doi.org/10.1016/j.net.2018.03.005>
- Borst, C., Flach, J. M., & Ellerbroek, J. (2015). Beyond ecological interface design: Lessons from concerns and misconceptions. *IEEE Transactions on Human-Machine Systems*, 45(2), 164–175. <http://doi.org/10.1109/THMS.2014.2364984>
- Burns, C. M., Skraaning, G., Jamieson, G. A., Lau, N., Kwok, J., Welch, R., & Andresen, G. (2008). Evaluation of ecological interface design for nuclear process control: Situation awareness effects. *Human Factors*, 50(4), 663–679.
<http://doi.org/10.1518/001872008X312305>
- Buttigieg, M., Sanderson, P. M., & Flach, J. M. (1988). Object vs. separate displays for process failure detection: The emergent features approach. *Proceedings of the Human Factors Society Annual Meeting*, 32(19), 1340–1344.
<http://doi.org/10.1177/154193128803201907>
- Carswell, C. M., & Wickens, C. D. (1987). Information integration and the object display an interaction of task demands and display superiority. *Ergonomics*, 30(3), 511–527. <http://doi.org/10.1080/00140138708969741>
- Coury, B. G., Boulette, M. D., & Smith, R. A. (1989). Effect of uncertainty and diagnosticity on classification of multidimensional data with integral and separable displays of system status. *Human Factors*, 31(5), 551–569.
<http://doi.org/10.1177/001872088903100505>
- Coury, B. G., & Pietras, C. M. (1989). Alphanumeric and graphic displays for dynamic process monitoring and control. *Ergonomics*, 32(11), 1373–1389.
<http://doi.org/10.1080/00140138908966912>
- Coury, B. G., & Purcell, J. A. (1988). The bar graph as a configural and separable display. *Proceedings of the Human Factors Society Annual Meeting*, 32(19), 1361–1365. <http://doi.org/10.1177/154193128803201912>
- de Kleer, J., & Brown, J. S. (1988). Assumptions and ambiguities in mechanistic mental models. In A. Collins & E. E. Smith (Eds.), *Readings in Cognitive Science* (pp. 270–287). <http://doi.org/10.1016/B978-1-4832-1446-7.50023-6>
- Eriksen, C. W., & Hoffman, J. E. (1972). Temporal and spatial characteristics of selective coding from visual displays. *Perception & Psychophysics*, 12(2), 201–204.
<http://doi.org/10.3758/BF03212870>
- Foley, J. D., & McMath, C. F. (1986). Dynamic process visualization. *IEEE Computer Graphics and Applications*, 6(3), 16–25. <http://doi.org/10.1109/MCG.1986.276628>
- Goodhew, S. C., Shen, E., & Edwards, M. (2016). Selective spatial enhancement: Attentional spotlight size impacts spatial but not temporal perception. *Psychonomic Bulletin and Review*, 23(4), 1144–1149.
<http://doi.org/10.3758/s13423-015-0904-6>
- Henneman, R. L., & Rouse, W. B. (1987). Human problem solving in dynamic environments: Understanding and supporting operators in large-scale complex systems.
- Ho, D., & Burns, C. M. (2003). Ecological interface design in aviation domains: Work domain analysis of automated collision detection and avoidance. *Proceedings of*

- the Human Factors and Ergonomics Society Annual Meeting*, 47(1), 119–123.
<http://doi.org/10.1177/154193120304700125>
- Holt, J., Bennett, K. B., & Flach, J. M. (2015). Emergent features and perceptual objects: Re-examining fundamental principles in analogical display design. *Ergonomics*, 58(12), 1960–1973. <http://doi.org/10.1080/00140139.2015.1049217>
- Jamieson, G. A. (2007). Ecological interface design for petrochemical process control: An empirical assessment. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 37(6), 906–920.
<http://doi.org/10.1109/TSMCA.2007.897583>
- Laaksoharju, M., Lind, M., & Jansson, A. A. (2017). Adding integral display properties to increase generalizability of a configural display. *SweCog 2017*. Uppsala.
- Lipsett, J. J., Olmstead, R. A., & Stevens, J. E. S. (1989). Balancing the roles of humans and machines in power plant control. *AECL--9955*.
- Löfvenberg, J. (2016). *Poietic design - Heuristics and applications*.
- Nakayama, K., & Silverman, G. H. (1986). Serial and parallel processing of visual feature conjunctions. *Nature*, 320(6059), 264–265.
<http://doi.org/10.1038/320264a0>
- Narbutas, V., Lin, Y. S., Kristan, M., & Heinke, D. (2017). Serial versus parallel search: A model comparison approach based on reaction time distributions. *Visual Cognition*, 25(1–3), 306–325. <http://doi.org/10.1080/13506285.2017.1352055>
- Nickerson, R. S. (1969). Man-computer interaction: A challenge for human factors research. *Ergonomics*, 12(4), 501–517. <http://doi.org/10.1080/00140136908931076>
- Pomerantz, J. R., Pristach, E. A., & Carson, C. E. (1989). Attention and object perception. In B. E. Shepp & S. Ballesteros (Eds.), *Object perception: Structure and process* (pp. 53–89).
- Rasmussen, J. (1986). Information processing and human-machine interaction. *An Approach to Cognitive Engineering*.
- Rasmussen, J., & Vicente, K. J. (1989). Coping with human errors through system-design: Implications for ecological interface design. *International Journal of Man-Machine Studies*, 31(5), 517–534. [http://doi.org/10.1016/0020-7373\(89\)90014-X](http://doi.org/10.1016/0020-7373(89)90014-X)
- Sanderson, P. M., Flach, J. M., Buttigieg, M. A., & Casey, E. J. (1989). Object displays do not always support better integrated task performance. *Human Factors*, 31(2), 183–198. <http://doi.org/10.1177/001872088903100207>
- SINTEF. (2013). Big data, for better or worse: 90% of world's data generated over last two years. Retrieved from www.sciencedaily.com/releases/2013/05/130522085217.htm
- Treisman, A., & Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology*, 12(1), 97–136. [http://doi.org/10.1016/0010-0285\(80\)90005-5](http://doi.org/10.1016/0010-0285(80)90005-5)
- Vicente, K. J. (1999). *Cognitive work analysis: Toward safe, productive, and healthy computer-based work*.
- Vicente, K. J., & Rasmussen, J. (1992). Ecological interface design: Theoretical foundations. *Systems, Man, and Cybernetics, IEEE Transactions On*, 22(4), 589–606. <http://doi.org/10.1109/21.156574>
- Wickens, C. D., & Hollands, J. G. (1984). *Engineering psychology and human performance*.

Appendix A — Test application screenshots

Figure 1: Bar graph experiment instructions34

Figure 2: Configural coordinate display experiment instructions34

Figure 3: Configural integral display experiment instructions.....35

Figure 4: Sequential Display Training instructions35

Figure 5: Bar graph display, Sequential Display Training, static, 8-display (mid-completion)36

Figure 6: Configural coordinate display, Sequential Display Training, static, 8-display (mid-completion)36

Figure 7: Configural integral display, Sequential Display Training, static, 8-display (mid-completion).....37

Figure 8: Bar graph display, Sequential Display Training, static, 32-display (mid-completion)37

Figure 9: Configural coordinate display, Sequential Display Training, static, 32-display (mid-completion)38

Figure 10: Configural integral display, Sequential Display Training, static, 32-display (mid-completion)38

Figure 11: Bar graph display, Known Unique State Training instructions.....39

Figure 12: Configural coordinate display, Known Unique State Training instructions.....39

Figure 13: Configural integral display, Known Unique State Training instructions 40

Figure 14: Bar graph display, Known Unique State Training, static, 8-display 40

Figure 15: Configural coordinate display, Known Unique State Training, static, 8-display41

Figure 16: Configural integral display, Known Unique State Training, static, 8-display41

Figure 17: Bar graph display, Known Unique State Training, static, 32-display.....42

Figure 18: Configural coordinate display, Known Unique State Training, static, 32-display42

Figure 19: Configural integral display, Known Unique State Training, static, 32-display.....43

Figure 20: Bar graph display, Known Unique State Training, static, 72-display43

Figure 21: Configural coordinate display, Known Unique State Training, static, 72-display44

Figure 22: Configural integral display, Known Unique State Training, static, 72-display.....44

Figure 23: Example of screen with display overlays shown after space bar is pressed45

Figure 24: Known Unique State Training screen after the correct display overlay was clicked45

Figure 25: Known Unique State Training screen after an incorrect display overlay was clicked46

Figure 26: Known Unique State Performance Check instructions46

Figure 27: Configural integral display, Known Unique State Performance Check, dynamic, 32-display47

Figure 28: Configural integral display, Known Unique State Performance Check, dynamic, 32-display, correct.....47

Figure 29: Configural integral display, Known Unique State Performance Check, dynamic, 32-display, incorrect48

Figure 30: Unknown Unique State Performance Check instructions48

Figure 31: Configural coordinate display, Unknown Unique State Performance Check, static, 32-display49

Figure 32: Configural coordinate display, Unknown Unique State Performance Check, static, 32-display, correct49

Figure 33: Configural coordinate display, Unknown Unique State Performance Check, static, 32-display, incorrect50

Experiment on Reading Charts

In this experiment you will see a large number of charts, each representing 4 variables.

The 4 variables of a chart represent a **system state**, which can be either 1, 2, 3, or 4.

The state is determined by the relative size difference (slope) between the first and second bar as well as the relative size difference (slope) between the third and the fourth bar.

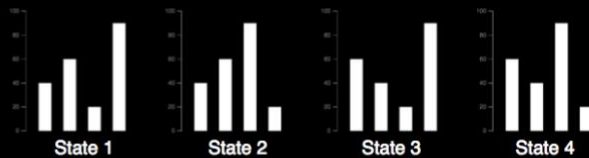
State 1 = First bar shorter than the second, and third bar shorter than the fourth. (Up, up)

State 2 = First bar shorter than the second, and third bar longer than the fourth. (Up, down)

State 3 = First bar longer than the second, and third bar shorter than the fourth. (Down, up)

State 4 = First bar longer than the second, and third bar longer than the fourth. (Down, down)

Chart examples



You should remember how each state looks.

Fast and accurate

It is important that you try to be as accurate as possible while completing the tasks without unnecessary delays. The pages are divided into training and performance checks. When training, you will receive immediate feedback on your answers. Do not waste time on analysing the results too much. During the performance check, you will receive no feedback on your answers.

[Begin the experiment →](#)

Appendix A, Figure 1: Bar graph experiment instructions

Experiment on Reading Charts

In this experiment you will see a large number of charts, each representing 4 variables.

The 4 variables of a chart represent a **system state**, which can be either 1, 2, 3, or 4.

To identify the state, look for the dot.

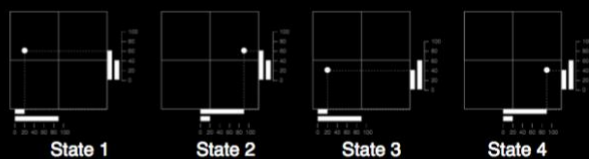
State 1 = The dot is in the upper left square.

State 2 = The dot is in the upper right square.

State 3 = The dot is in the lower left square.

State 4 = The dot is in the lower right square.

Chart examples



You should remember how each state looks.

Fast and accurate

It is important that you try to be as accurate as possible while completing the tasks without unnecessary delays. The pages are divided into training and performance checks. When training, you will receive immediate feedback on your answers. Do not waste time on analysing the results too much. During the performance check, you will receive no feedback on your answers.

[Begin the experiment →](#)

Appendix A, Figure 2: Configural coordinate display experiment instructions

Experiment on Reading Charts

In this experiment you will see a large number of charts, each representing 4 variables.

The 4 variables of a chart represent a **system state**, which can be either 1, 2, 3, or 4.

To identify the state, judge the centre of gravity of the light-gray box.

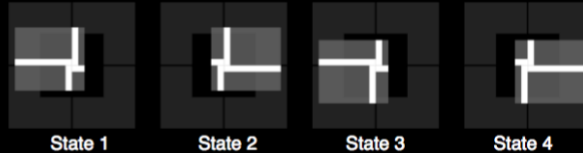
State 1 = The centre of gravity is offset towards the upper left corner.

State 2 = The centre of gravity is offset towards the upper right corner.

State 3 = The centre of gravity is offset towards the lower left corner.

State 4 = The centre of gravity is offset towards the lower right corner.

Chart examples



You should remember how each state looks.

Fast and accurate

It is important that you try to be as accurate as possible while completing the tasks without unnecessary delays. The pages are divided into training and performance checks. When training, you will receive immediate feedback on your answers. Do not waste time on analysing the results too much. During the performance check, you will receive no feedback on your answers.

[Begin the experiment →](#)

Appendix A, Figure 3: Configural integral display experiment instructions

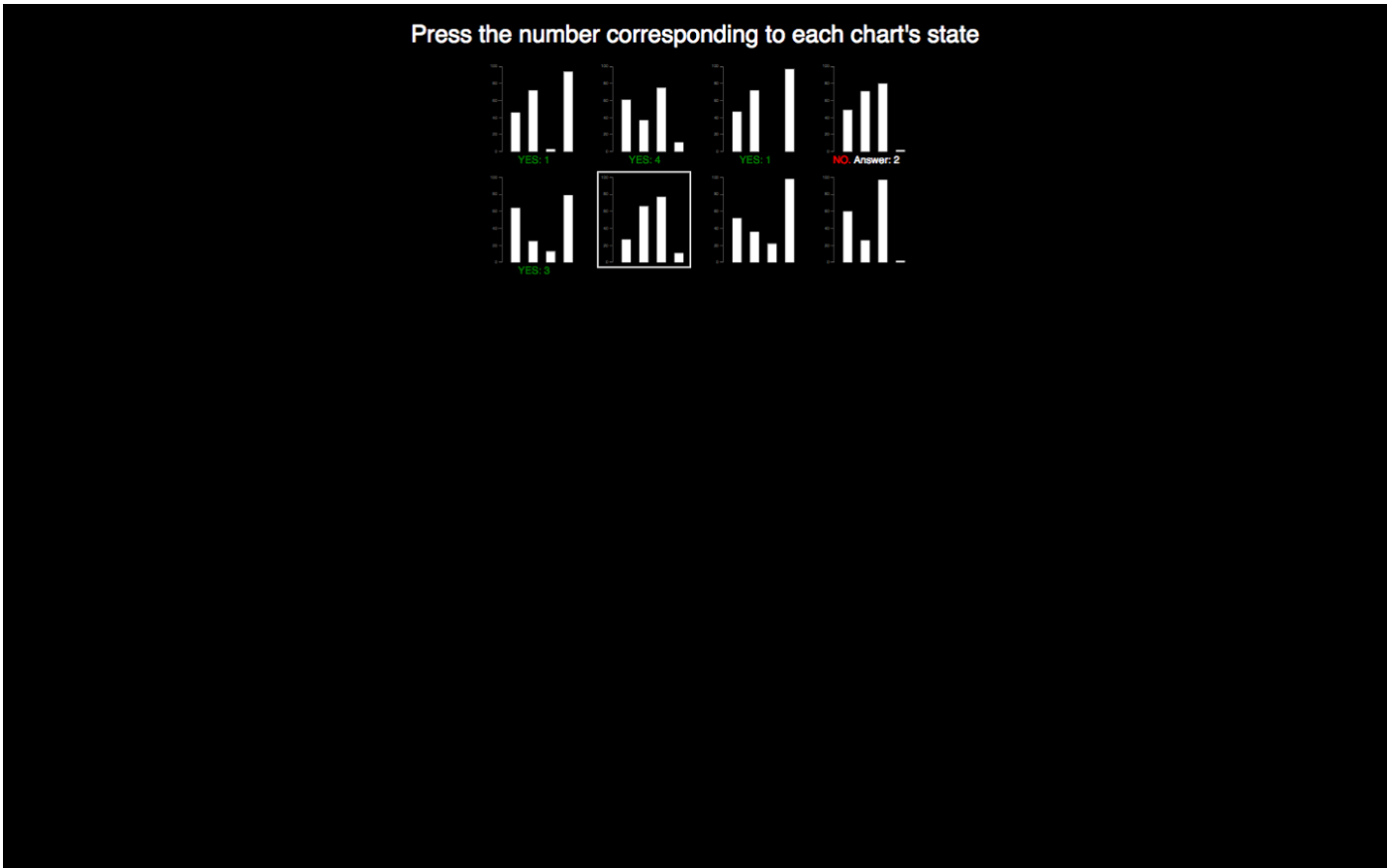
Training - Sequential Charts

You will now see sets of charts whose states you must identify in sequential order, starting from top-left. When you believe you have identified which state a chart is in (1, 2, 3, or 4), you should press the corresponding number key on your keyboard. After pressing the number key, you will receive feedback on your answer.

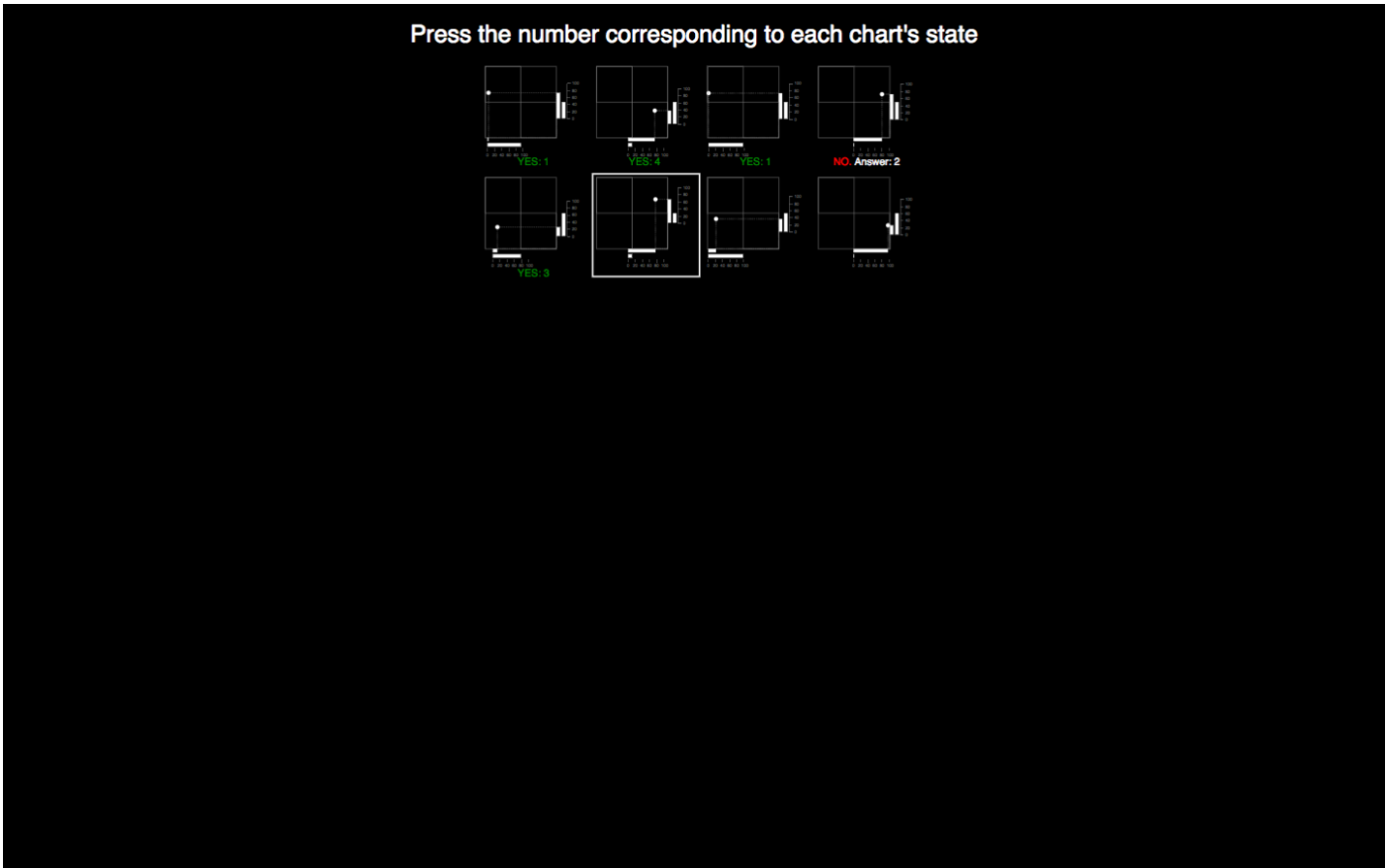
When you reach the end of the page and review your answers, press Enter to continue.

[Continue →](#)

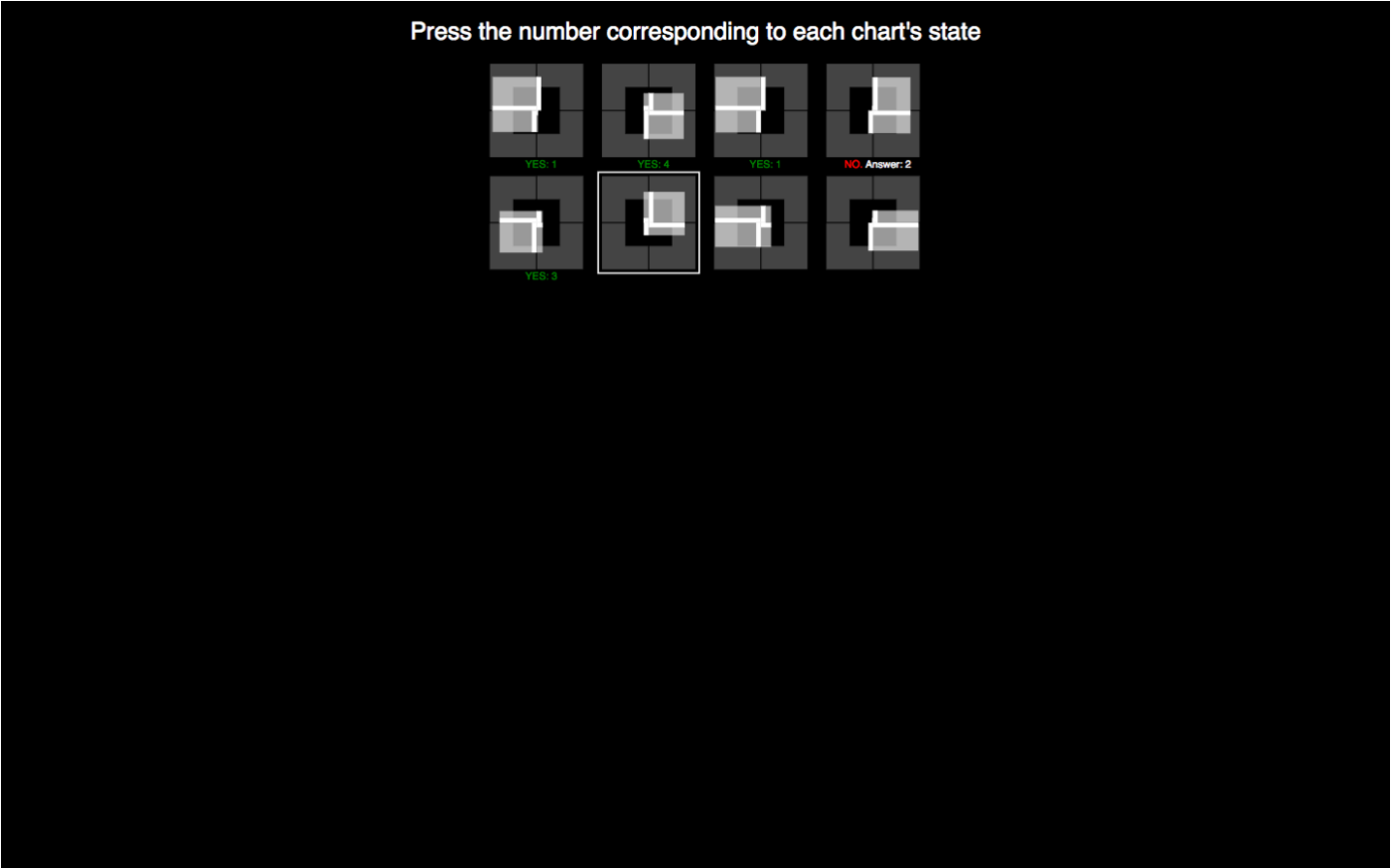
Appendix A, Figure 4: Sequential Display Training instructions



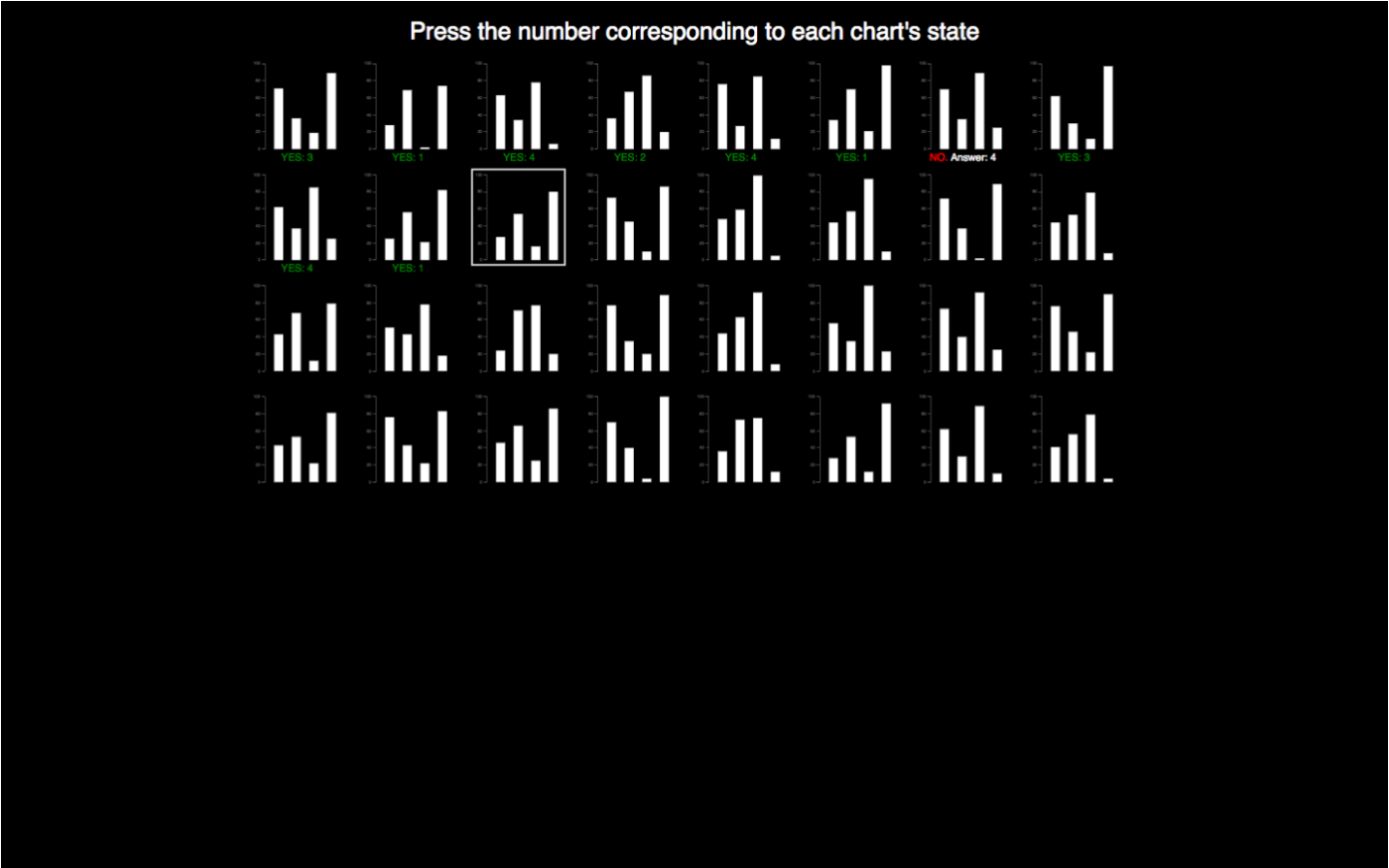
Appendix A, Figure 5: Bar graph display, Sequential Display Training, static, 8-display (mid-completion)



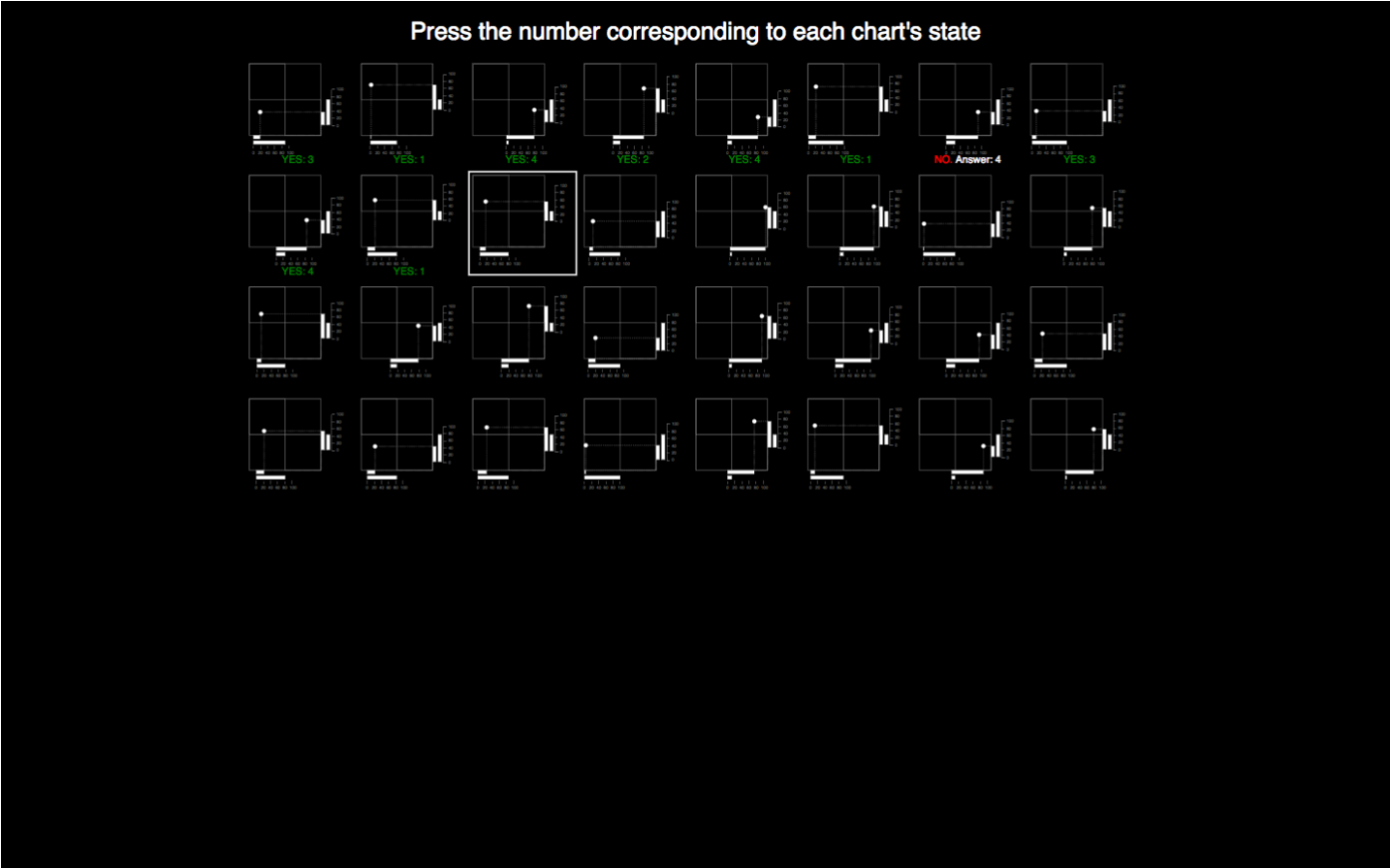
Appendix A, Figure 6: Configural coordinate display, Sequential Display Training, static, 8-display (mid-completion)



Appendix A, Figure 7: Configurational integral display, Sequential Display Training, static, 8-display (mid-completion)



Appendix A, Figure 8: Bar graph display, Sequential Display Training, static, 32-display (mid-completion)



Appendix A, Figure 9: Configural coordinate display, Sequential Display Training, static, 32-display (mid-completion)

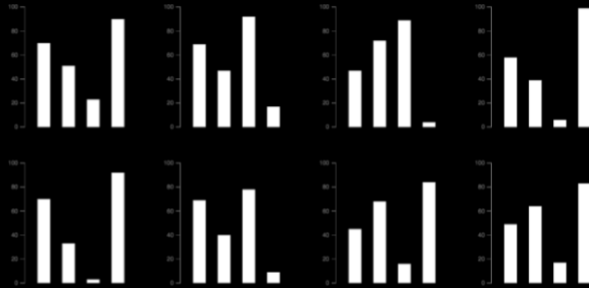


Appendix A, Figure 10: Configural integral display, Sequential Display Training, static, 32-display (mid-completion)

Training - Unique Chart

Next, you will see charts similar to those in the previous task. However, this time **one** of the charts will be in a different state than all the others.

Unique chart example



In this example, the **third** chart is unique because it is the only chart in state 2.

Your Task

When you identify this unique chart, **press the space bar**. All of the charts will then be covered by a square. Use the mouse to **click** on the square that is covering the unique chart you just identified.

On some pages, all charts will be static, and the unique chart will exist from the beginning.

On other pages, all charts will be dynamic (moving), and there will be no unique chart at the beginning. After some time, **one** of the charts will transition into an unique state *and remain there*. This can take up to a minute.

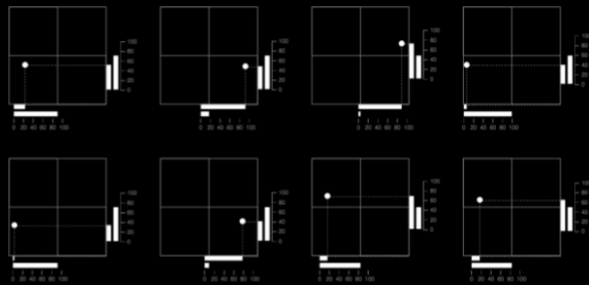
Continue →

Appendix A, Figure 11: Bar graph display, Known Unique State Training instructions

Training - Unique Chart

Next, you will see charts similar to those in the previous task. However, this time **one** of the charts will be in a different state than all the others.

Unique chart example



In this example, the **third** chart is unique because it is the only chart in state 2.

Your Task

When you identify this unique chart, **press the space bar**. All of the charts will then be covered by a square. Use the mouse to **click** on the square that is covering the unique chart you just identified.

On some pages, all charts will be static, and the unique chart will exist from the beginning.

On other pages, all charts will be dynamic (moving), and there will be no unique chart at the beginning. After some time, **one** of the charts will transition into an unique state *and remain there*. This can take up to a minute.

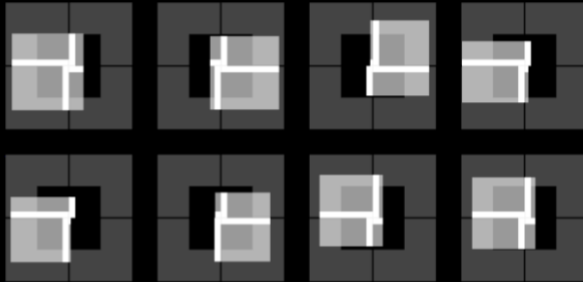
Continue →

Appendix A, Figure 12: Configural coordinate display, Known Unique State Training instructions

Training - Unique Chart

Next, you will see charts similar to those in the previous task. However, this time **one** of the charts will be in a different state than all the others.

Unique chart example



In this example, the **third** chart is unique because it is the only chart in state 2.

Your Task

When you identify this unique chart, **press the space bar**. All of the charts will then be covered by a square. Use the mouse to **click** on the square that is covering the unique chart you just identified.

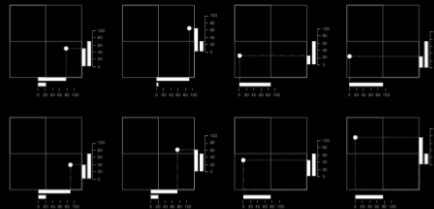
On some pages, all charts will be static, and the unique chart will exist from the beginning.

On other pages, all charts will be dynamic (moving), and there will be no unique chart at the beginning. After some time, **one** of the charts will transition into an unique state *and remain there*. This can take up to a minute.

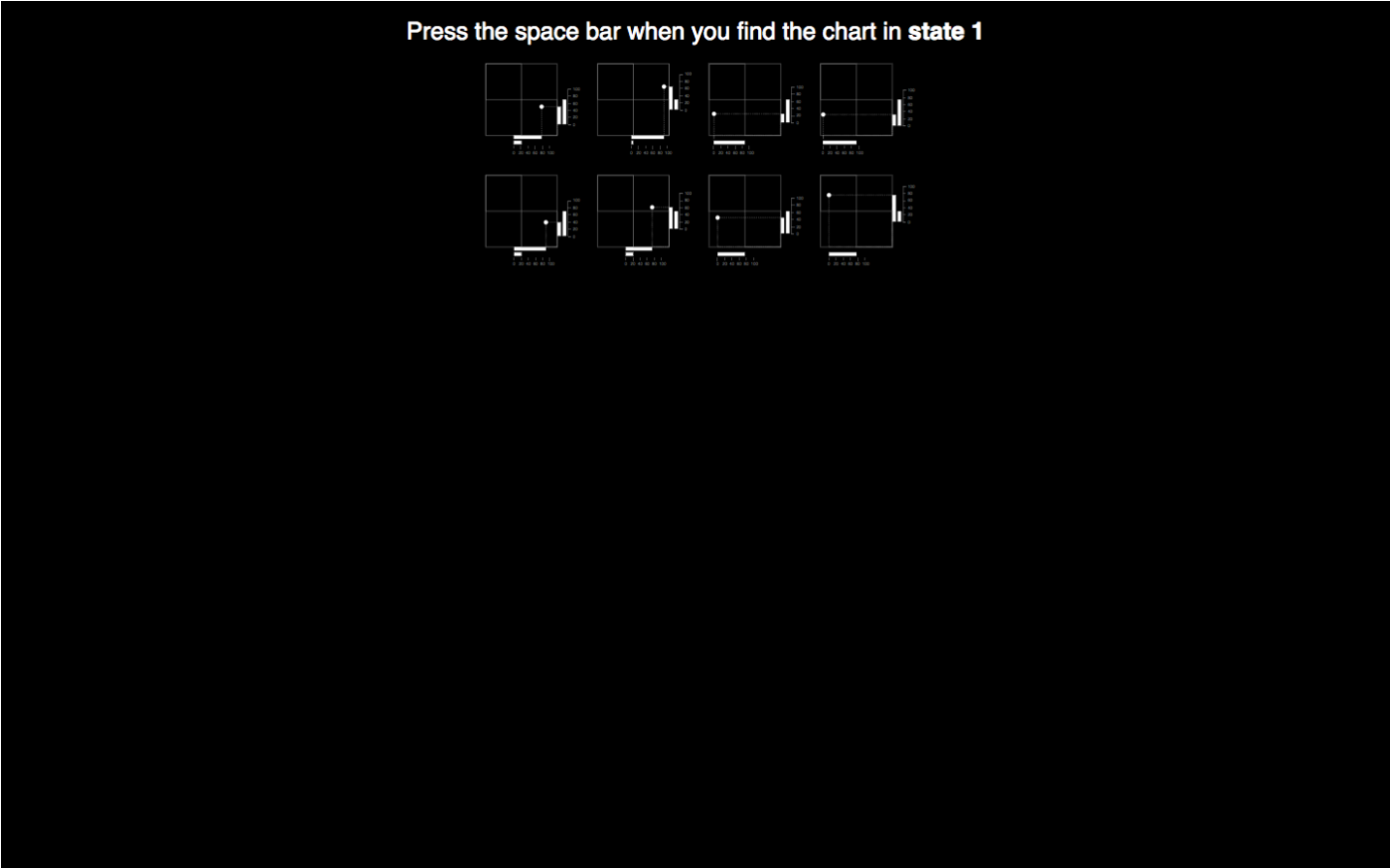
Continue →

Appendix A, Figure 13: Configural integral display, Known Unique State Training instructions

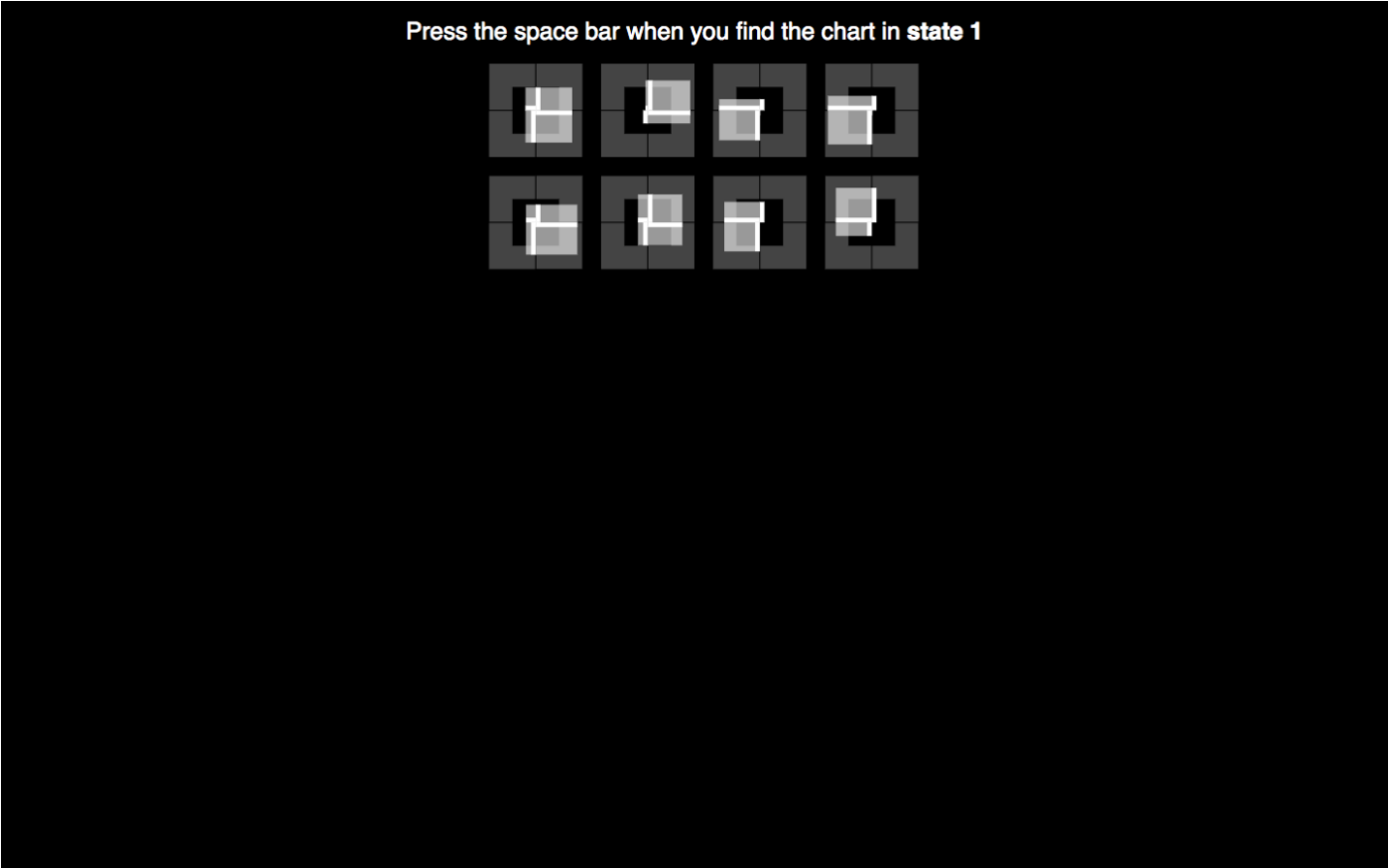
Press the space bar when you find the chart in **state 1**



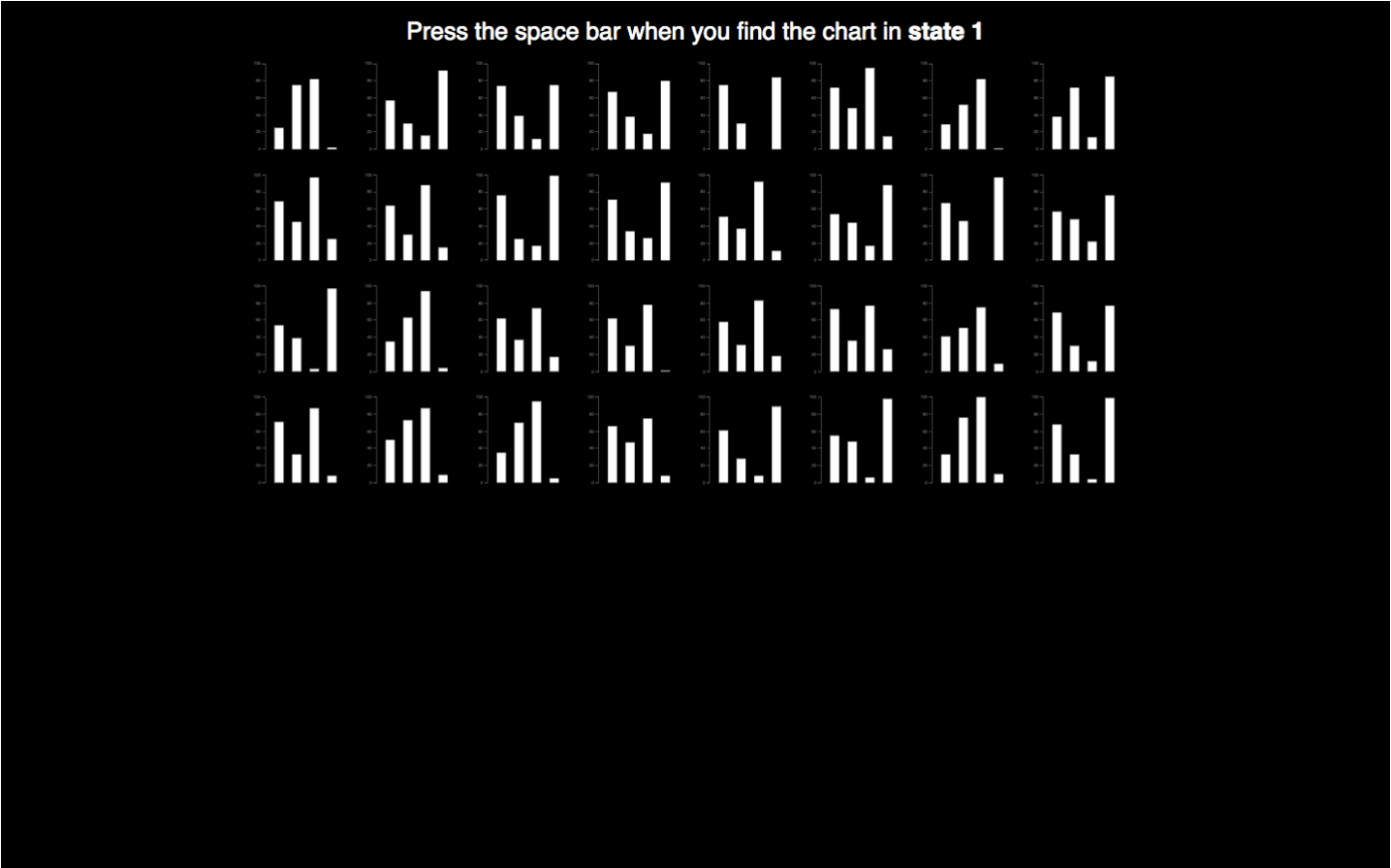
Appendix A, Figure 14: Bar graph display, Known Unique State Training, static, 8-display



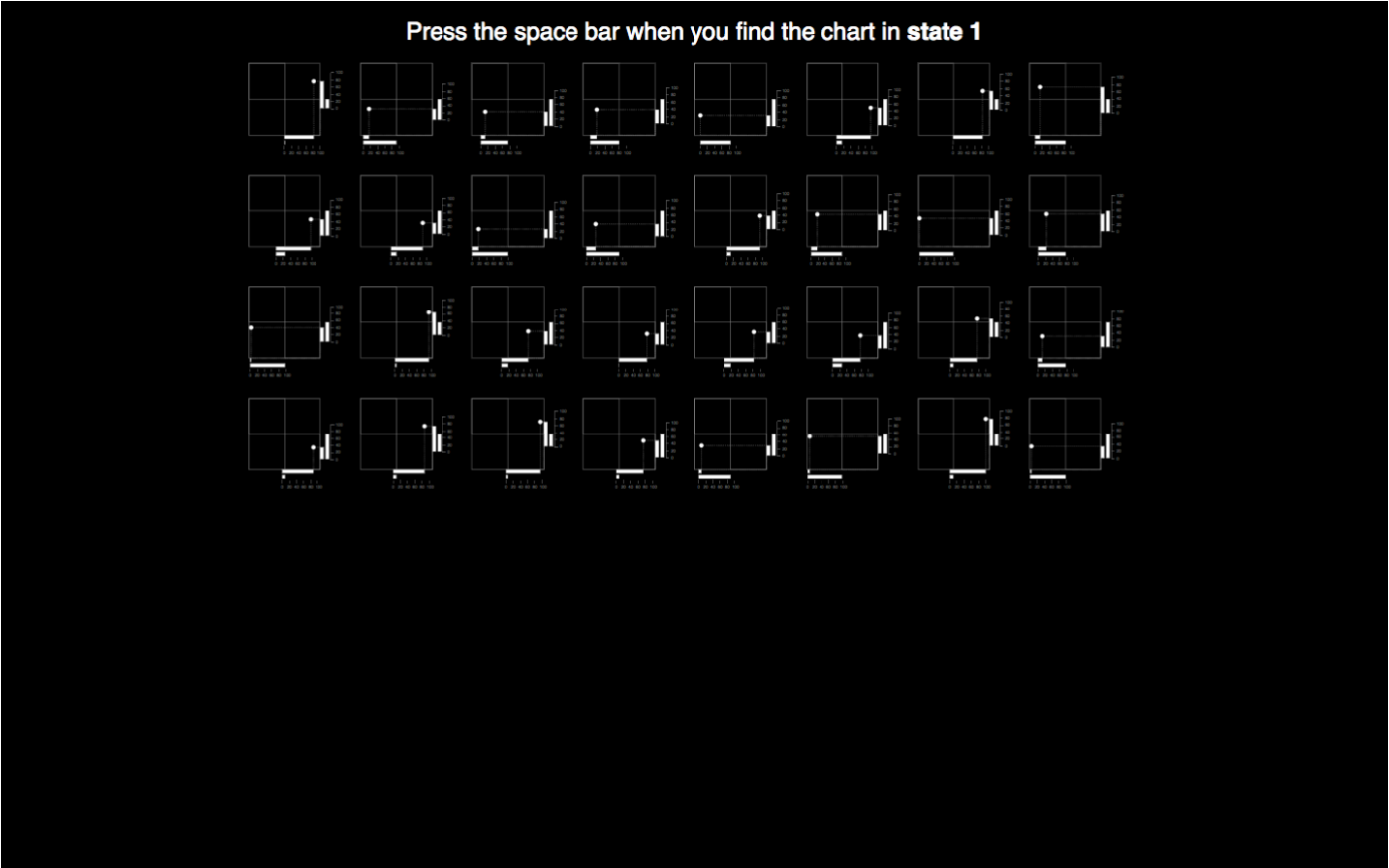
Appendix A, Figure 15: Configural coordinate display, Known Unique State Training, static, 8-display



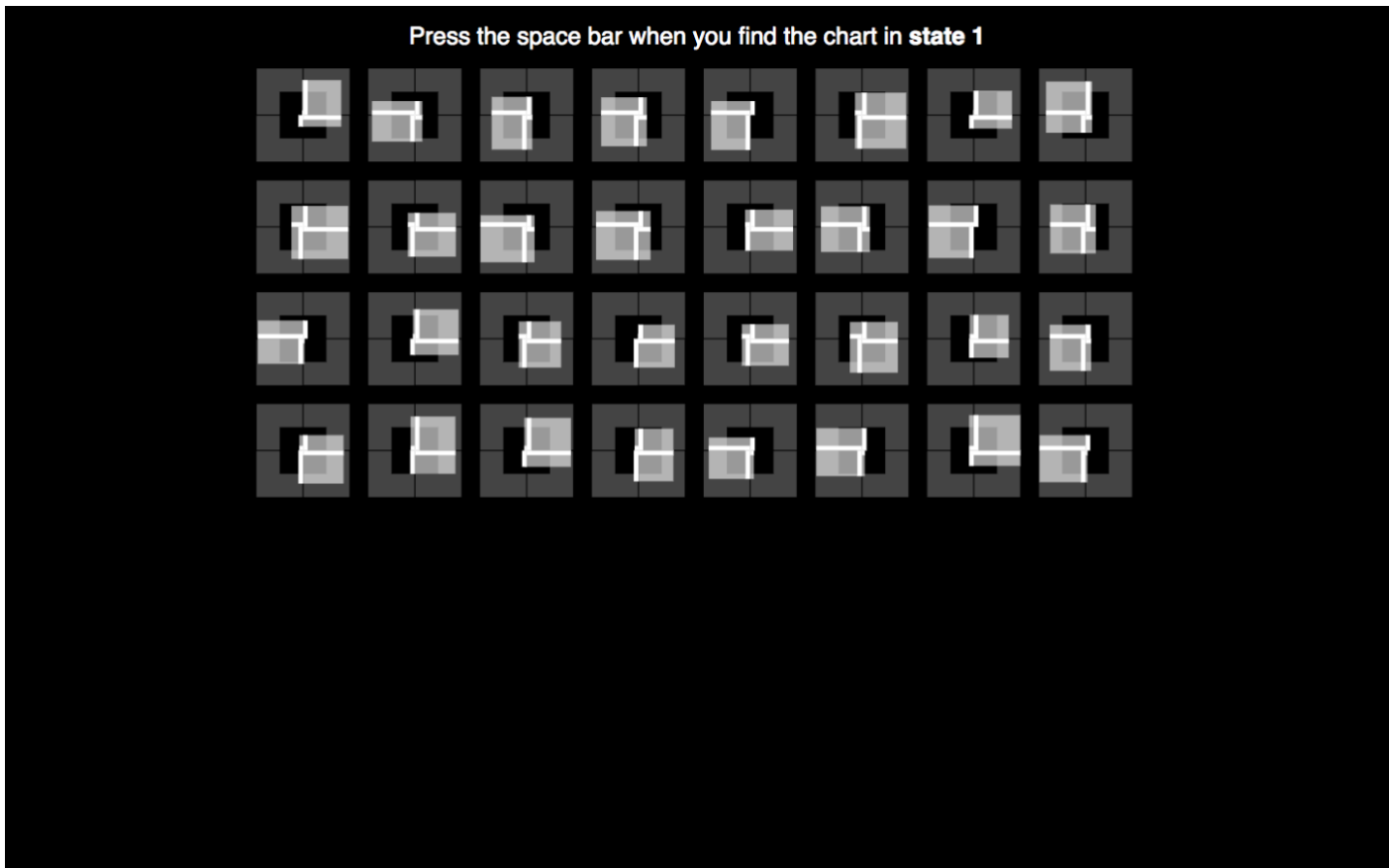
Appendix A, Figure 16: Configural integral display, Known Unique State Training, static, 8-display



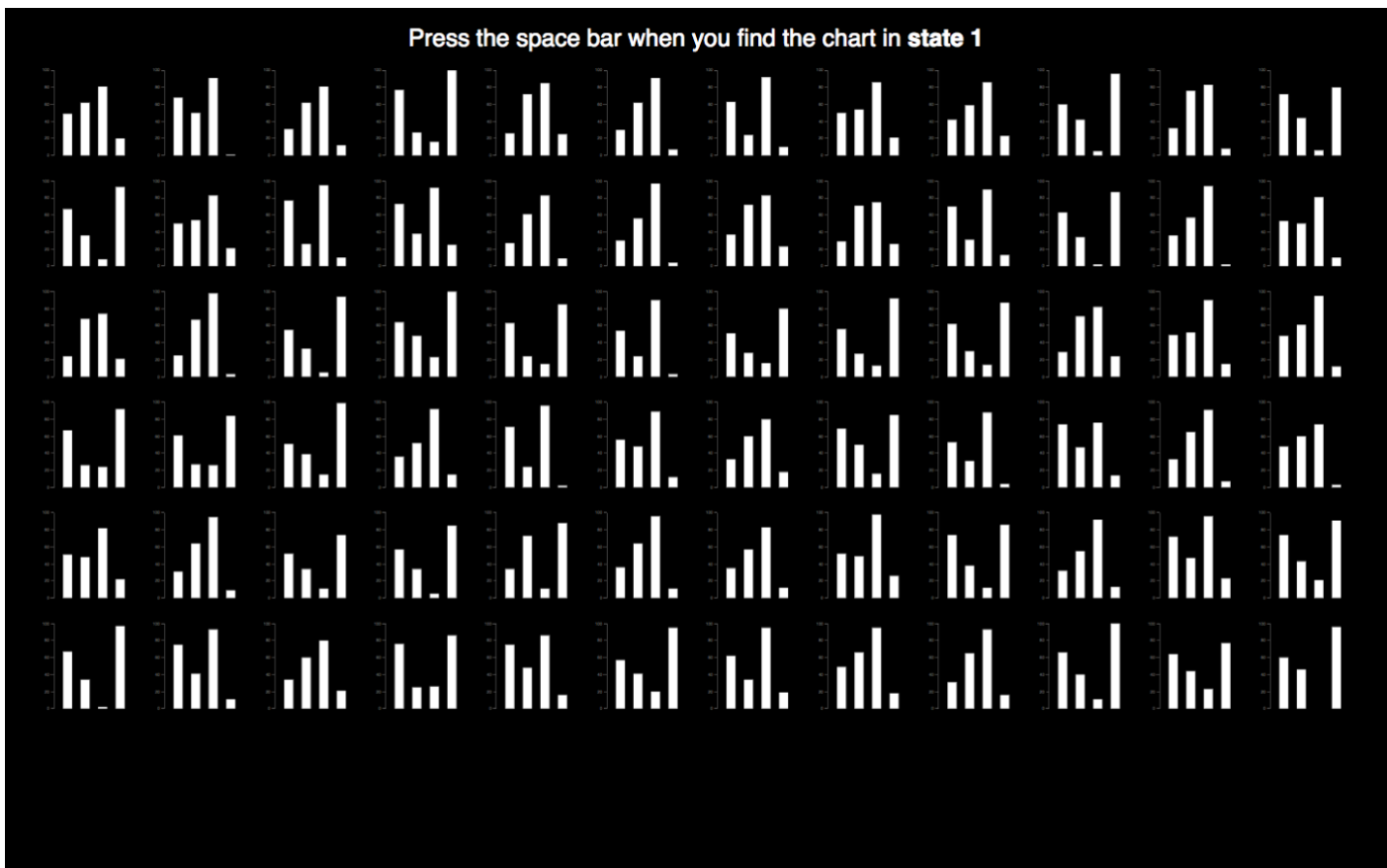
Appendix A, Figure 17: Bar graph display, Known Unique State Training, static, 32-display



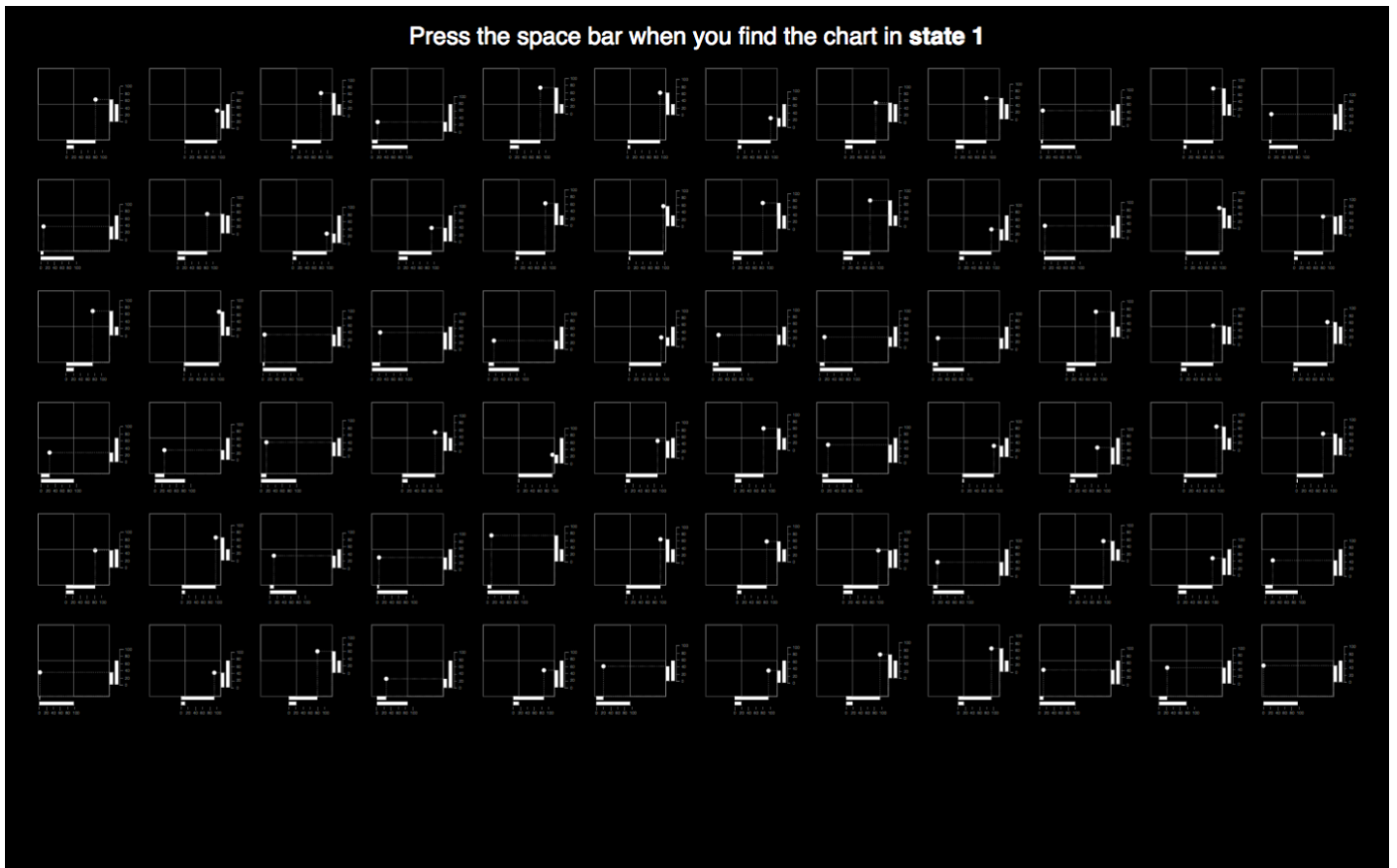
Appendix A, Figure 18: Configural coordinate display, Known Unique State Training, static, 32-display



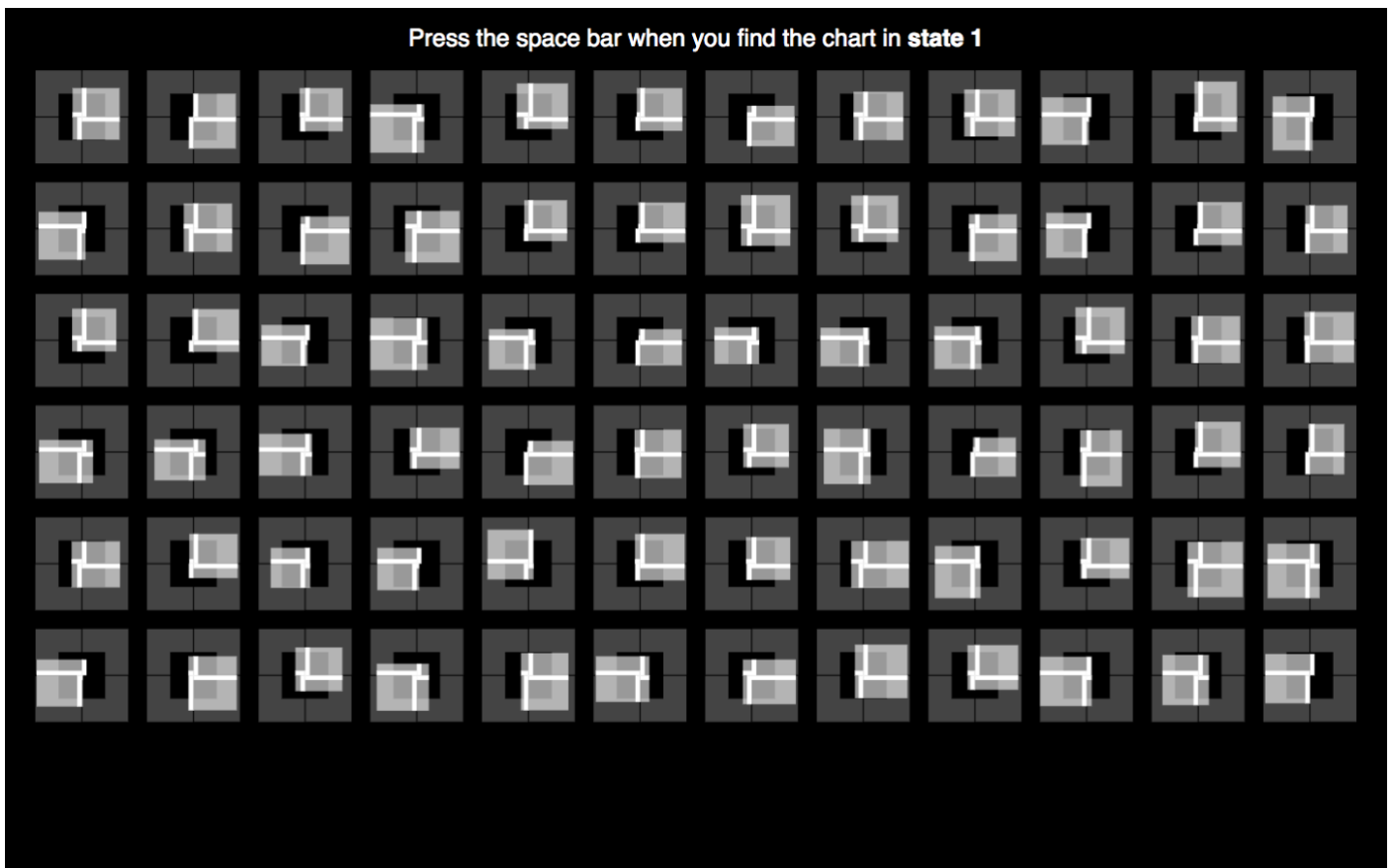
Appendix A, Figure 19: Configurational integral display, Known Unique State Training, static, 32-display



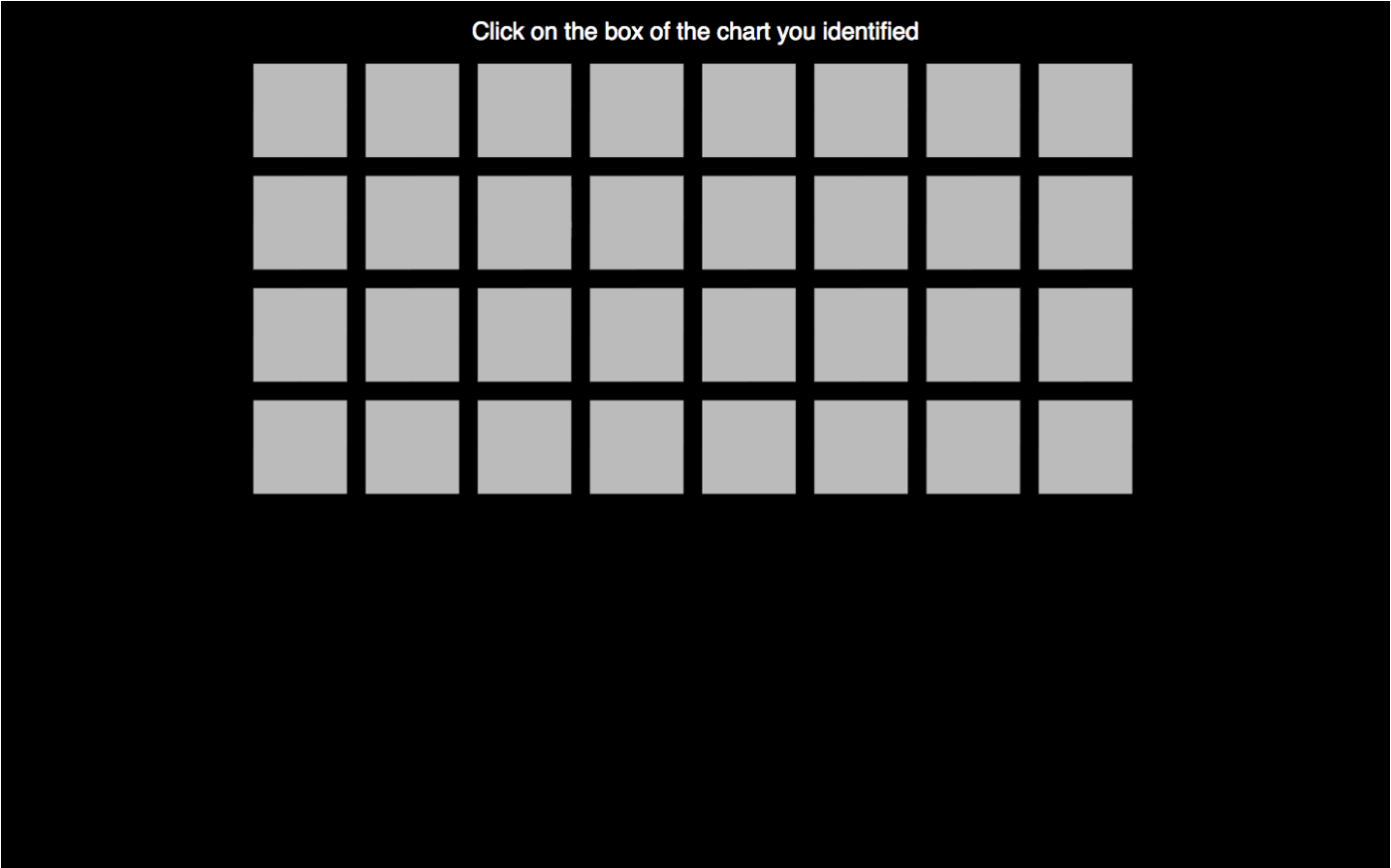
Appendix A, Figure 20: Bar graph display, Known Unique State Training, static, 72-display



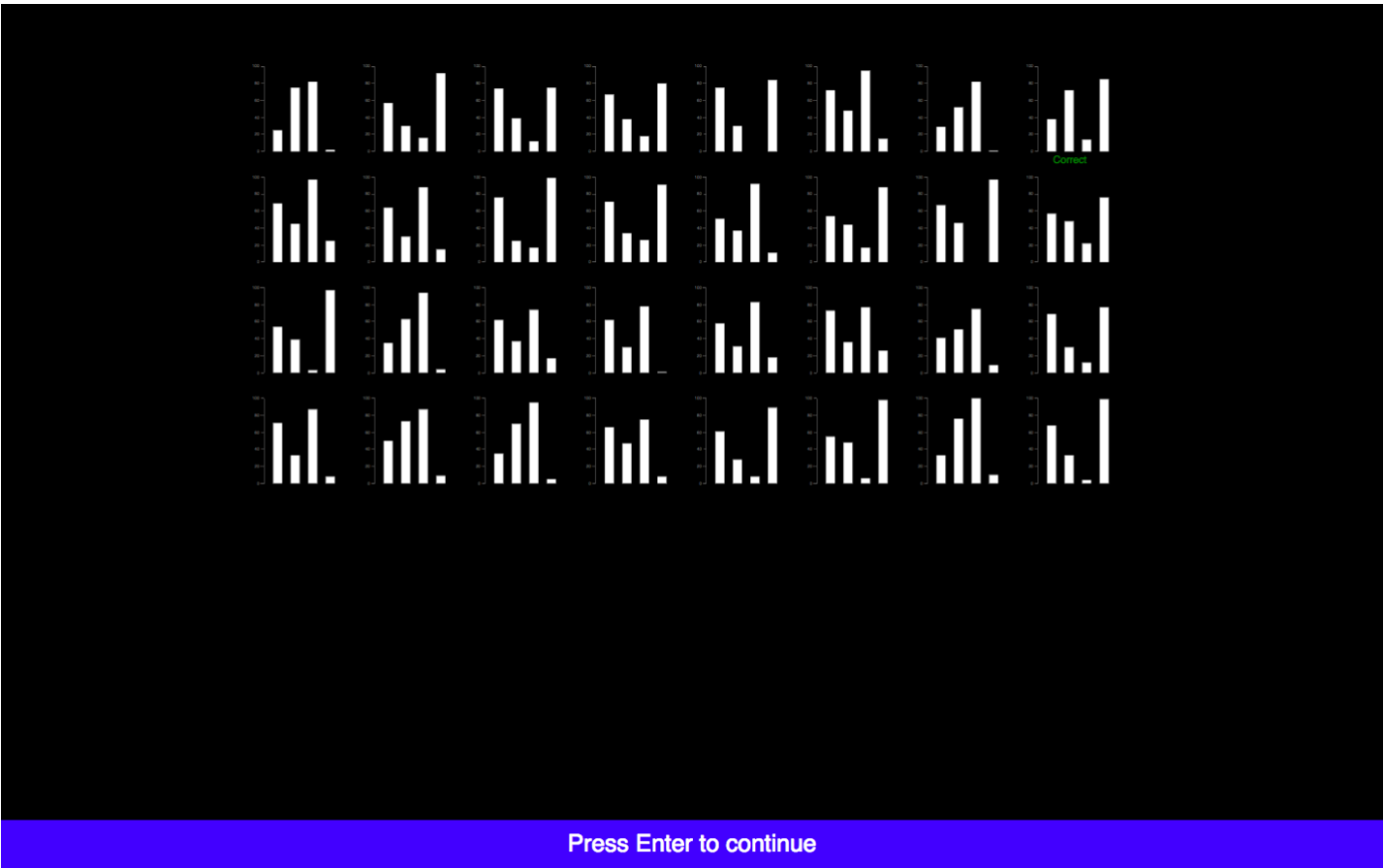
Appendix A, Figure 21: Configural coordinate display, Known Unique State Training, static, 72-display



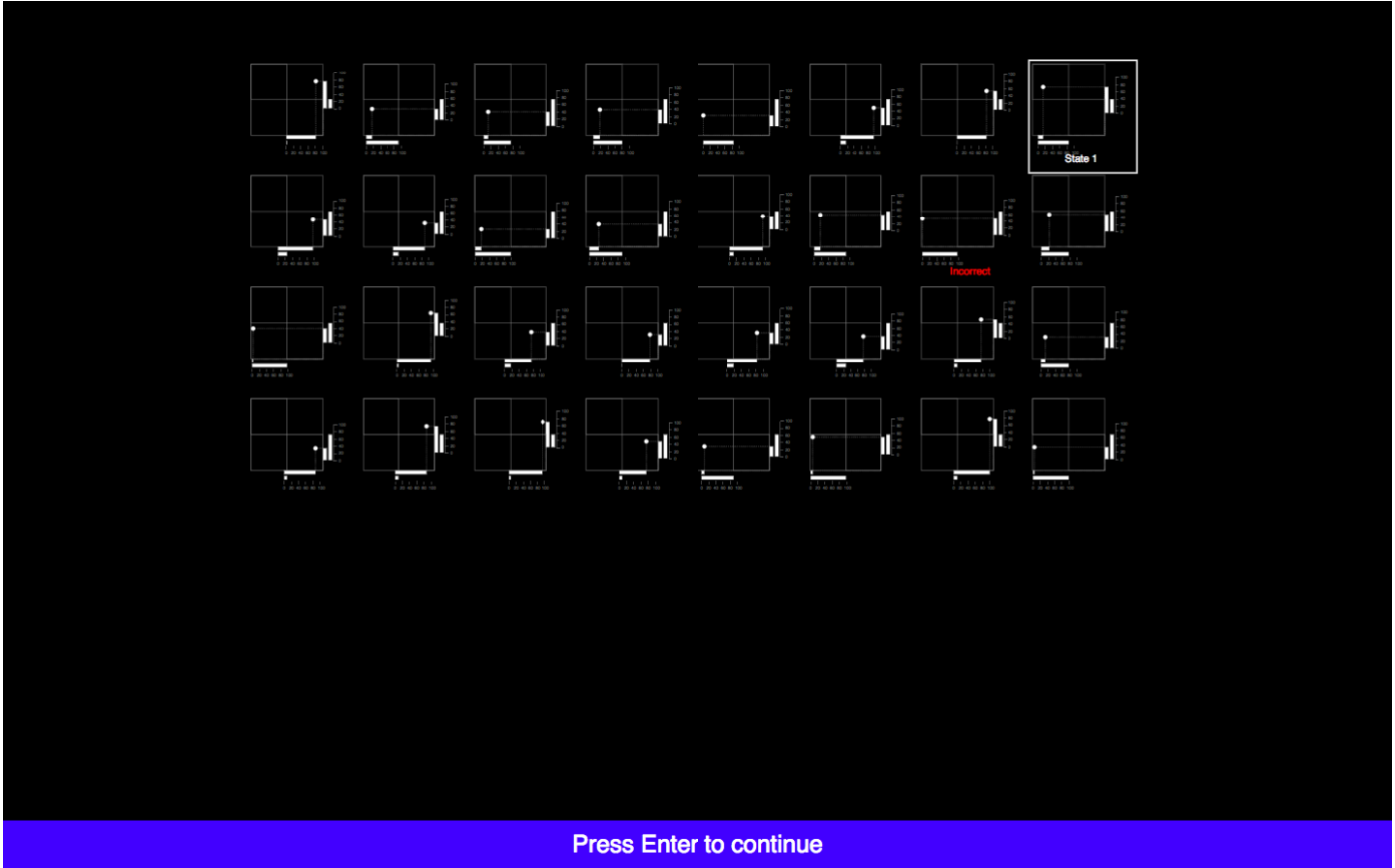
Appendix A, Figure 22: Configural integral display, Known Unique State Training, static, 72-display



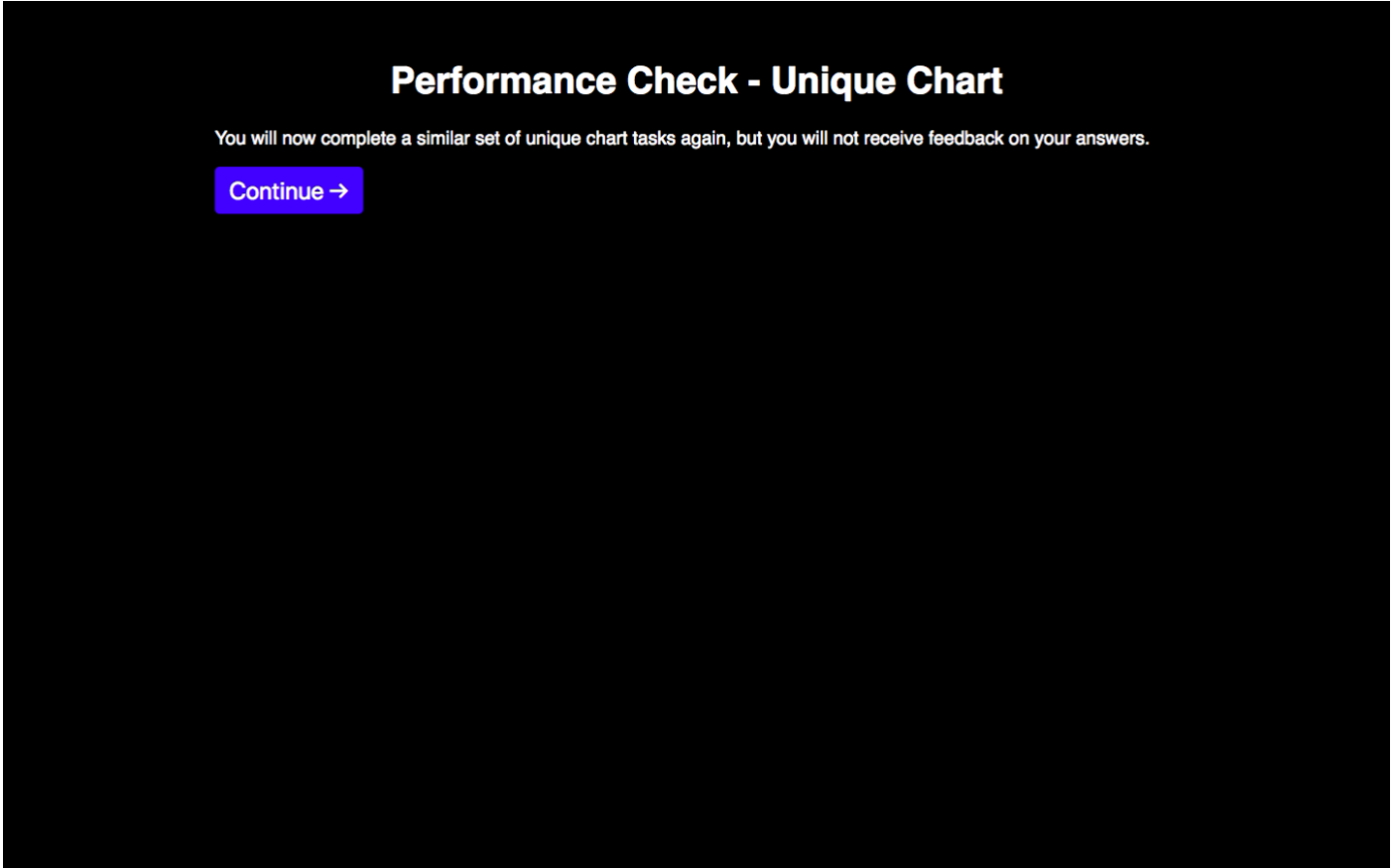
Appendix A, Figure 23: Example of screen with display overlays shown after space bar is pressed



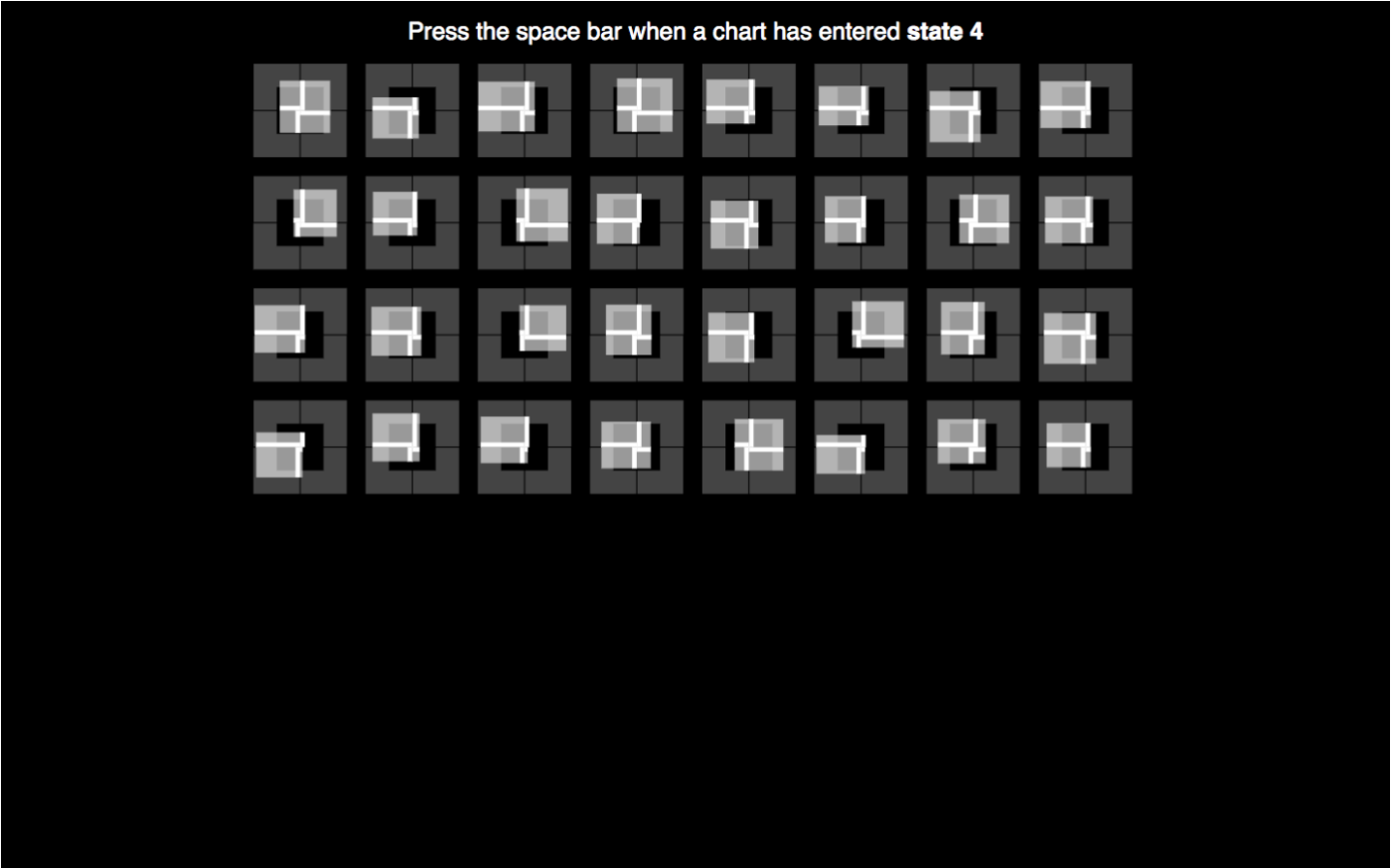
Appendix A, Figure 24: Known Unique State Training screen after the correct display overlay was clicked



Appendix A, Figure 25: Known Unique State Training screen after an incorrect display overlay was clicked



Appendix A, Figure 26: Known Unique State Performance Check instructions



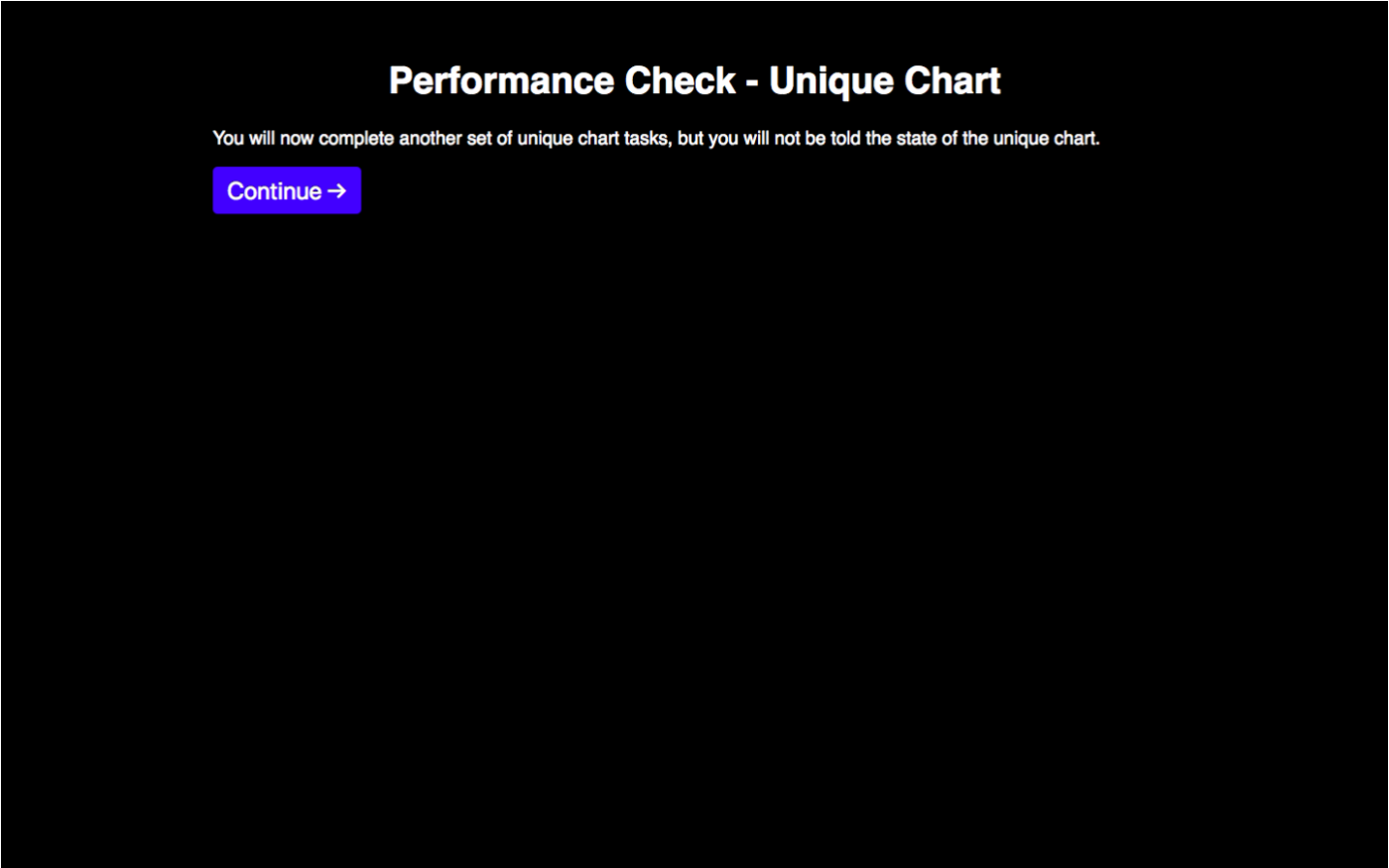
Appendix A, Figure 27: Configural integral display, Known Unique State Performance Check, dynamic, 32-display



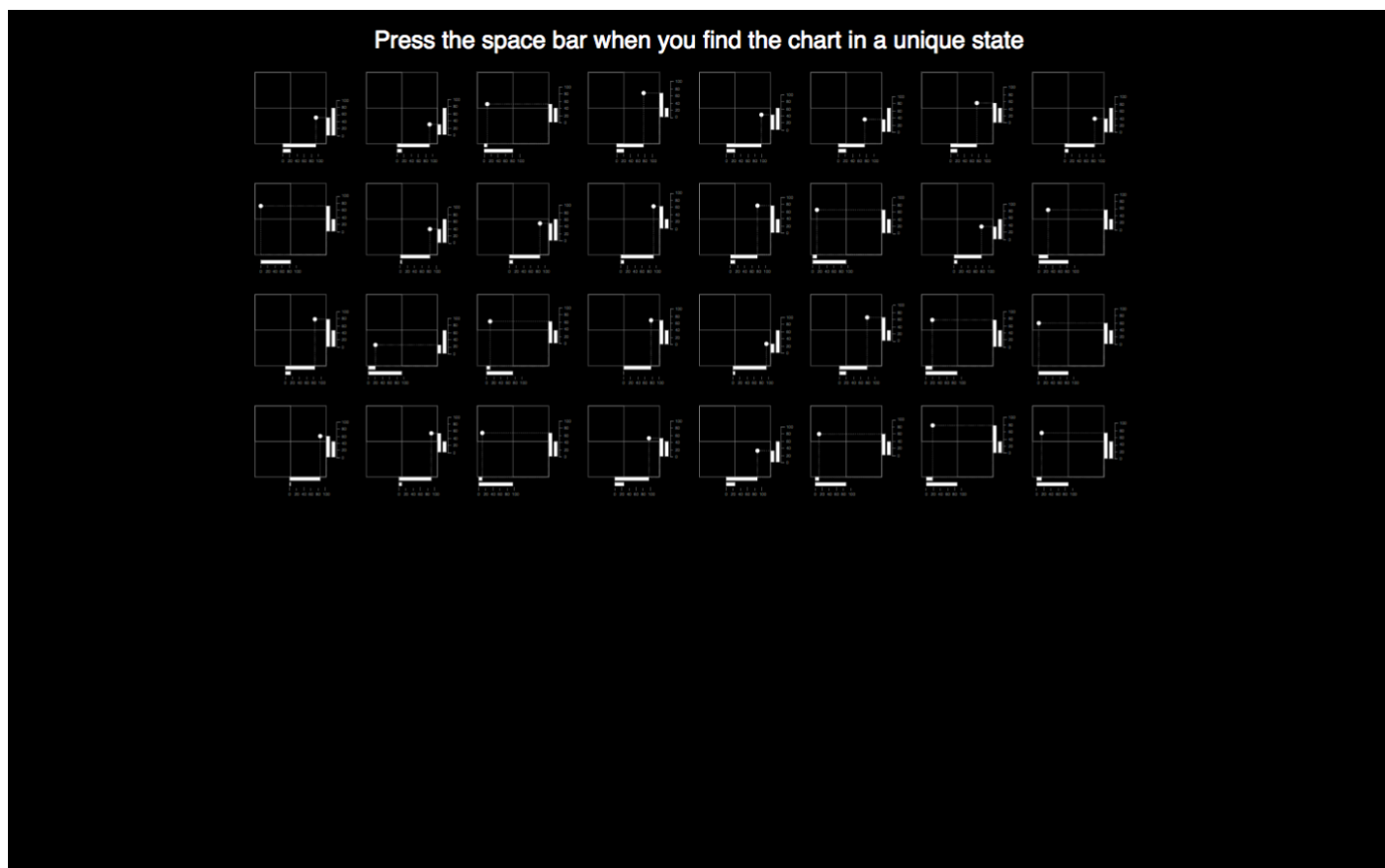
Appendix A, Figure 28: Configural integral display, Known Unique State Performance Check, dynamic, 32-display, correct



Appendix A, Figure 29: Configural integral display, Known Unique State Performance Check, dynamic, 32-display, incorrect



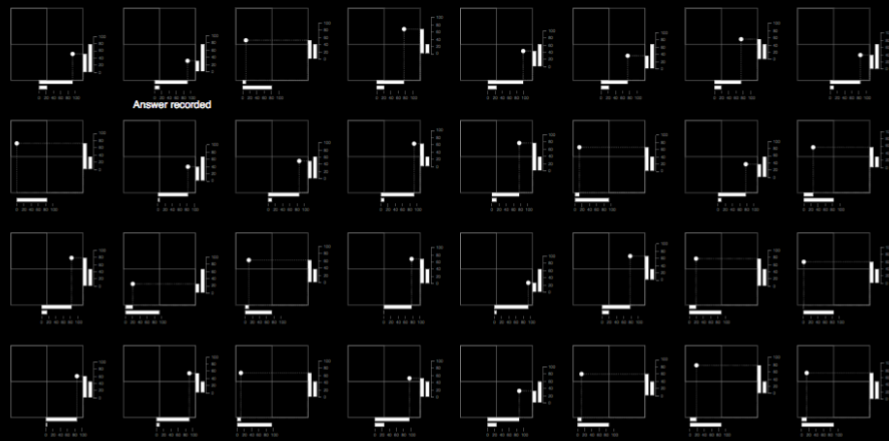
Appendix A, Figure 30: Unknown Unique State Performance Check instructions



Appendix A, Figure 31: Configural coordinate display, Unknown Unique State Performance Check, static, 32-display



Appendix A, Figure 32: Configurational coordinate display, Unknown Unique State Performance Check, static, 32-display, correct



Press Enter to continue

Appendix A, Figure 33: Configural coordinate display, Unknown Unique State Performance Check, static, 32-display, incorrect

Appendix B: Pre-Test Survey

Pre-Test Survey

Participant Number (on screen): _____

What is your gender?

- ☐ Male
- ☐ Female
- ☐ Other

What is your age?

Age: _____

In which country have you lived most of your life?

- ☐ Sweden
- ☐ Other: _____

How many years did you have English as a subject in school?

- ☐ 5 or fewer
- ☐ 6 to 10
- ☐ 11 or more

Are you currently studying?

- ☐ Yes, studying programme name: _____
- ☐ Yes, studying individual courses
- ☐ No, not currently studying

Thank you! Please click the button on screen to begin the test.

Appendix C: Post-Test Questionnaire

Post-Test Questionnaire

How confident are you that you identified the unique chart on the last four screens?

| | | | | |
|-----------------------|-----------------------|-----------------------|-----------------------|-----------------------|
| Not at all confident | Not very confident | Neutral | Confident | Very confident |
| <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |

What was your way of finding the unique chart on the last four screens?

Feel free to write or illustrate in your answer.