# Measuring and Modeling Pursuit Detection in Dynamic Visual Scenes

**Maria Kon (maria.kon.ctr@nrl.navy.mil)**
**Sangeet Khemlani (sangeet.khemlani@nrl.navy.mil)**
**Andrew Lovett (andrew.lovett@nrl.navy.mil)**
Navy Center for Applied Research in Artificial Intelligence
U.S. Naval Research Laboratory, Washington, DC 20375 USA

**Abstract**

Although we are generally good at observing a busy scene and determining whether it contains one agent pursuing another, we are not immune to making errors and may identify a pursuit when there is none. Further, we may have difficulty articulating exactly what information allowed us to determine whether there was a pursuit. To gain a better measure of when people correctly or erroneously detect pursuit, we designed a novel pursuit detection task. To compare performance given different strategies, we developed a cognitive model that can perform this task. The results of our pursuit detection experiment indicate that, indeed, people typically identify pursuit events correctly, but they make infrequent yet systematic errors for particular scenes. When the model implements specific strategies, simulation results are well correlated with empirical results. Moreover, the model makes the same errors as human participants. We show how the empirical results can be accounted for in terms of decision criteria indicated by high performing model strategies.

**Keywords:** pursuit detection; chasing; relations; dynamic scenes

## Introduction

To determine whether one object is pursuing another, people must track the objects over time and compare their locations. The psychological notion of *pursuit* can implicate intentionality: a cat intends to catch the mouse it pursues (Schultz & Frith, 2022). Thus, the perception of pursuit is critical to our understanding of the behavior of those around us. Perceptual failures can yield maladaptive behavior: for example, a 2022 CCTV video shows how a group of CrossFit athletes running through a bar's outdoor seating area caused mass panic among its patrons who, wrongly thinking that the athletes were running away from some danger, fled the bar and ran with the athletes (Bass, 2022). If we can better understand where failures in pursuit detection originate, we can anticipate such problems.

A large body of research has explored the features that support and hamper pursuit detection. Whereas early research used free response questions or animacy rating scales (e.g., Bassili, 1976; Dittrich & Lea, 1994; Heider & Simmel, 1944), recent pursuit studies used tasks designed to probe the perception of pursuit in dynamic stimuli (e.g., Meyerhoff, Huff, & Schwan, 2013). These latter studies sought to test various cues that impact how people identify a 'pursuer' object (the agent pursuing something) and a 'target' object (the agent being pursued): for instance, the number of objects in a scene (Gao et al., 2019; Meyerhoff et al., 2013), the direction that the pursuer faces (Gao, McCarthy, & Scholl, 2010; Gao, Newman, & Scholl, 2009), the degree to which the pursuer deviates from the most direct path to the target (Gao et al., 2019; Gao et al., 2009; Meyerhoff et al., 2013), and the distance between the pursuer and target (Meyerhoff, Schwan, & Huff, 2014a,b) can all affect detection of pursuit.

However, the tasks used in these studies have two limitations in their ability to identify factors contributing to success or failure in pursuit detection. (A) They did not isolate the pursuit detection task from other cognitively demanding tasks. For example, in a typical task, participants looked for pursuit among a field of identical, moving objects, resulting in potential difficulties with tracking the objects over time. (B) They did not provide a full picture of the error patterns produced by people; when participants incorrectly responded that pursuit was occurring, they were never asked to indicate which objects they believed were the pursuer and the target. In addition, many studies showed a complete video before asking participants to respond, limiting their ability to determine at what point in the video participants recognized that pursuit was occurring.

In this paper, we present a novel experiment designed to provide a better understanding of the factors that support or impair perception of pursuit. Additionally, we describe a computational model that explores the procedural steps undertaken to detect pursuit. Because human participants often cannot articulate the approach they used to complete a task, such a model has the potential to allow us to test pursuit detection strategies (Briggs et al., 2023; Kon & Francis, 2022, 2023). The results from the behavioral experiment are used to evaluate computational strategies and stopping rules and derive hypotheses about how people complete the task. Furthermore, the model makes predictions about: particular dynamic scenes that are more likely to induce errors in pursuit detection, and the type and timing of such errors.

## Experiment 1

This experiment tested two robust patterns of pursuit detection observed in previous work: people are faster to detect pursuit than its absence, and people are slower to detect pursuit as set size increases. It presented participants with a series of videos of colored circles moving on a black background. The videos depicted either 2 or 6 circles. Half the videos depicted a red circle pursuing a circle of another color ('pursuit present' trials), and the other half depicted no such pursuit ('pursuit absent' trials).

The experiment addressed the two limitations identified above. (A) To make it easier to re-identify and track the objects over time, the circles were all given different colors and the potential pursuer circle was always red. On each trial, participants first assessed whether the red circle was pursuing another circle by tapping an appropriate key, which caused



In L. K. Samuelson, S. L. Frank, M. Toneva, A. Mackey, & E. Hazeltine (Eds.), *Proceedings of the 46th Annual Conference of the Cognitive Science Society.* 

the video to pause. (B) To identify error patterns, after a participant indicated that there was pursuit, they were instructed to click on the target circle. They did this even on trials where they were incorrect, and no actual pursuit was occurring. (On trials where participants indicated an absence of pursuit, they used the mouse to click on any circle.)

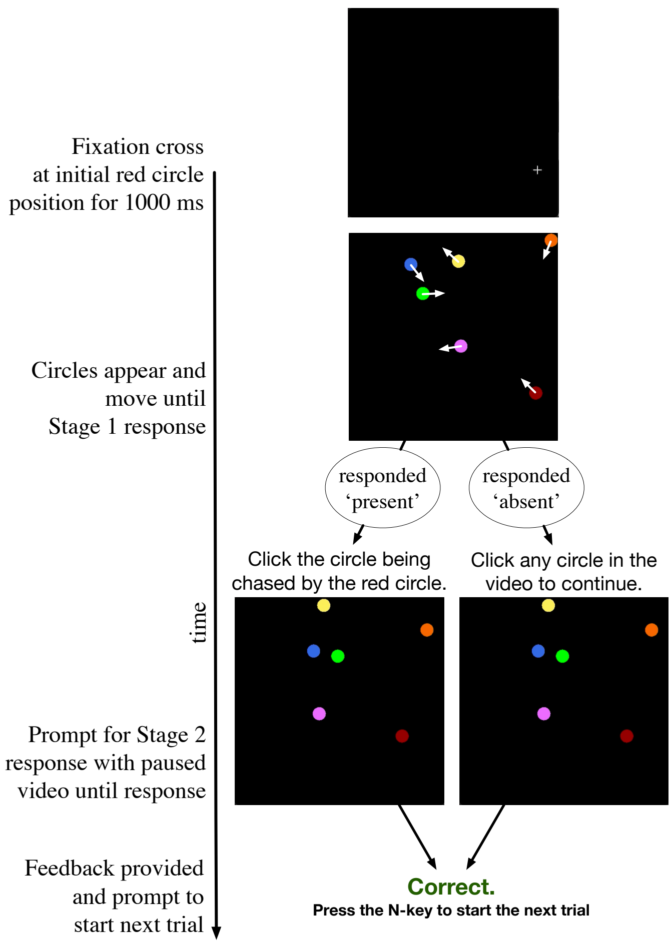


Figure 1: Stimulus sequence for an experimental trial with set size 6. After seeing a fixation cross that indicated the starting position of the red circle, participants saw a video of moving circles, which are indicated here by the white arrows. (Note that participants did not see such arrows; they are only included on the static image here in order to give the reader a sense of their motion. The entire video is here: `https://osf.io/49pra`.) After participants pressed a key to indicate whether they thought the red circle was pursuing another circle, the video paused and a prompt indicating the Stage 2 task appeared above. After they clicked on a circle, feedback and a prompt to initiate the next trial were provided.

## Method

**Participants** 99 participants (mean age = 42.05 years; 46 females, 52 males, 1 prefer not to answer) completed the study on the Amazon Mechanical Turk online platform in exchange for US$2.50. All but one participant reported normal color vision; the participant who reported suspected colorblindness yielded 97% accurate responses, so we retained their data. Another participant yielded low accuracy data (61% accurate responses, $>2$ standard deviations from pooled mean accuracy). We analyzed the remaining data from $n = 98$ participants.

**Materials, Procedure and Design** The study was developed using custom JavaScript and HTML code within the `nodus-ponens` package (Khemlani, 2022). Figure 1 provides a schematic example of the videos participants saw in a trial. A trial began with a black square and a fixation cross to highlight where the pursuer circle would be in a video that would begin 1000 ms later. The video showed 2 or 6 uniquely colored circles moving at the same fixed speed for 22 seconds. Half the videos were pursuit-present, half pursuit-absent, and the videos varied the color of the target circle. Circles could take on one of five separate colors that came from a colorblind-friendly palette, and so materials consisted of 20 separate videos. The experiment code and all videos are available here: `https://osf.io/65ezq/`.

For each video, the study assigned circles a random initial direction and position such that no circle overlapped with another. Circles bounced off the edges of the square. If one circle contacted another, it continued on its trajectory and passed through that circle. Every 50 ms, the red circle updated direction such that it went towards the present location of the target circle. The red circle was visible for the entirety of the video, i.e., it overlapped all other circles on contact. The remaining circles were drawn on randomized layers, but all videos depicting pursuit contained no frame in which the target circle was completely covered by a distractor. Likewise, no video showed the red circle contacting its target.

After navigating to the webpage with the study, participants saw instructions that described the task, informed them that the circle in pursuit was always red, described the role of the fixation cross, and presented a few example trials. After the participants pressed a key to initiate a trial and the fixation cross appeared, circles then appeared and moved until participants pressed a key indicating the presence or absence of pursuit (Stage 1 response). For the duration of the circles' movement, text remained on the screen that reminded participants of the response key assignments. Participants responded using the 'F'- and 'J'-keys on their keyboard to indicate whether a pursuit was present or absent; key assignment was counterbalanced across participants. The video paused on a key press and the study then prompted participants to click a circle on the screen (Stage 2 response). They used their mouse to click on the target circle for 'present' responses, and to click on any circle in the video for 'absent' responses. After they clicked, the display became black and feedback appeared on the screen. They received feedback for all trials ("Correct."; "Incorrect - red was chasing."; "Incorrect - no chasing."; "Incorrect circle.") as well as notifications if their Stage 1 response was too fast ($<500$ ms) or too slow ($>20$ s). We dropped trials with reaction times that were too fast or too slow from analysis.

Participants received 4 practice trials and then carried out

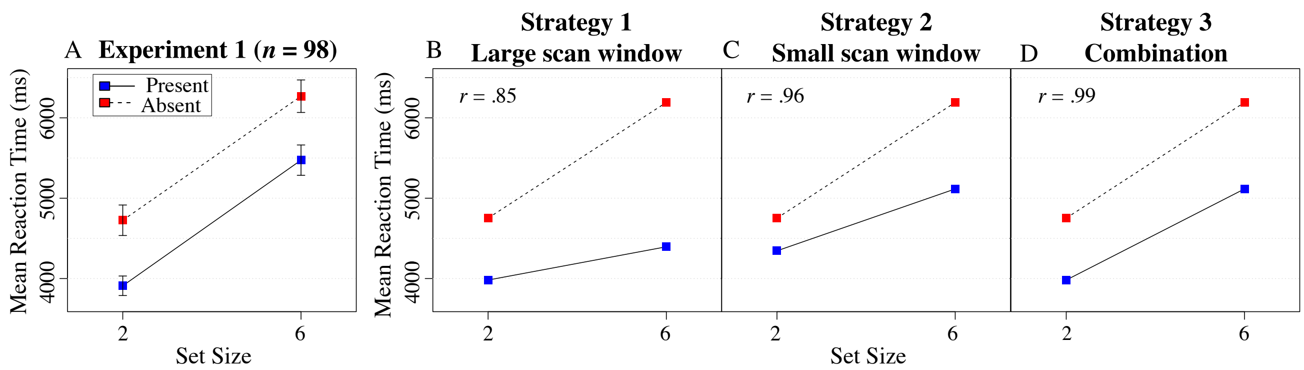


Figure 2: Plot A: Mean reaction times for Experiment 1. Plots B-D: Mean model reaction time for three different strategies, with correlations between the data and model results for each strategy. All response times were calculated from correct trials. Error bars represent one standard error of the mean.

32 experimental trials in randomized order. The experiment yielded a 2 (pursuit present vs. absent) *x* 2 (2 circles vs. 6 circles) repeated-measures design, with those 4 total conditions repeating 8 times. Across the 8 repetitions, the target circle's color was selected from 5 possible values by rotating values across participants using a Latin square.

## Results and Discussion

We ran three ANOVA models in R (version 4.3.1; R Core Team, 2023) using the ez package (Lawrence, 2016), one that analyzed participants' Stage 1 accuracy, one that analyzed Stage 2 accuracy, and one that analyzed reaction times.

Participants were largely accurate for each individual condition, i.e., they were accurate on 94% of the trials. However, the data revealed slight but reliable main effects, e.g., participants were less accurate in assessing the presence of pursuits when multiple distractors were present. (For brevity, we focus our analyses on reaction time, but full analyses are available at `https://osf.io/c2dju`.)

The ANOVA on reaction times, which was run for correct trials only (see Figure 2A for mean reaction times of these trials), showed that participants were significantly slower for absent trials than present trials, $F(1,97) = 61.28$, $p < .001$, $\eta_p{}^2 = .050$. They were also slower when more objects were in the display, $F(1,97) = 371.52$, $p < .001$, $\eta_p{}^2 = .164$. The interaction between the present/absent condition and set size was not significant, $F(1,97) = 0.03$, $p = .871$, $\eta_p{}^2 < .001$.

These results suggest that the greater the number of objects in a scene, the longer it takes to detect whether there is a pursuit, which replicates previous findings (Meyerhoff et al., 2013). Likewise, as the number of objects in a scene increases, participants are more likely to misjudge whether there is a pursuit and misidentify the target circle. These results establish benchmark patterns in pursuit detection: reaction times increase as set size increases, and reaction times tend to be greater for absent than present trials.

A key strength of this experimental paradigm is that it can provide detailed information on human error patterns. Due to space constraints, we focus on the errors for one particular video (hereafter called 'video 1' and provided at `https://osf.io/52yuv`). This was a pursuit-absent trial where a disproportionate number of participants incorrectly responded 'present' in Stage 1. In fact, this video received 45% of the total incorrect 'present' responses across all videos. The Stage 2 responses to this video, in which participants indicated what target they falsely believed the red circle was pursuing, are depicted in the left graph of Figure 3A. As the graph indicates, nearly all participants believed the red circle was pursing the green circle. On reviewing video 1, which had 6 circles, we believe that the red circle could be interpreted as tracking the green circle from about the 4.5 second mark until the 7 second mark, after which time it heads away from the green circle. Notably, the mean Stage 1 reaction time for these incorrect trials was $M = 5.73$, $SD = 2.06$, which was during that window of time when the red circled appeared to be pursuing the green circle, whereas the mean time for correct responses was $M = 7.29$, $SD = 2.77$, after it may have become apparent that there was no pursuit.

As video 1 shows, systematic errors in pursuit perception may occur at various points of a sequence of movements. If errors are robust, then a cognitive model of pursuit detection should be able to simulate them. We turn to the development of a computational model that explains both benchmark patterns of pursuit detection as well as errors.

# Model Simulations

We developed a computational model to simulate the processes that underlie pursuit detection behavior. A core idea of this model is that perception depends on directing spatial attention to task-relevant locations in a scene (Posner, 1980; Ullman, 1984). For example, to determine where an object is going, you attend to the object and then scan forward along the object's path, allowing your attention to be drawn to any other objects that overlap this scan pattern. To simulate the relationship between perception and attention, we used ARCADIA (Bridewell & Bello, 2016), a computational platform designed to explore attention's role in perception, cognition, and action. ARCADIA has been used to model several other attentionally demanding visual tasks, including multiple object tracking (Lovett, Bridwell, & Bello, 2019) and enumeration (Briggs et al., 2023).

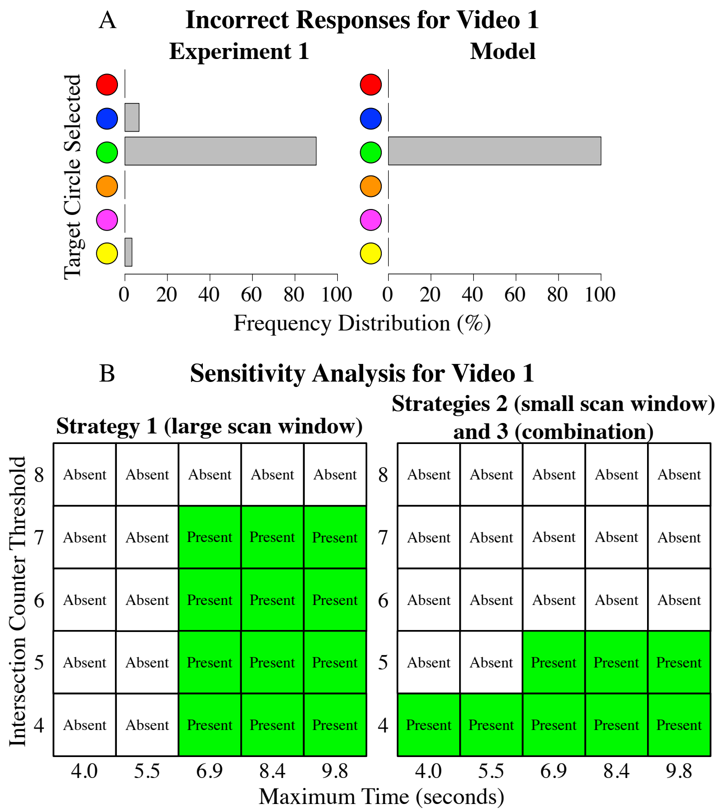


Figure 3: A: Percent of trials where a particular colored circle was selected as the target after observers incorrectly responded 'present' on a difficult pursuit-absent video ('video 1'). B: Sensitivity analysis showing, for each strategy, what pairs of parameter values resulted in correct 'Absent' responses or incorrect 'Present' responses for video 1. The green coloring indicates that the model always selected the green circle as the target after an incorrect 'Present' response.

Our objective in developing the model was to outline a general strategy for detecting pursuit, while providing parameters to tweak its strategy and stopping rules, in order to explore the space of possible human strategies. The general strategy works as follows. 1) Pick out the red circle, track it as it moves, and determine its trajectory. 2) Scan forward along that trajectory to determine whether the red circle is moving towards another circle. 3) If the scan intersects another circle, increment an *intersection count* for that circle. 4) Repeat steps 1-3. Over time, this strategy builds up a representation of how often the red circle appears to be moving towards each other circle (how often the scan along the red circle's trajectory intersects that circle). If the red circle is frequently seen moving towards another particular circle, then that circle is likely the one being pursued.

The model's general strategy is accompanied by three stopping rules indicating when to respond 'present' or 'absent' on a trial. 1) If the intersection count for any circle exceeds a threshold, respond 'present'. That circle is likely being pursued by the red circle. This means the model will respond 'present' more quickly if the red circle is consistently seen moving towards another circle. 2) If a sufficient time has passed without responding 'present', then respond 'absent'. This rule captures ending the task because there isn't sufficient evidence that pursuit is occurring. This type of 'absent' response will always take longer than a 'present' response. 3) If, after observing a video for a short time, there are no intersections observed at all, then respond 'absent'. This rule provides a means to short-circuit the task, producing a fast 'absent' response when there is no evidence of pursuit.

The pursuit model is implemented on the ARCADIA platform. Further details about ARCADIA are available elsewhere (e.g., Briggs et al., 2023), but we provide a high-level description here. ARCADIA models consist of sensors, components (e.g., see Table 1), and an attentional strategy. On each cycle of processing, ARCADIA components operate in parallel, taking data from the sensors, processing information, and producing output. Afterwards, a single focus of attention is selected from the output of all the components, according to the model-specific attentional strategy; this strategy is a set of priorities indicating what types of elements should receive the focus of attention for the modeled task. Finally, the focus of attention, as well as the output from other components, is made available to all components as input on the following cycle.

Table 1: Summary of model components

| Component | Function | Support |
|---|---|---|
| image segmenter | figure-ground segmentation | Palmer & Rock, 1994 |
| color highlighter | pop out effect | Wolfe & Horowitz, 2004 |
| trajectory scanner | scan along trajectory | Gerstenberg et al., 2017 |
| scan highlighter | focus on scan | Bello et al., 2018 |
| object file binder | feature integration | Treisman & Gelade, 1980 |

An example from the pursuit model can illustrate ARCADIA's operations (Figure 4, and see `https://osf.io/65ezq/` for simulation output for all videos). Here, a sensor grabs a still frame from the pursuit video on each cycle. Thus, each cycle of processing corresponds to one frame of video, in this case 18 ms of real time. An **image segmenter** component takes this image and segments out any possible figures, like the six colored circles shown in Figure 4. Next, a **color highlighter** component identifies the color of each circle. Because this model's attentional strategy prioritizes focusing on objects that have been classified as "red," the red circle becomes the focus of attention (indicated by the small orange box on this circle in Figure 4, during cycle A), allowing information about its motion trajectory to be generated. When sufficient information is available to calculate the red circle's trajectory (represented by the cyan line in Figure 4), a **trajectory scanner** component projects a small window

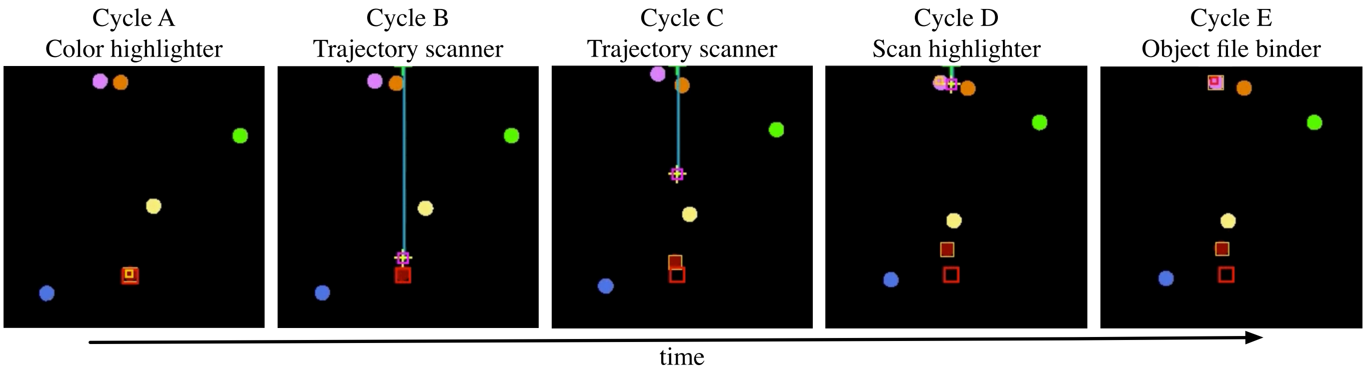


Figure 4: Example of model performance. The relevant model component for each step is listed above the images.

forward (represented by the magenta box in Figure 4) along this trajectory. If that window intersects another circle (Figure 4, cycle D), a **scan highlighter** component suggests it as a candidate for the focus of attention. Finally, because the attentional strategy prioritizes objects that have intersected the scan even above "red" objects, the intersected circle becomes the focus of attention, causing an **object file binder** to generate a representation of this object (Figure 4, cycle E), and causing its intersection count to be incremented. See Table 1 for a summary of these components.

The model has several free parameters that can be adjusted to change its strategy and stopping rules. The first, *scan window size*, is how large the scan window is. There is some simulation-based support for the claim that scan window size is part of a task-dependent strategy for optimizing performance (e.g., Kon & Francis, 2022). The larger the window is, the more likely it is to intersect a circle near the scan path, whether that be the actual pursuit target or a distractor. For example, the size of the scan window in the image from cycle D in Figure 4 is large enough to intersect with the pursued pink circle, thereby making that circle a possible focus of attention. However, it is small enough that it does not intersect the orange circle distractor.

The remaining free parameters relate to the stopping rules. The *intersection counter threshold* is the number of times the scan must intersect a circle before the model responds 'present'. Increasing this threshold means the model must gather more evidence before responding 'present', and it is more likely to run out of time and respond 'absent'. This parameter interacts with *maximum time*, the amount of time that must pass without responding 'present' before the model will respond 'absent'. The final stopping rule, for generating quick 'absent' responses, depends on two parameters: the *initial intersection threshold* is the minimum number of intersections that must be counted for a circle to prevent a quick 'absent' response, and the *initial time check* is the time at which that threshold is checked. For the present simulation, we kept these final two parameters constant while varying the others. The *initial intersection threshold* was 1, and the *initial check time* was 185 cycles, or about 3.3 seconds, meaning that after 3.3 seconds, if the model had not detected at least one intersection, it gave a quick 'absent' response.

By varying three of the model's free parameters, we were able to explore a space of possible strategies. The model could be more or less sensitive to objects that appear near the red circle's trajectory (*scan window size*), it could be faster or slower to respond 'present' (*intersection counter threshold*), and it could be faster or slower to abandon the trial and respond 'absent' (*maximum time*).

## Simulation Stimuli, Method and Procedure

We ran the model on the same set of videos shown to participants in Experiment 1. Recall that the videos covered 2 (pursuit present vs. absent) *x* 2 (2 circles vs. 6 circles) *x* 5 (target color) conditions, for 20 total videos. Because the model's behavior is non-random, there was no need to repeat some videos to produce the 32 trials provided to humans.

To see how the model performed with different strategies, we ran simulations with three different scan window size strategies. Strategy 1 ('large scan window') used a large-sized scan window, which had a diameter that was the same as one of the circles. Strategy 2 ('small scan window') used a small scan window (0.6 times the size of a circle diameter, and shown in Figure 4). Notably, these two strategies may be overly simplistic: they assume that participants utilize an identical strategy regardless of the stimuli or task difficulty. In reality, when there is a large number of distractors, a smaller window may help to eliminate noise from nearby objects; whereas when no distractors are present, a larger window may allow one to determine whether the pursuer is approaching the target more quickly. To capture this possibility, Strategy 3 ('combination') implemented a hybrid policy, using a large scan window for set size 2, but a small window for set size 6.

To examine the impact of stopping rule parameter values on performance, we also conducted sensitivity analyses where we varied stopping rule parameter values. The *maximum time* ranged from 280 to 600 cycles from video onset in increments of 80 (which is approximately 4.0, 5.5, 6.9, 8.4, or 9.8 seconds after circle onset), and the *intersection counter threshold* ranged from 4 to 8 in integer increments. This analysis was conducted for each window size strategy, and it allowed us to determine the impact of different stopping parameters for each strategy.

### Model Results and Discussion

Figure 2 provides a side-by-side comparison of the Experiment 1 reaction time results with those of the model simulations for each of the three window size strategies. We conducted several simulations where we systematically varied the stopping rule parameter values, but the model results shown in Figure 2 were for simulations with the following parameter values: *intersection counter threshold* = 8, *maximum time* = 440 cycles (or about 6.9 seconds). These values were chosen to facilitate comparison across strategies as they produced the best performance (no errors and lowest response times) across the three strategies.

Mean response time by condition for each strategy is shown in Figure 2B-D. For all strategies, response time is in accord with the benchmark pattern, with response times tending to be longer for the higher set size and for the absent condition. However, when the model implements the combination strategy, the results have the highest correlation with the data (correlation values are listed in Figure 2B-D).

Notably, response time for absent trials was constant across the three strategies. This was because correct absent trials always took time equal to either *initial time check* (meaning a fast 'absent' response due to no intersections with any circle early in the video) or *maximum time* (meaning a slow 'absent' response after failing to detect pursuit), and these two parameters were the same across the three models. However, the response times for present trials varied between strategies. Under the large scan window strategy, the model generated fast 'present' responses because its larger window resulted in more intersections with potential targets. Under the small scan window strategy, the model generated slower 'present' responses—it took more time for evidence to accumulate because the smaller scan window resulted in fewer intersections. Finally, under the combination strategy, which adjusted the window size, the model generated fast 'present' response when there were no distractors (set size 2), but slower responses when distractors were present (set size 6). Essentially, the combination strategy was more cautious in cases where there was a higher risk of encountering distractors. In doing so, it produced the best fit to the human data.

We returned to the particularly difficult video 1 to explore whether the model could explain human performance on that video. Given the parameters selected for the comparison to humans in Figure 2, the model correctly responded 'absent' on this video. However, as we varied the stopping rule parameters (*intersection counter threshold* and *maximum time*), a different pattern emerged (Figure 3B). Consider Strategy 3 ('combination strategy'), which had the best fit to human data. When *intersection counter threshold* was high (meaning the model required a large amount of evidence to respond 'present'), the model correctly indicated that there was no pursuit, but when this parameter was lower (meaning less evidence was needed), the model incorrectly responded 'present'. Critically, when it did so, it indicated that the green circle was being pursued, just as human observers had done (Figure 3A, left plot). The timing of its response (around 5.6 s) was also aligned with that of humans observers measured in Experiment 1, which was during the window of time when the red circle appeared to be moving towards the green. Thus, the model demonstrated how an error could emerge when individuals are not careful to gather sufficient evidence before responding 'present'.

## General Discussion

Pursuit detection has been recognized as an important area of study, as it enables inferences about the intent of humans and other animals. However, many questions remain about the factors that contribute to success or failure. To better answer those questions, this paper presents a novel experimental design for studying pursuit detection. This design isolates the task from other demanding perceptual tasks, such as object tracking, while providing information about the timecourse of perception and the error patterns that emerge when the task is difficult. The empirical results replicate typical findings from previous studies, while allowing us to study difficult problems in ways that have not been possible in the past.

The novel experiment is complemented by a computational model, developed based on the idea that perception requires directing spatial attention to task-relevant locations in a scene. When it implements particular strategies, the model is able to achieve a close fit to human data, and it demonstrates how people can be influenced by brief periods of apparent but false pursuit, particularly when their decision criterion is not strict enough.

Why study errors in pursuit detection if people were largely accurate in their responses? Some of these errors happen systematically. Study of such systematic errors can not only help provide insights into ways in which pursuit detection breaks down, but it can also provide clues about what features we use to detect pursuit successfully. For example, in some cases where people wrongly detect pursuit, one object is frequently in front of the proposed pursuer for a short time. Thus, the identification of the type and timing of errors made possible by our novel experiment combined with how the model makes similar errors provides support for the claim that one cue to pursuit detection is an object being in front of the pursuer frequently.

The computational model allows us to develop a theory of pursuit perception beyond the stimuli and the videos we tested in the experiment. For instance, by conducting an analysis across a wide range of parameter values and strategies, we are able to derive a novel, testable hypothesis: participants adjust their pursuit detection strategy based on the difficulty of a task, notably focusing attention within a narrower area when there is a large number of distracting objects in a scene. Future research will allow us to evaluate this hypothesis and refine our understanding of how people perceive pursuit.

## Acknowledgments

This work was supported by a National Research Council Associateship Award to MK, and data were collected by Knexus Research Corporation. This work was also supported by a grant from the Naval Research Laboratory awarded to SK. The views expressed in this paper are solely the authors' and should not be taken to reflect any official policy or position of the United States Government or the Department of Defense.

## References

Bass, C. (2022, September 27). *Terrified diners flee busy restaurant – but not all is as it seems*. Yahoo News Australia. https://au.news.yahoo.com/runners-mistaken-robbers-terrified-diners-run-080038477.html

Bassili, J. (1976). Temporal and spatial contingencies in the perception of social events. *Journal of Personality and Social Psychology*, *33*, 680–685.

Bello, P., Lovett, A. M., Briggs, G., & O'Neill, K. (2018). An Attention-Driven Computational Model of Human Causal Reasoning. In *CogSci*.

Bridewell, W., & Bello, P. F. (2016). A theory of attention for cognitive systems. In *Proceedings of the 4th Annual Conference on Advances in Cognitive Systems* (pp. 1–16). Evanston. Conference on Advances in Cognitive Systems (pp. 1–16). Evanston.

Briggs, G., Lovett, A., Bridewell, W., & Bello, P. F. (2023). Attentional strategies and the transition from subitizing to estimation in numerosity perception. *Cognitive Science*, *47*, e13337.

Dittrich, W. H., & Lea, S. E. (1994). Visual perception of intentional motion. *Perception*, *23*, 253–268.

Gao, T., Baker, C. L., Tang, N., Xu, H., & Tenenbaum, J. B. (2019). The cognitive architecture of perceived animacy: Intention, attention, and memory. *Cognitive Science*, *43*(8), e12775.

Gao, T., McCarthy, G., & Scholl, B. J. (2010). The wolfpack effect: Perception of Animacy irresistibly influences interactive behavior. *Psychological Science*, *21*(12), 1845–1853.

Gao, T., Newman, G. E., & Scholl, B. J. (2009). The psychophysics of chasing: A case study in the perception of animacy. *Cognitive Psychology*, *59*, 154–179.

Gerstenberg, T., Peterson, M. F., Goodman, N. D., Lagnado, D. A., & Tenenbaum, J. B. (2017). Eye-tracking causality. *Psychological Science*, *28*(12), 1731–1744.

Heider, F., & Simmel, M. (1944). An experimental study of apparent behavior. *American Journal of Psychology*, *57*, 243–259.

Khemlani, S. (2022). nodus-ponens: A light, full-stack framework for running high-level reasoning and cognitive science experiments in Node.js. [Computer software]. Retrieved from https://www.npmjs.com/package/nodus-ponens

Kon, M., & Francis, G. (2022). Perceptual grouping strategies in a letter identification task: Strategic connections, selection, and segmentation. *Attention, Perception, & Psychophysics*, *84*(6), 1944–1963. https://doi.org/10.3758/s13414-022-02515-1

Kon, M., & Francis, G. (2023). Perceptual grouping strategies and texture segmentation: Strategic connections and selection. *Vision Research*, *210*, 108263.

Lawrence, M. A. (2016). EZ: Easy analysis and visualization of factorial experiments (R Package Version 4.4-0) [Computer software]. Retrieved from https://CRAN.R-project.org/package=ez

Lovett, A., Bridewell, W., & Bello, P. (2019). Selection enables enhancement: An integrated model of object tracking. *Journal of Vision*, *19*(14), 1–31.

Meyerhoff, H. S., Huff, M., & Schwan, S. (2013). Linking perceptual animacy to attention: Evidence from the chasing detection paradigm. *Journal of Experimental Psychology: Human Perception and Performance*, *39*, 1003–1015.

Meyerhoff, H. S., Schwan, S., & Huff, M. (2014a). Perceptual animacy: Visual search for chasing objects among distractors. *Journal of Experimental Psychology: Human Perception and Performance*, *40*, 702–717.

Meyerhoff, H. S., Schwan, S., & Huff, M. (2014b). Interobject spacing explains the attentional bias toward interacting objects. *Psychonomic Bulletin & Review*, *21*, 412–417.

Palmer, S., & Rock, I. (1994). Rethinking perceptual organization: The role of uniform connectedness. *Psychonomic Bulletin & Review*, **1**(1), 29–55.

Posner, M. I. (1980). Orienting of attention. *The Quarterly Journal of Experimental Psychology*, *32*(1), 3–25.

R Core Team (2023). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. [Computer software]. Retrieved from https://www.R-project.org

Schultz, J., & Frith, C. D. (2022). Animacy and the prediction of behaviour. *Neuroscience & Biobehavioral Reviews*, *40*, 104766.

Treisman, A. M., & Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology*, *12*(1), 97-136.

Ullman, S. (1984). Visual routines. *Cognition*, *18*, 97–159.

Wolfe, J. M., & Horowitz, T. S. (2004). What attributes guide the deployment of visual attention and how do they do it?. *Nature Reviews Neuroscience*, *5*(6), 495-501.