EPIC Progress

David Kieras & David Meyer

University of Michigan

June 5, 2005

Introduction

Microsummary of Empirical Work

Agenda for EPIC Architecture Development

Diagram of the Current EPIC Architecture

Agenda, continued

Actual MMWS Display

Preliminary (First Draft) MMWS Task Display

Why Visual Issues are Major

Agenda, concluded

But things turned out to be more exciting ...

Brief Preliminary Rant about Conventional Theoretical Wisdom

Microsummary of Empirical Work

Updating of Verbal Working Memory

Adam Krawitz - Psychonomics, 2004

Modeling has begun - Adam Krawitz

Learning of Individual Production Rules in Choice Reaction Tasks

Steven Lacey, Psychonomics 2004

Two processes clearly evident in learning individual S-R pairs:

- An immediate learning process - flat learning curves.
- A slower learning process - exponential learning curves.
- Indication that immediate learning is verbally mediated.
  - Final asymptotic performance is faster for slowly-learned pairs.
  - Interfered with by articulatory suppression.

Nonhomogeneity of S-R pairs:

- Powerful differences related to immediate learning.
- Spatially-based - will return to.

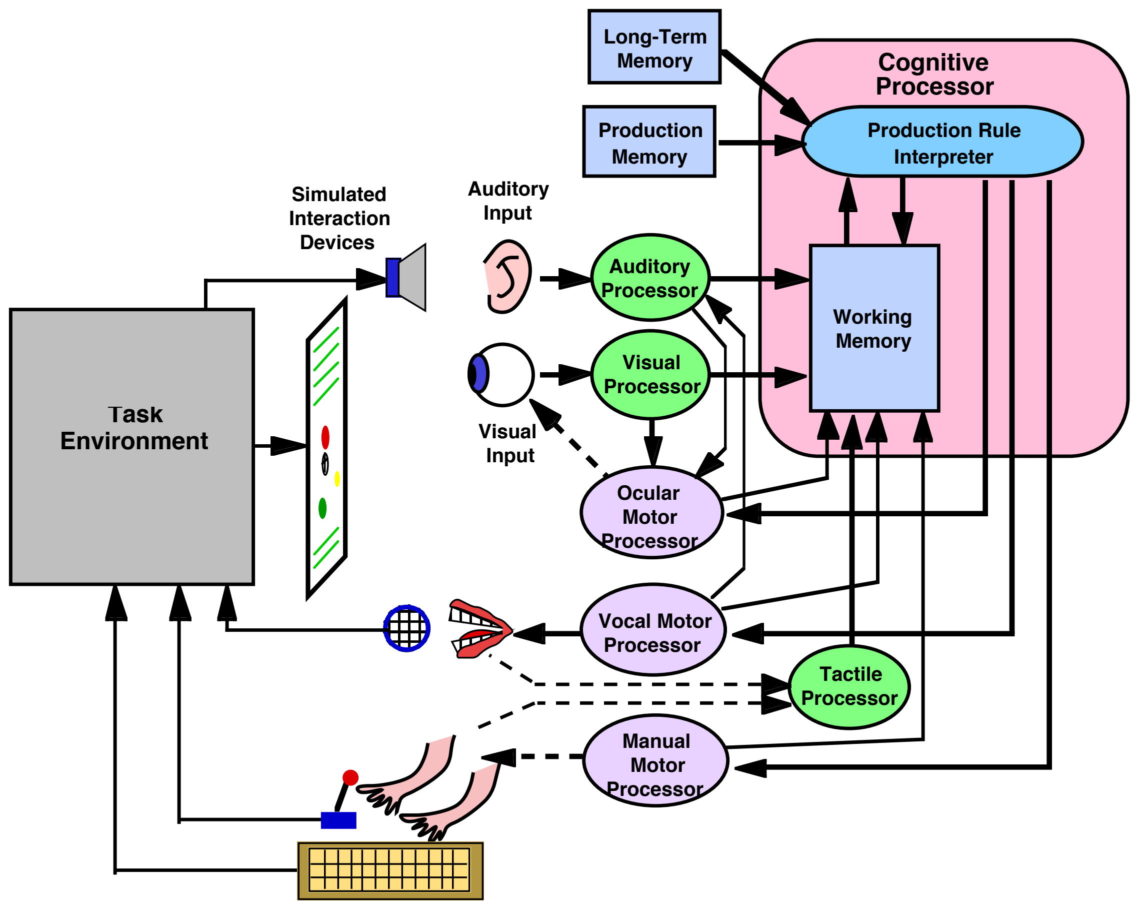
Agenda for EPIC Architecture Development

Original agenda for last year's work:

Add literature background to EPIC documentation.

Show where each assumption in the current architecture was based on the literature.
Much of this in earlier publications, but needed consolidation, updating.
Goal: Review, update of key parts of literature in relation to EPIC.

Diagram of the Current EPIC Architecture



Agenda, continued

Construct EPIC model for a task of full realistic complexity.

Demonstrate “scalability” - have done to some extent elsewhere.

Started on CIC Radar operator task - visual search is key part.

Task Domain: Navy CIC using advanced workstation, MMWS

- Separate project work using computational GOMS Models.
  - Modeling tasks (also in teams) to evaluate interface designs.
  - Works very well, but GOMS models are very approximate.

Goal of EPIC Complex Task work:

Piggy-back on GOMS models to provide overall task strategy.

- Including realistic scenario for realistic task scenario:
  - Dozens of moving visual objects, 1.5 hours of simulated time.
  - Considerable visual, manual auditory, vocal activity.
  - Includes a critical event that is often missed by human operators.

See if greater detail is useful, and what is involved in providing it.

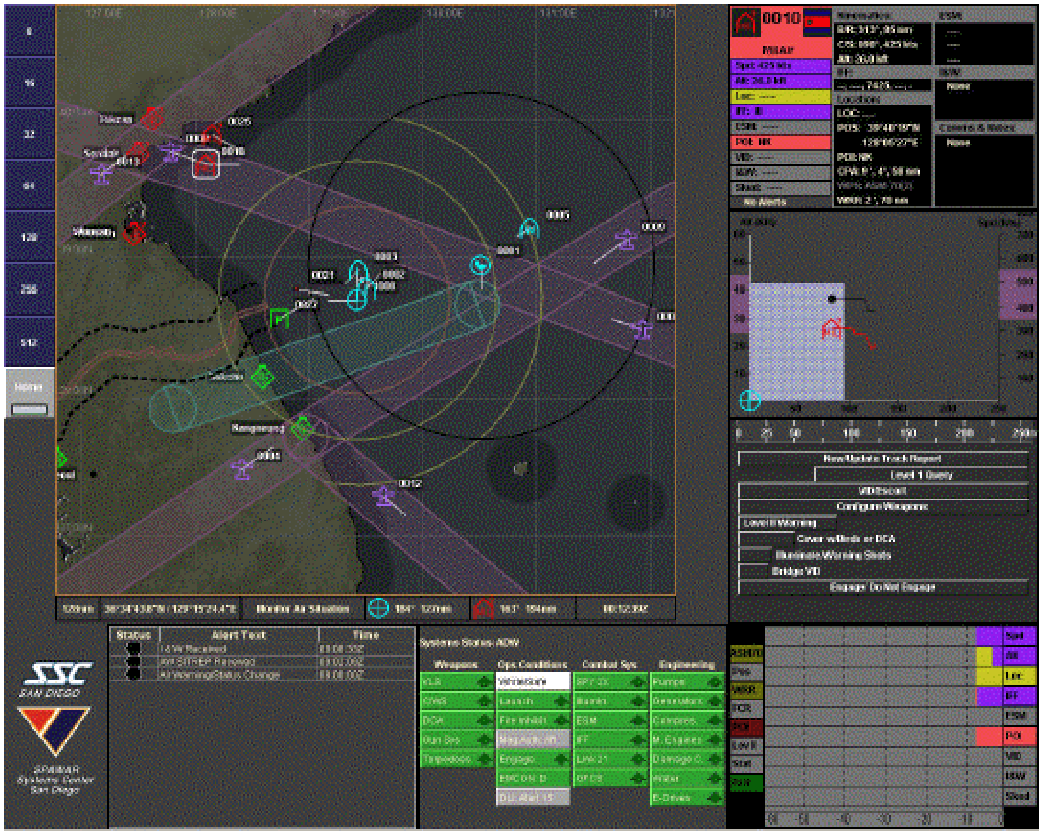
- E.g. provide full explanation of difficulty of detecting critical events.

However, first problem: Task is extremely vision-intensive.

Many problems in vision, visual search, visual memory.

Actual MMWS Display

Quite a few things not directly used in the task.



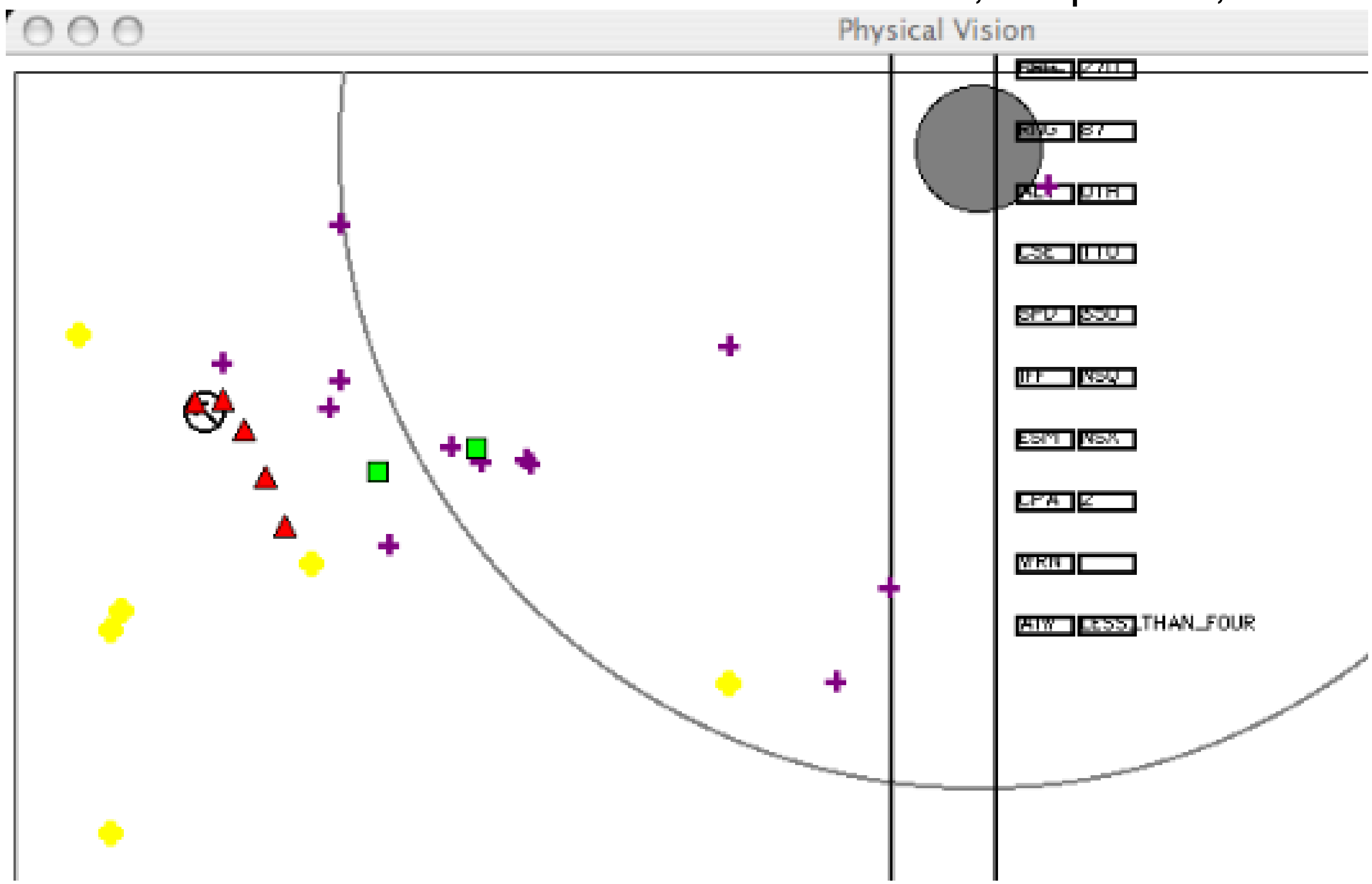
Preliminary (First Draft) MMWS Task Display

Physical Display presented to human simulated with EPIC.

Many "tracks" - radar blips, coded by color and shape.

Simulated human is now starting to examine the track data table.

- Make decisions about whether track is hostile, suspicious, etc.



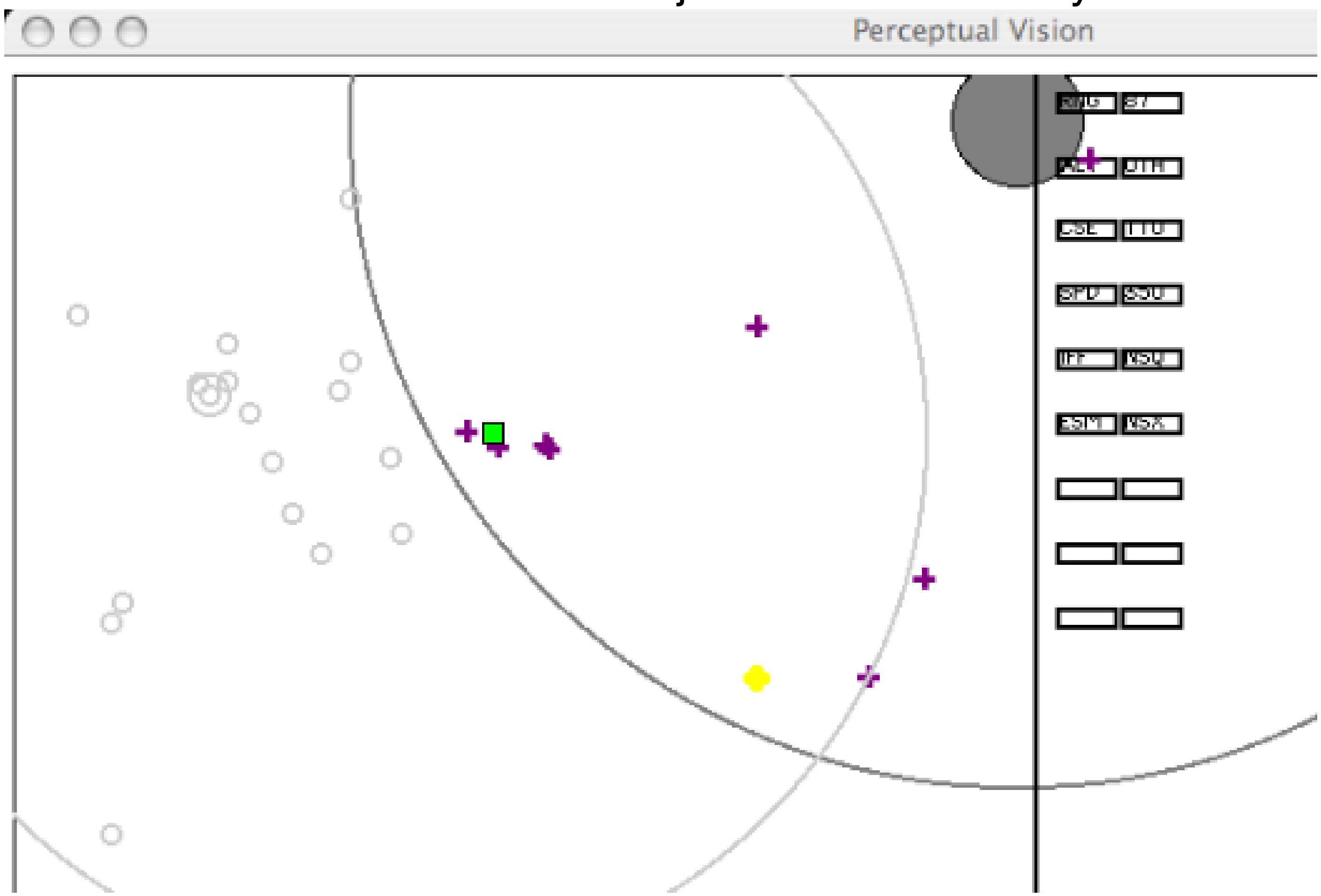
Why Visual Issues are Major

Perceptual Situation - what is available to cognition.

Distance (eccentricity) makes colors, shapes unavailable.

How much information is remembered between eye movements?

Must visual search for relevant objects start over every time?



Agenda, concluded

Complete current modeling work on visual search tasks.

Focus on complex visual search task with realistic displays.

- Marshall & St. John's eye-tracking data.

Goal: Provide basic mechanisms, parameter values for modeling display-intensive tasks like MMWS task.

But things turned out to be more exciting ...

Good news! Bad news! Good news!

Good news is that architecture is testable - aspects of visual search tasks could not be accounted for properly.
Latencies could not be made short enough reasonably.

Bad news is that the problem is pretty serious.
Motor processors were based on apparently dubious conclusions in the motor control literature about motor feature programming.
So literature update meant rethinking a major component!

Good news is a new architectural concept: primacy of spatiality in interacting with objects visually and manually.
Processors work directly and efficiently with object locations.
Humans are optimized to interact with objects located in space.

Best way to illustrate this is start with visual search models, then overview results from literature update.
Results seriously question several aspects of conventional wisdom.

Brief Preliminary Rant about Conventional Theoretical Wisdom

Conventional wisdom - explanation for an interesting phenomenon that seems to work well and becomes generally accepted.

Will be making various negative remarks about conventional wisdom.

Since usually based on verbal theorizing, with only limited quantitative analysis, is often seriously wrong.

No substitute for quantitative computational modeling that takes task strategies and architectural constraints fully into account.

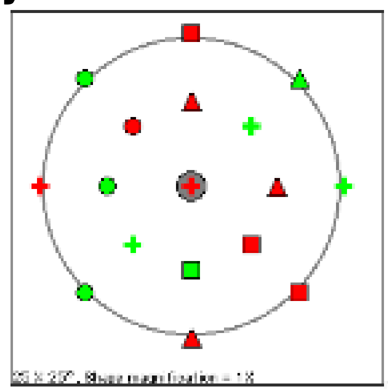
Modeling Findlay Visual Search Task

Current Findlay Task Modeling

Findlay, J.M. (1997). Saccade target selection during visual search. *Vision Research*, 37(5), 617-631.

Paradigm:

Sample Model Display - Physical Stimulus



Inner circle is 5.7 degree radius, outer is 10.2 degrees.
Four shapes, two colors. Initial fixation point.

Subject must fixate target specified as a conjunction of two features.
E.g. the red cross only one in the display.

First saccade is the focus of interest - what can be seen and used to guide eye movements in visual search?

Findlay Task (continued)

Paradigm:

Recorded: Latency and ending point of first saccade.

- Result: Latency about 250 ms, independent of saccade target.

Of interest: Proportion of fixations to combinations of the two features.

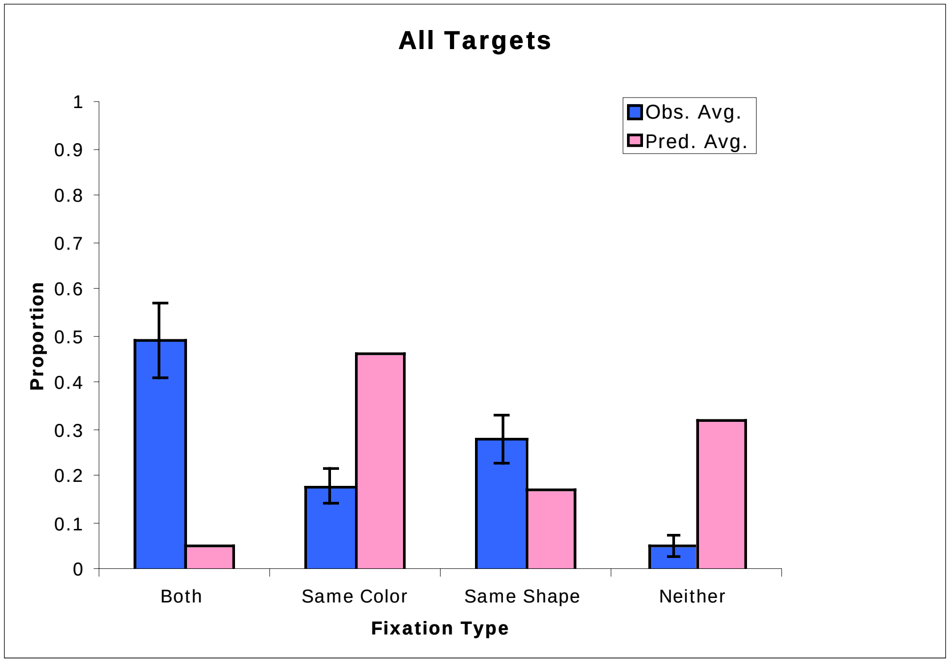
- Both, Same Color (wrong shape), Same Shape (wrong color), Neither.

New data: Findlay supplied fine-grain break out of data set.

Series of models justify basic conclusions.

Random-Search Model

A baseline model: Pick an object at random and saccade to it.



Since two colors, four shapes, predicted results make sense, but data is very different.

Models Using Visual Guidance

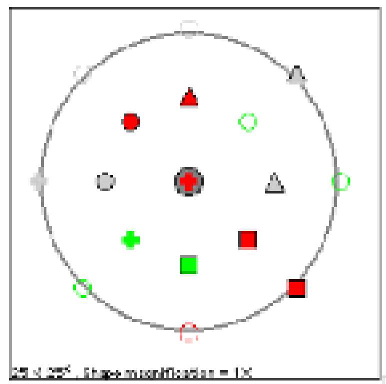
Simple probability model of color, shape availability:

Inner ring (5.7 degrees): 0.7

Outer ring (10.2 degrees): 0.3

Each property available independently of the other.

Sample Model Perceptual Display



Models are "optimal" in that they never choose to look at an object known to be incorrect.

If no matching choice, choose an object with unknown properties, rather than wrong properties.

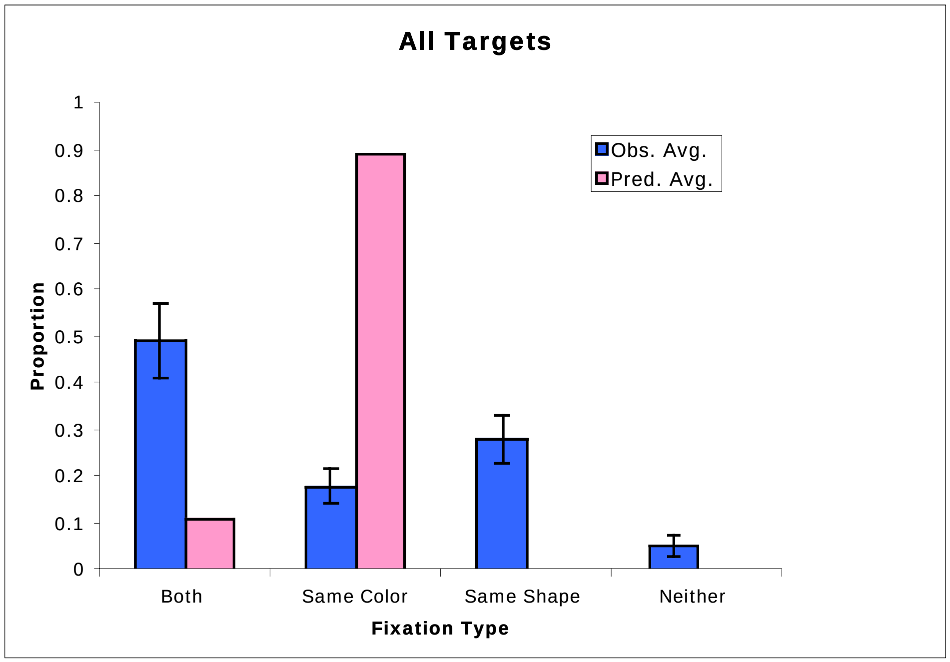
Start with Single-Feature Search Models

Conventional wisdom: Only one feature can be used in a fast parallel search; slow serial search required for conjunction searches.
"Feature integration theory" - Treisman & Gelade (1980).

Can a one-feature model account for the results?

Color-only Model

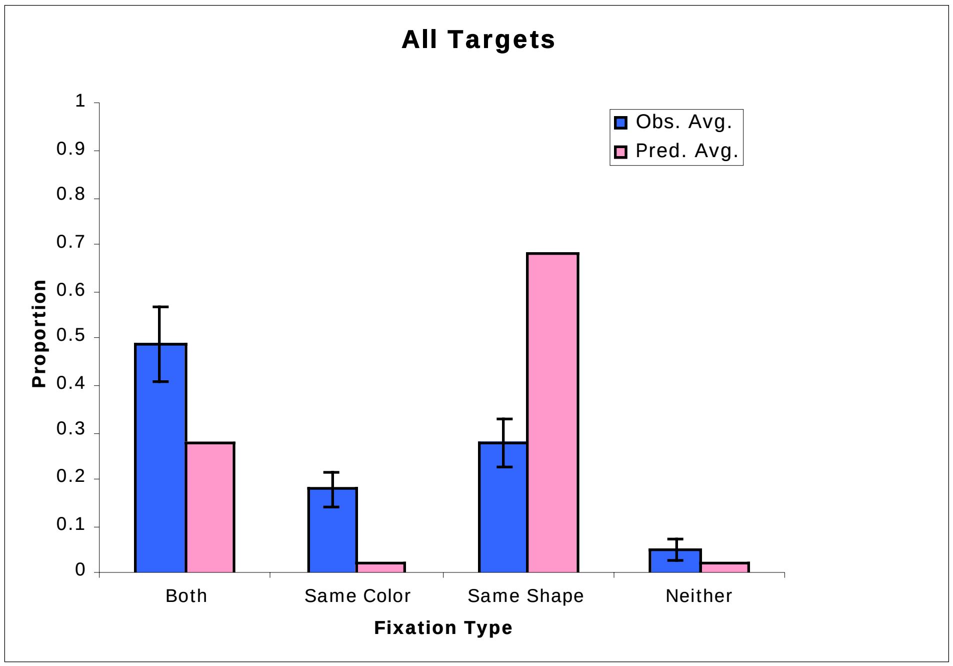
Pick an object of the right color, look at it. If no match, look at unknown color object.



Same Shape means right shape, wrong color - never happens since right color is almost always visible.

Shape-only Model

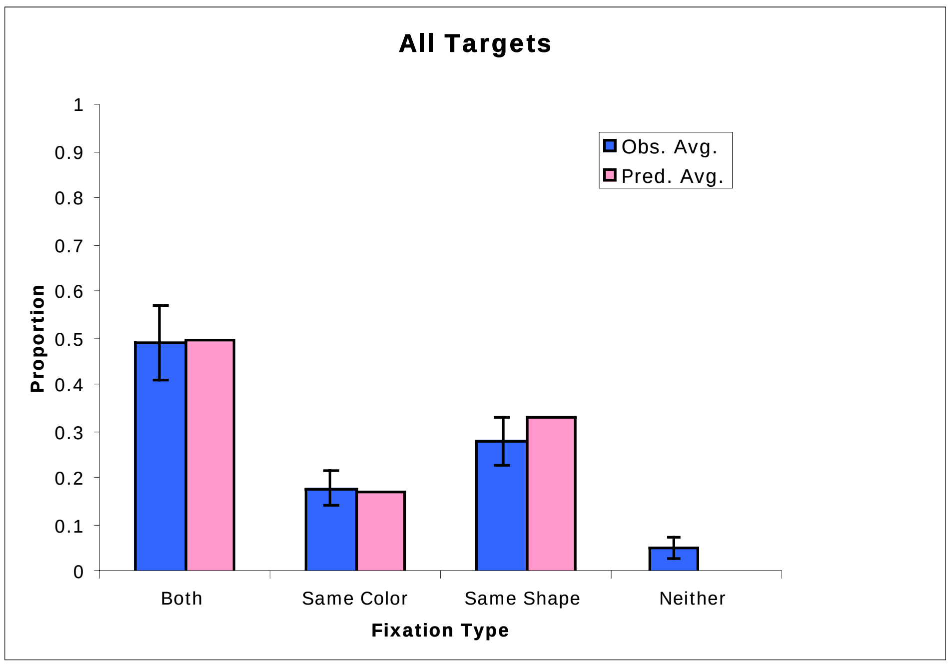
Pick an object of the right shape, look at it. If no match, look at unknown shape object.



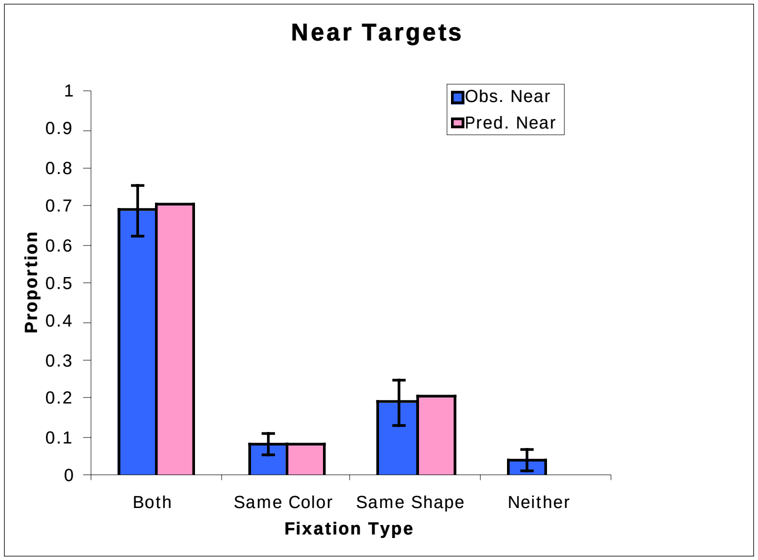
Sometimes no shape match visible, so we get a few Same Color and Neither cases.
Looks better than Color-only, but still far off from data.

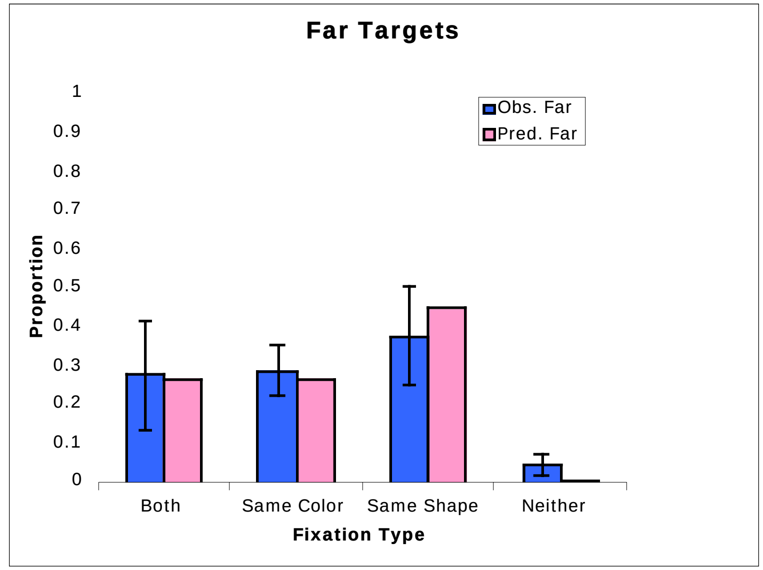
Conjunction Model

If both color and shape match, move to it; if not, move to a matching shape with unknown color; if none, move to a matching color with unknown shape; if none, pick an object with both unknown color and shape.



Breakout by Near vs. Far Targets





Shape is more diagnostic than color, so preferring it is a good strategy, matches the data.

Neither cases are under-predicted.

Result of independent property availability.

Some kind of "noise" process produces an incorrect fixation independently of stimulus properties.

Saccade Latency Problem

Saccade latency is only about 250 ms - very fast.

Same regardless of whether conjunction detected or not.

- Note that 250 ms movement latencies are common in the literature - this is a pervasive "best case" value.

Problem for "typical standard" EPIC parameters:

100 ms transduction/recognition times + 50 ms avg motor feature programming time + 50 ms initiation time = 200 ms

Leaves only one 50 ms cycle available for production rules.

Ouch! How can we fit the latency data with only one cycle?

Could assume faster encoding times, but gives two cycles at the most.

50 ms transduction/recognition times + 50 ms avg. feature programming time + 50 ms initiation time = 150 ms

Might be plausible given experimental procedure, but hard to swallow.

Hard Way to Get Fast Responses

Augment PPS matching algorithm and syntax to allow hierarchical priority matching in a single rule cycle:

- If there is something that matches both color and shape, look at it.
- If there is nothing that matches on both color and shape, but there is something that matches the shape and has unknown color, then look at it.
- If there is nothing that matches on both color and shape, and nothing that matches the shape and has unknown color, but there is something that matches the color and has an unknown shape, then look at it.
- If there is nothing that matches on both color and shape, and nothing that matches the shape and has unknown color, and nothing that matches the color and has unknown shape, but there is something that has both unknown shape and unknown color, then look at it.

Complex and difficult - even for Soar's matcher.

Rule conditions must include negations of conjunction.

Model using this capability and standard time parameters gives predicted latency = 230 ms.

Comfortably fast; within “tweak” range to match.

Easy Way to Get Fast Responses

Simple two-cycle selection process using simple production rules.

1. Nominate objects that match different criteria.
   - E.g. right shape & unknown color.
2. Choose best match from nominated objects and look at it.
   - No Both-match, but do have a Shape-match - look at it.

Assume that saccades can be initiated a lot faster.

Zero movement feature programming time!

With normal other parameters, predicted latency = 270 ms.

Within “tweak” range.

Question: Is zero feature programming time justified?

Either super-complex production rules, implausible encoding times, or discard assumptions about saccade programming.

Will return to this problem.

Summary of Findlay Task Results

Findlay's results, and ease of modeling them, imply:

Perceptual identification of objects takes the same time for single feature and two-feature target.

Conventional theoretical wisdom doesn't fit the data.

EPIC's high-capacity parallel visual system can account for the data.

Eccentricity effects on acuity limit accuracy.

Selection of targets can be done optimally given accuity effects.

Preparation of saccades appears to be very short fast (e.g. zero time).

Modeling Marshall Visual Search Task

Visual Search of Displays of Navy Symbols
Display Decluttering Manipulation
Sample 0% Decluttered Display with Eye Trace
Sample 75% Decluttered Display
Tradeoffs in Density Manipulations
Current Model
Good News about RT and Number of Fixations
Good News about Fixation Durations
Mixed News about Proportion of Fixations
Worst News is about Saccade Distance
Memory for Fixated Objects
No-Memory Guided Search Model
Good RT and Number of Fixation Predictions
Mispredicted Fixation Types
Repeat Fixation Problem
Overpredicted Saccade Distances
Limited Availability No-Memory Model?
Motor Programming Problems
in the Marshall Task Models
Summary

Visual Search of Displays of Navy Symbols

Studies of effects of crowding in visual search of realistic displays.

St. John, M., Marshall, S.P., Knust, S.R, & Binning, K.R. (2003). Attention management on a geographical display: Minimizing distraction by decluttering. Unpublished manuscript.

New data: Complete fixation data set supplied by Sandy Marshall!

Objects are realistic Navy symbols for radar-type displays.

Probe object appears, then display of objects.

Subject must click on the two targets identical to probe.

Origin property:

- Different colors and overall shapes.
- Apparently only the color is used to guide search.

Platform and *Direction* properties:

- Internal symbols or text labels.
- “Leader lines” that normally indicate course and speed.

Display Decluttering Manipulation

"Declutter" manipulation: Can objects be found better if irrelevant objects "decluttered" - removed from the display in some way?

In this subset of the data, object replaced with a gray dot.

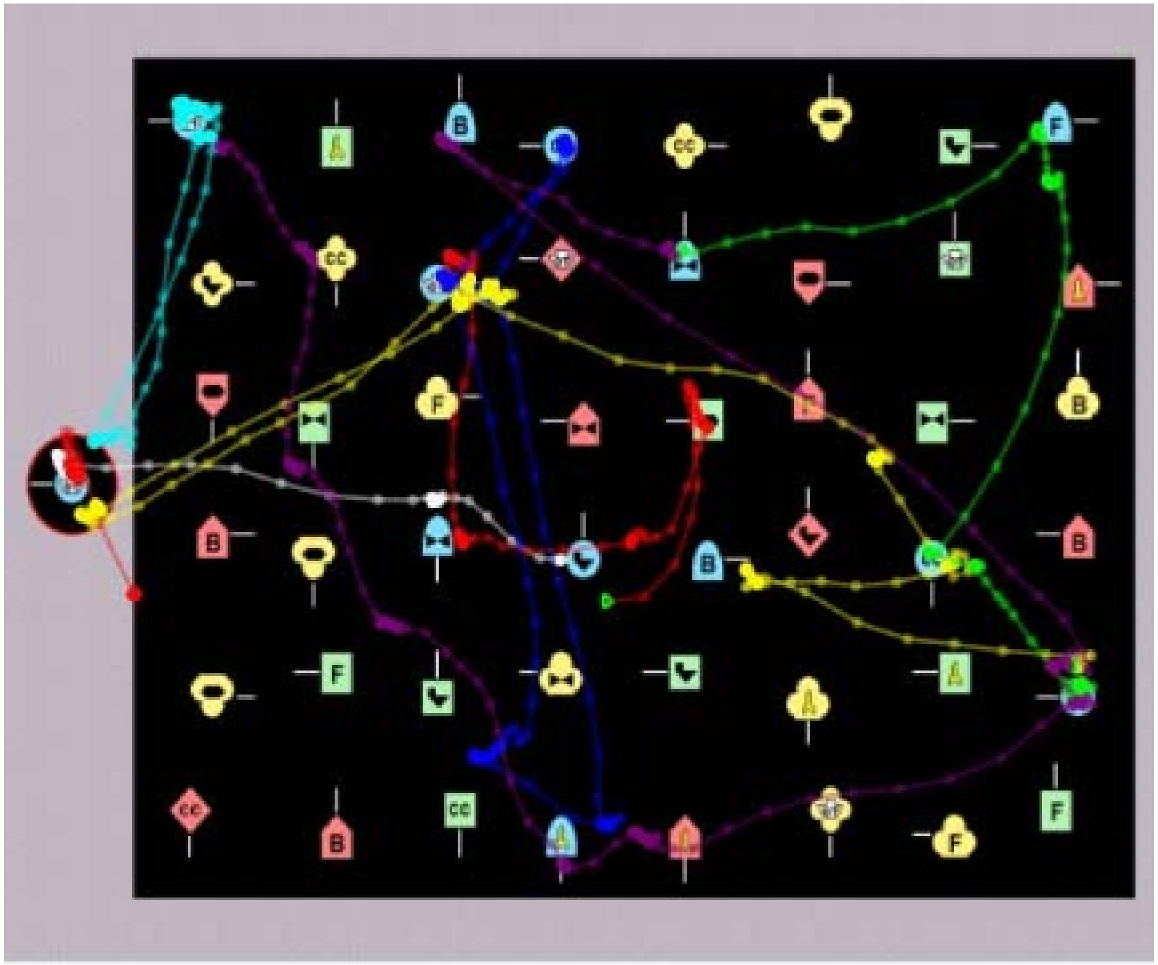
Four levels of decluttering: 0%, 25%, 50%, 75% of objects removed.

Realistic manipulation:

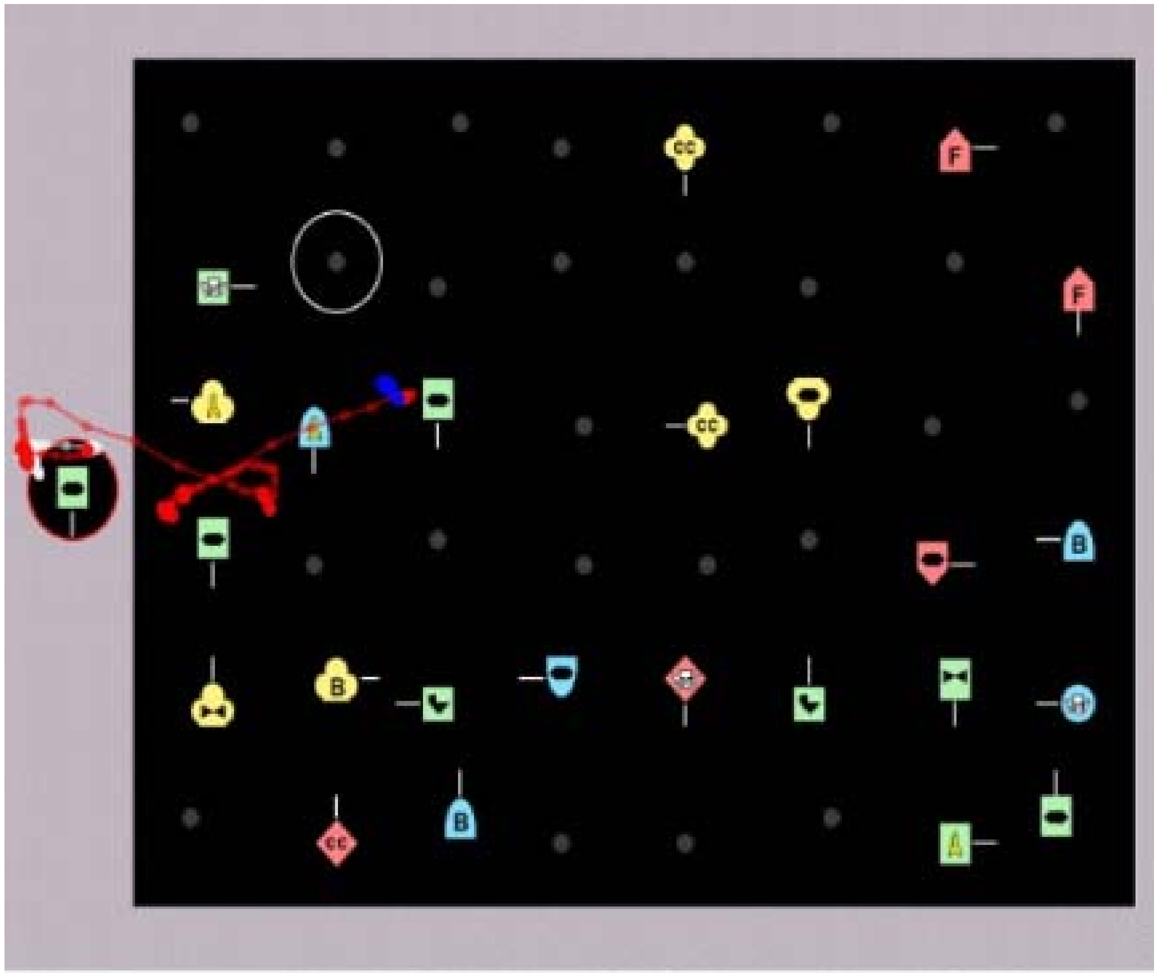
Highly similar distractors were always left on the display, so decluttered objects not randomly selected.

Real radar display decluttering would be based on object properties, not at random.

Sample 0% Decluttered Display with Eye Trace



Sample 75% Decluttered Display



Tradeoffs in Density Manipulations

While always 48 objects on the screen, higher levels of declutter mean fewer icons on the screen, which on the average are further apart.

With more decluttering:

Average eccentricity greater, but average crowding less.

Crowding effects increase with eccentricity, but decrease with distance between target and crowding objects.

Bouma's rule: Area of crowding effect is eccentricity / 2

Vision literature: Area of crowding effect is larger on peripheral side of target than central side.

Thus eccentricity and crowding due to density may not trade off in any simple way.

E.g. Everett & Byrne found only very small effects of icon density in a visual search task.

Current Model

Basic strategy: Choose matching color object to look at; if none, choose an *unvisited* object at random.

Assume large and reliable memory for whether object has been examined.

- Will return to.

Foveal property (the label) might be mis-encoded with some probability (e.g. .08). If encoding failure detected, give that object priority to re-fixate.

- Shape, Platform, Direction available only in fovea.
- Color availability adjusted using simple zone model.

Basic result: Color availability must decrease as icon density increases in order to match the data.

Color of object available out to 9, 8, 7.5, 7 degrees as display goes from lowest density to highest density (75% to 0% decluttered).

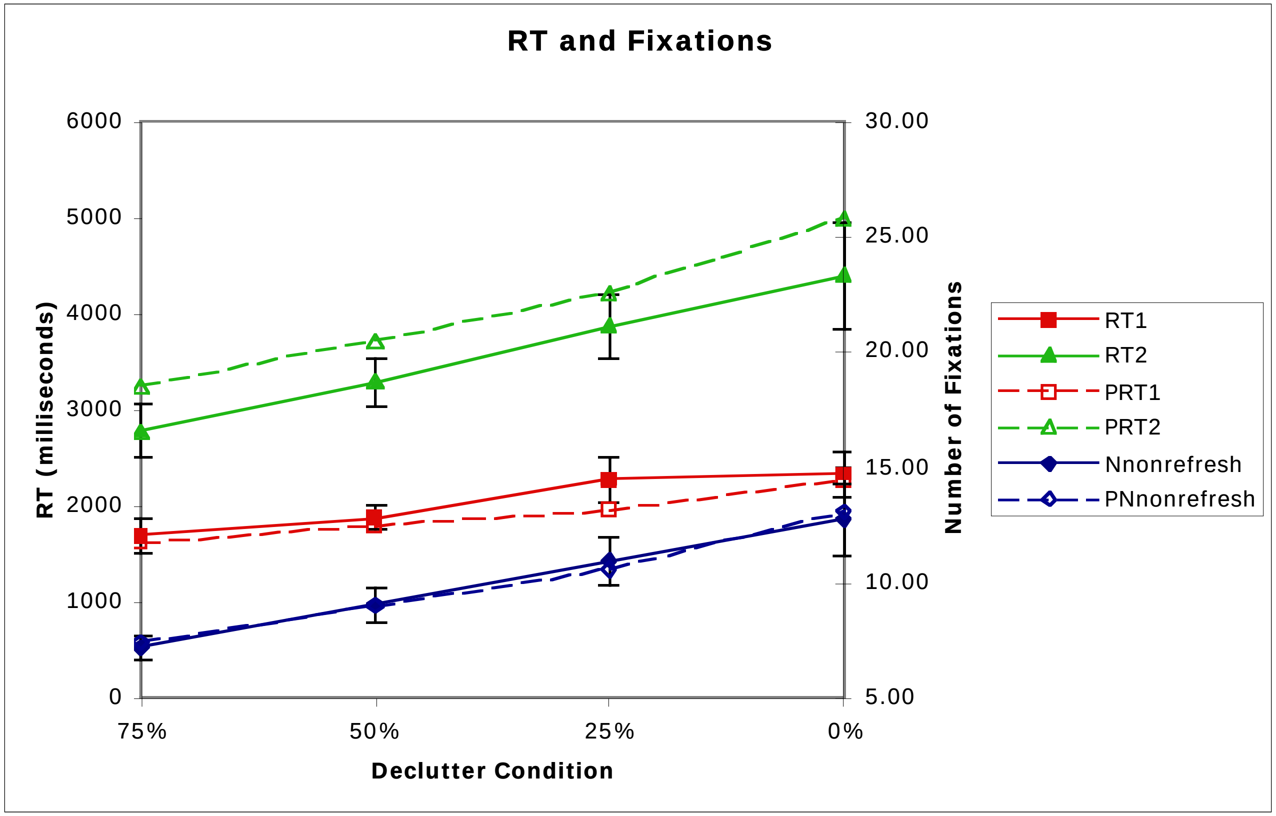
- If leave the color availability fixed at a single value, get gross misfits at some decluttering levels.

These are approximate values that could be used in modeling similar displays.

E.g. the CIC Radar Operator Task.

Compute availability dynamically as a function of local density of each object.

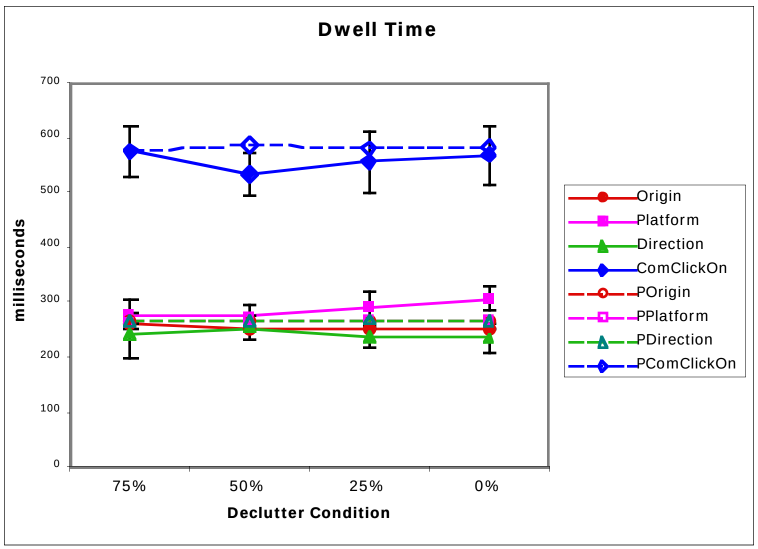
Good News about RT and Number of Fixations

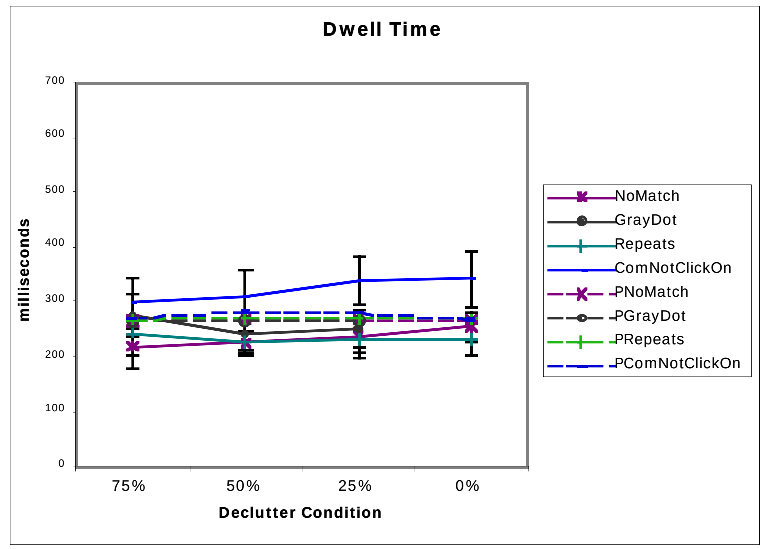


RT to second target is faster than predicted - some evidence in the data that eye goes ahead before click-on is complete - could match it.

Good News about Fixation Durations

Fixation Durations ("dwell" time) is about right - governs basic relation between number of fixations and RT.

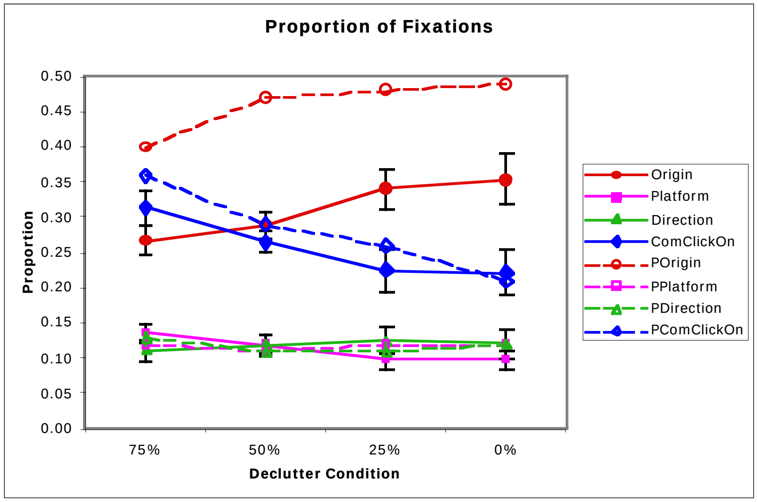


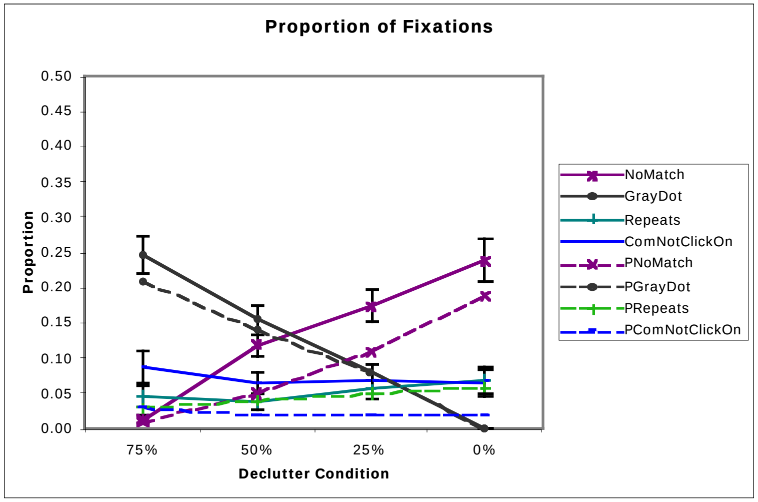


Fairly constant with declutter level.

The 250 ms time is pretty typical in the literature.

Mixed News about Proportion of Fixations





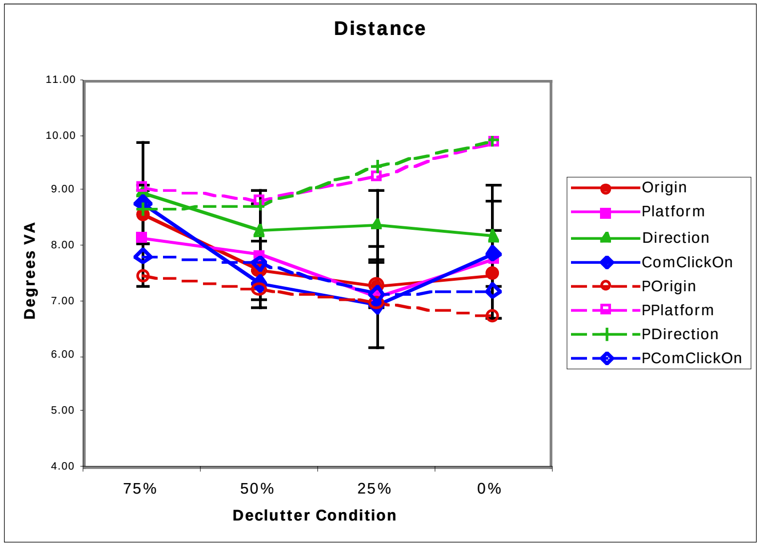
Color-guidance is too strong: Model looks at same-color targets more often than it should.

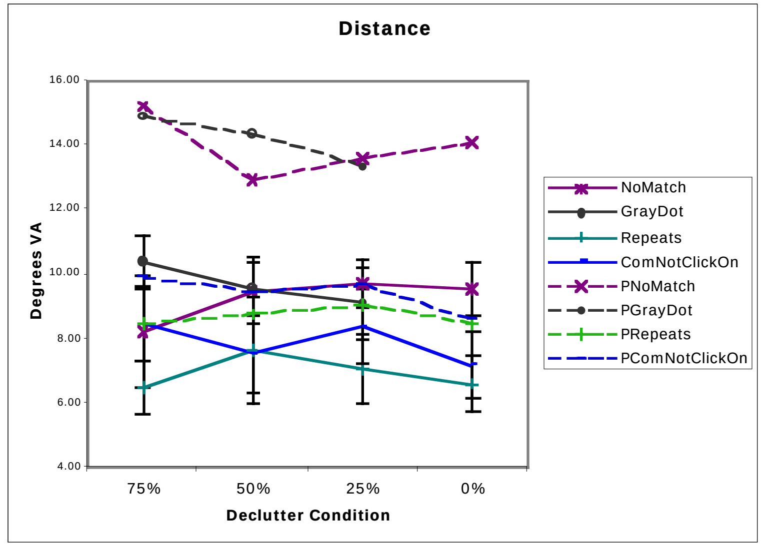
Not often enough at objects that don't match any properties.

Too few repeat fixations - more parameter adjustment?
"Double checking" of targets - a different repeat mechanism.

Worst News is about Saccade Distance

Overpredicted for saccades unguided by color.





Model makes "guesses" by choosing an object at random from the entire display, spreading out the searched area.

Currently, availability of objects is independent of eccentricity.

"Objecthood" is always perceived.

Should decrease with eccentricity - cf. various results in the literature.

Eccentricity + Crowding => objects “smoosh together” in the periphery.

Memory for Fixated Objects

Conventional wisdom in cognitive psychology is that visual search has no memory (Horowitz & Wolfe, 1998).

Still maintained with few qualifications (e.g. Wolfe, 2003, 2005).
Was demonstrated with RT paradigms and changing displays.
Several counter-arguments and demonstrations in the literature.

Key eye movement study: Peterson et al., 2001.

Under conditions where foveation of targets is necessary and 12 targets, only a 5.7% refixation rate!

- Statistical structure of data is essentially sampling without replacement.
- Most refixations are after one intervening object, possibly a result of encoding failures.
  - Used in Marshall model.
  - Can model the Peterson task this way as well.

No-Memory Guided Search Model

Marshall and St. John suggest that no-memory guided search like that proposed by Wolfe accounts for the number of fixations in their data.

Does it? Might be fun to find out.

Previous model modified:

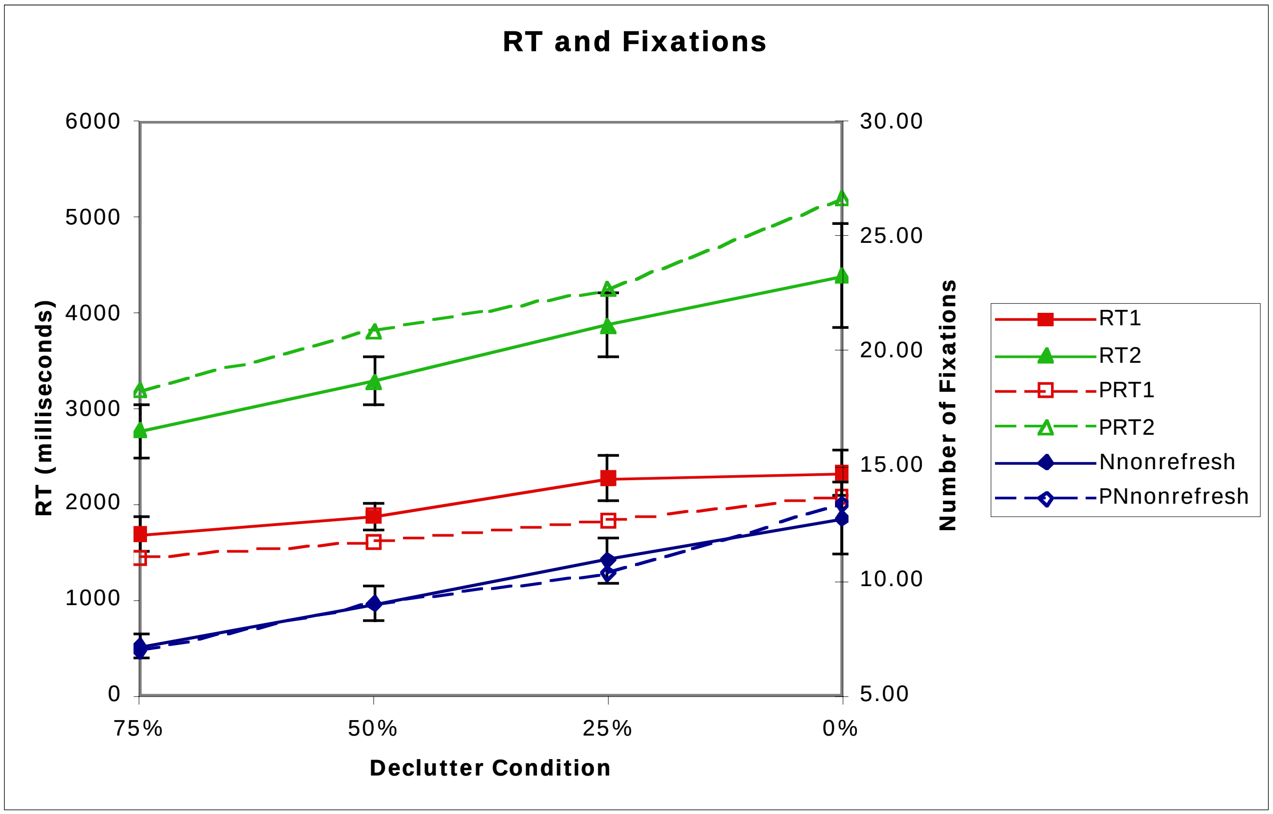
No memory for previously visited objects, no encoding failures, short-duration visual storage.

- Gives sampling with replacement behavior.

Color available within 60 degrees (almost the whole display), used to guide search - no eccentricity effects.

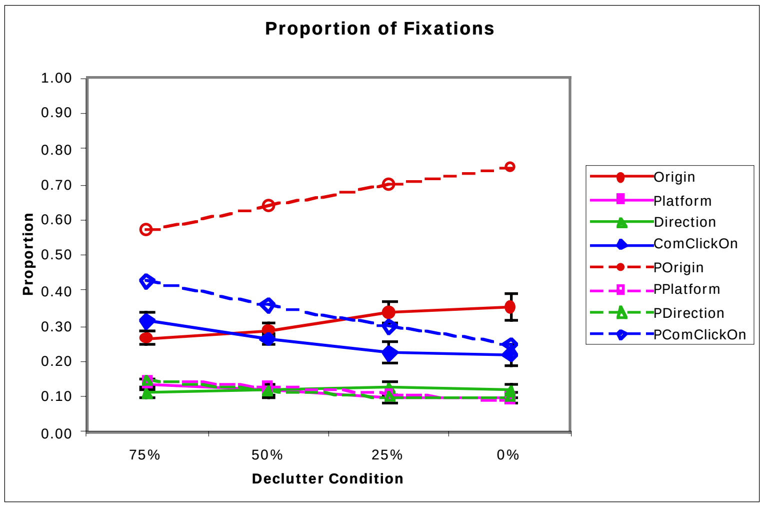
- Consistent with “passive vision” approach of ignoring retinal nonhomogeneity and eye movements during the search.

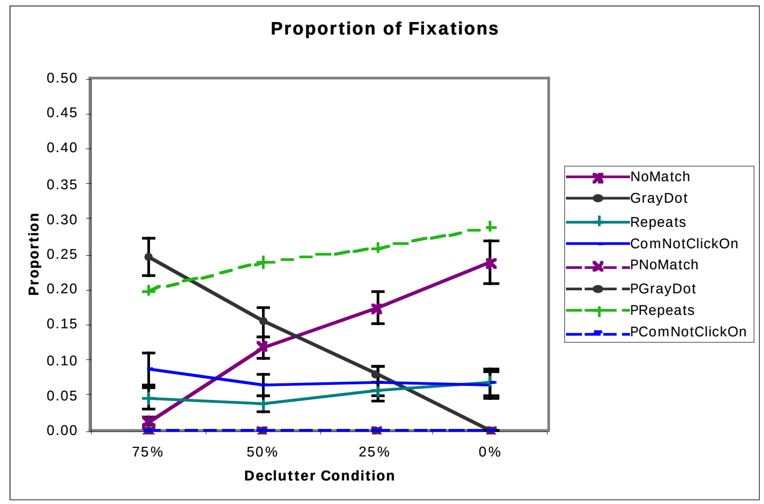
Good RT and Number of Fixation Predictions



Number of fixations is almost perfect.
Could tweak pointing actions to match RT.
But other aspects of search?

Mispredicted Fixation Types



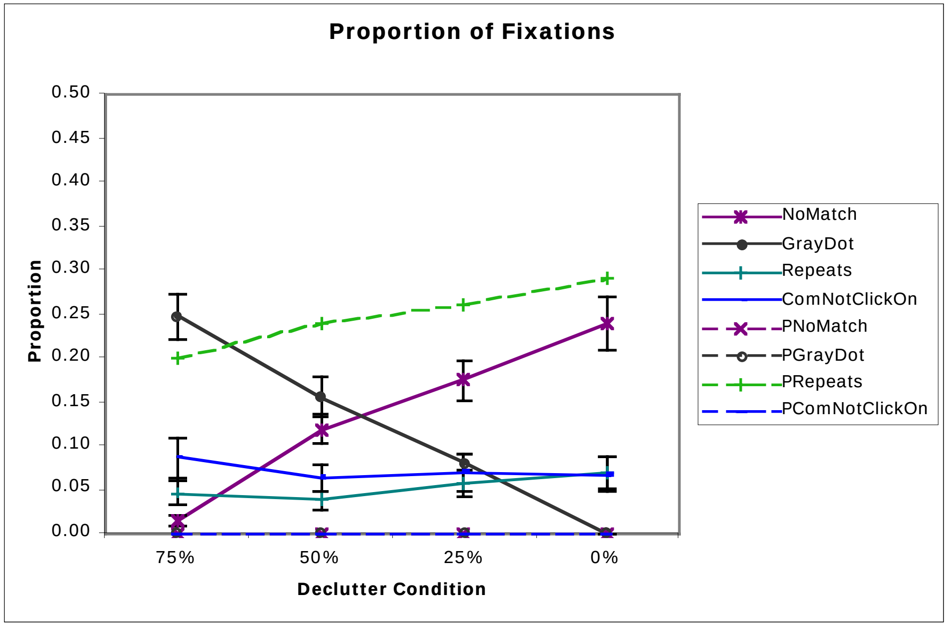


Color guidance for fixations is excessively strong.

Too many same-color fixations.

No Gray Dot or No Match fixations at all.

Repeat Fixation Problem



Way too many repeat fixations - around .25

In data, value is only about 5%!

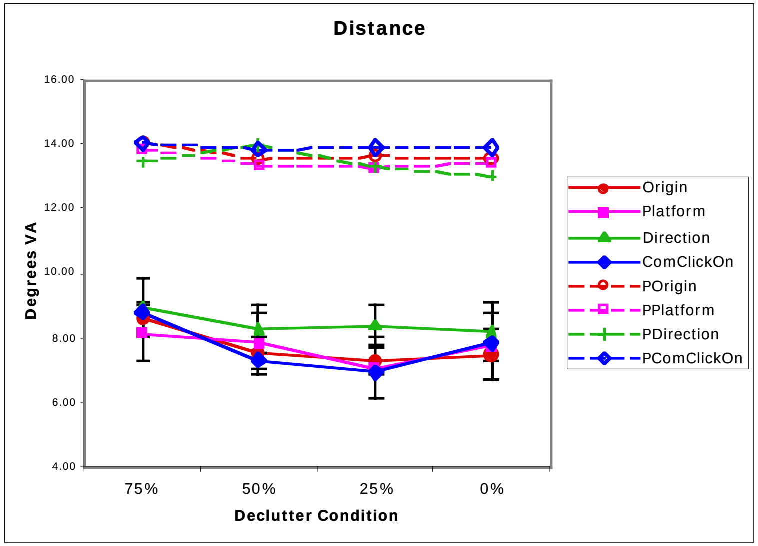
Roughly constant across declutter level.

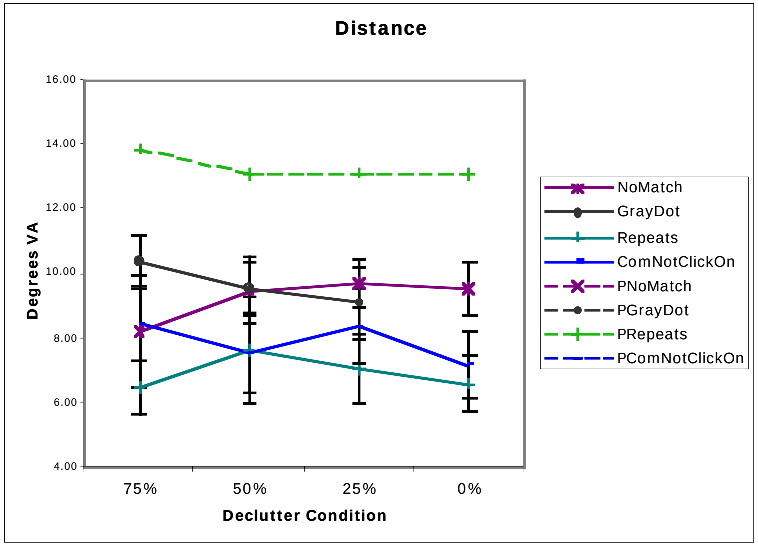
This is a serious error in prediction - not a fluke.

Other eye tracking results in the literature also show low rates.

Overpredicted Saccade Distances

Distances are terribly wrong, even for same-color fixations, because chosen at random from every where - no nearby bias from eccentricity.





Limited Availability No-Memory Model?

Does making color less available lead to more reasonable results without throwing off good results?

Lack of memory for previous visited objects causes many hangs under decluttered conditions.

E.g. oscillates between a small set of objects that are the only color-matching ones visible.

Could fix the hang with either:

- A very limited memory for previous visits.
  - McCarley et al., results suggest 3-4 objects may be enough.
- With some probability, look at a random object instead.
  - But drives up number of fixations and RT.

But possible that a very limited memory might suffice if search guidance limits the number of relevant objects.

But doesn't explain Peterson's results!

Premature to accept Wolfe's theory as reasonable account of visual search of realistic computer displays.

Don’t do it!

Motor Programming Problems in the Marshall Task Models

Cascade effect of many fixations and two mouse movements.

Even a little bit of time error will add up.

To get the latencies and dwells even approximately right, had to assume zero feature programming time for both saccades and mouse pointing movements.

In addition, much faster mouse moves because effective pointing target much larger than icon size.

How the experimental software actually works.

Now can specify effective target size instead of using visible object size to motor processor.

Summary

Conventional wisdom is wrong again:
Wolfe's no-memory guided search model cannot account for effects.
Memory for previously fixated objects is large; few refixations.
Eccentricity/crowding effects are present and must be represented.

EPIC's mechanisms scale up for realistic displays.
EPIC's high-capacity parallel visual system again can account for data.
Visual parameter values useful for the MMWS task displays.

Both eye movements and aimed manual movements appear to be programmed very rapidly.

Problems with Motor Feature Programming

Basic Motor Feature Programming Concept
Is there Something Wrong with EPIC's Motor Processors?
Where We are Going

Basic Motor Feature Programming Concept

Motor movements must be programmed before they are started, in terms of some number of motor features, each of which are serially prepared.

A movement can be initiated faster if the features are prepared in advance using precues to the movement, or remembered from a previous movement.

Popularized by Rosenbaum (1980); has become "conventional wisdom" in motor control field.

Adopted for EPIC as capturing key constraints provided by motor processes in human performance.

EPIC's motor processes compute and compare motor features to determine the motor preparation time.

- 50 ms for each feature that must be prepared anew for a movement.

Is there Something Wrong with EPIC’s Motor Processors?

If we really need zero movement feature programming time to match the visual search data, then something is wrong.

Do we need to make this change to the motor processors?

Let's go back and look at the basis for movement feature programming.

Some background on what the classical results show, and what some newer results show.

Sorry, a bit of literature review of low-level tasks.

But a major change to the architecture can’t be made lightly!

Where We are Going

Claimed motor feature programming effects are apparently a result of translations due to poor S-R compatibility - actually a Hick's Law effect.

Once a visual target has been identified, an aimed manual movement or an eye movement can be quickly launched to it without any S-R translation or motor feature programming delays.

Illustrate with focus on aimed manual movements, then skim over eye movements.

Aimed Manual Movement Programming

Aimed Manual Movements: Rosenbaum (1980)

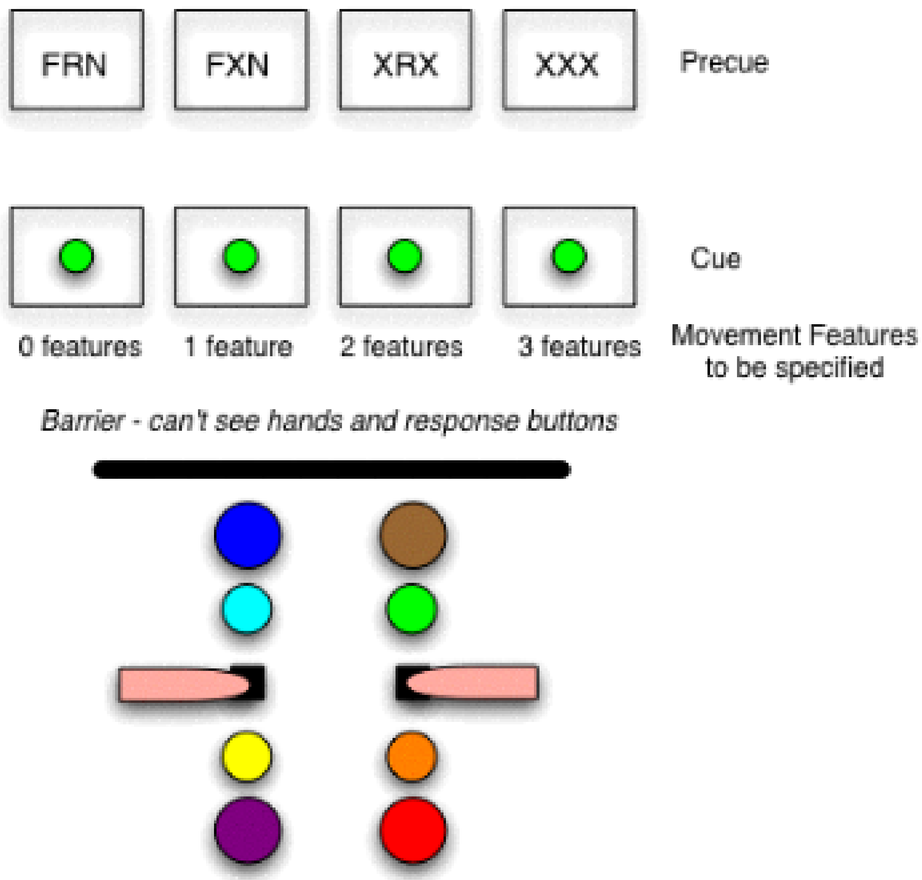
First major presentation of the motor feature programming concept.

Precue arm, direction, extent (distance) features for pointing movement to a target, then measure RT to a specific cue.

Manipulate number of features that are precued, which changes how many features have to be computed before starting to make the movement.

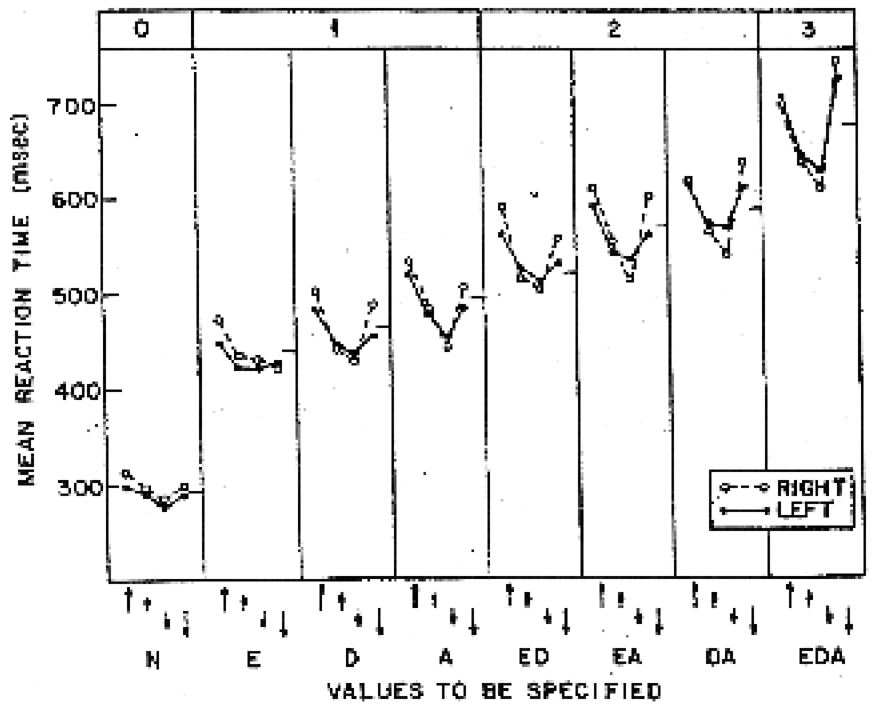
Movement targets identified via memorized color codes; movement precues given by text codes.

Rosenbaum's Task



RT is time when finger leaves “home” button.

Rosenbaum's Results



Movement RT increases with number of features that must be specified by the cue.

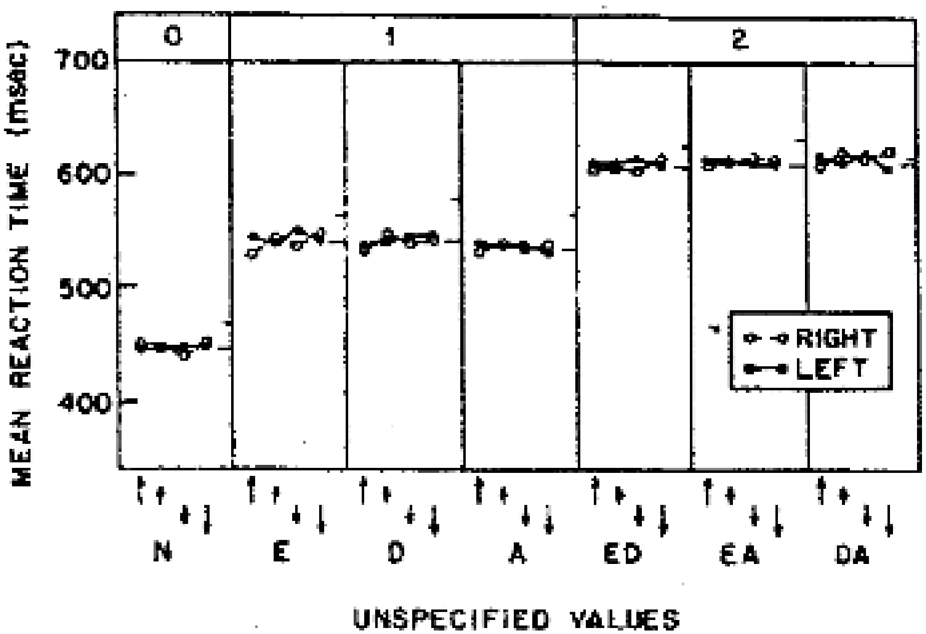
No such effects on the execution time for the movement.

Also, some feature and movement-specific effects.

Response Selection Effects?

Were effects due to differences in processing the precue and cue information to identify the movement target?
Rosenbaum's "stimulus identification" process.

Use decision task instead of button-pressing.
Subject had to respond (by voice) whether cued movement was a valid target given the precue.

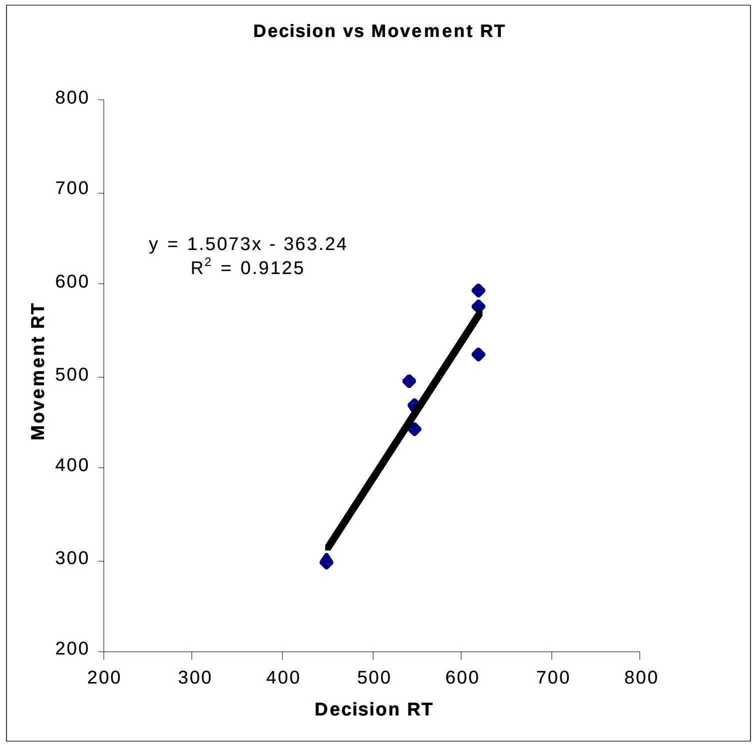


Effect of number of precued features, but nothing else.
Rosenbaum: Not an important component of the movement RT.

But still ...

Movement RT is clearly related to the decision RT.

91% of variance in mean movement RT for each feature type is accounted for by mean decision RT.



Suggests that a lot of the RT is actually due to response selection.

Doesn't explain the movement-specific effects, but they are quite small by comparison.

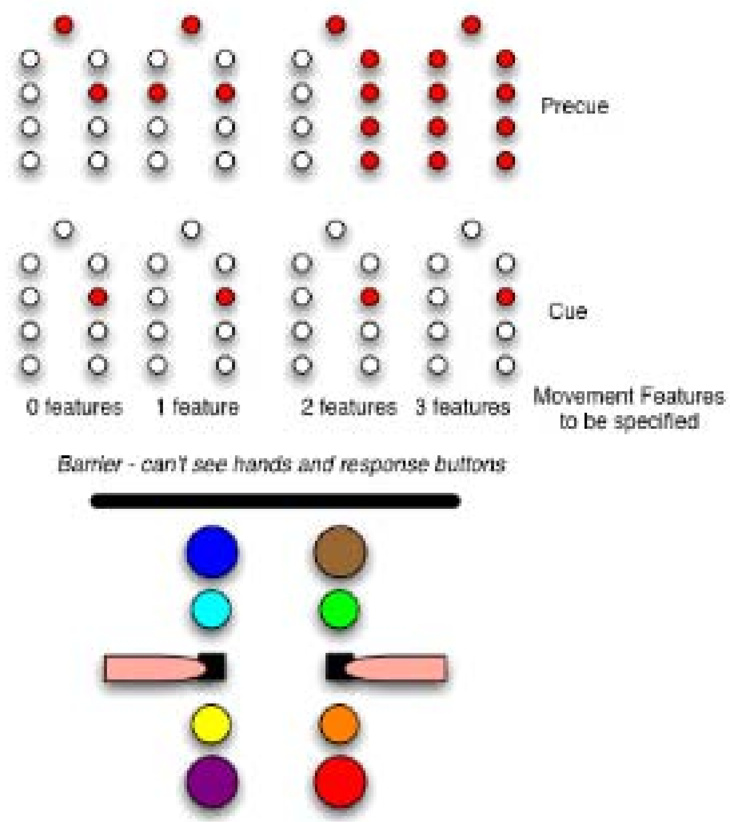
Goodman & Kelso (1980)

First, replicated Rosenbaum's results using color words or numbers instead of presented color dot cue.

Then used precues and target cue presented in fairly spatially compatible way.
   Light panel above response board with light layout matching button layout.

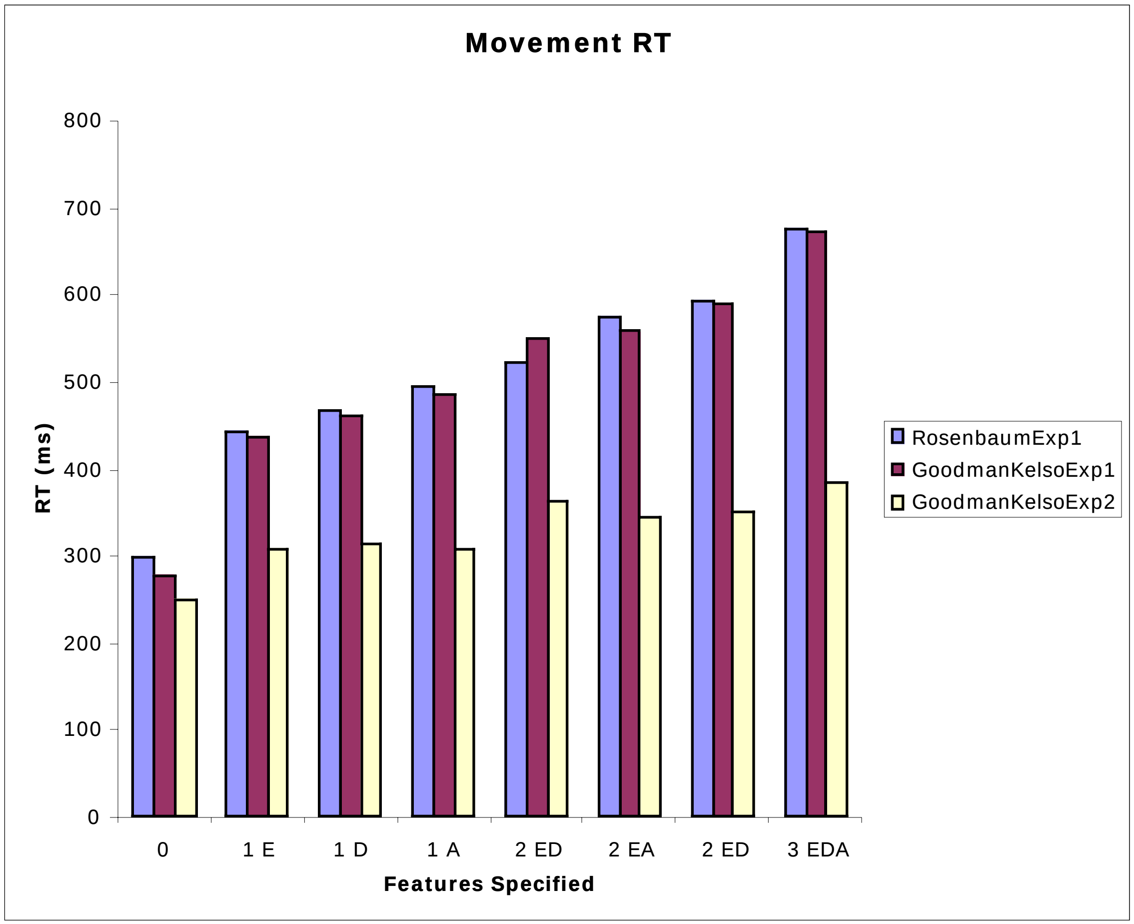
Result: More compatible task considerably diminished effect of precues, no movement-specific effects.
   Why would motor programming depend on S-R compatibility?

Goodman & Kelso Compatible Task

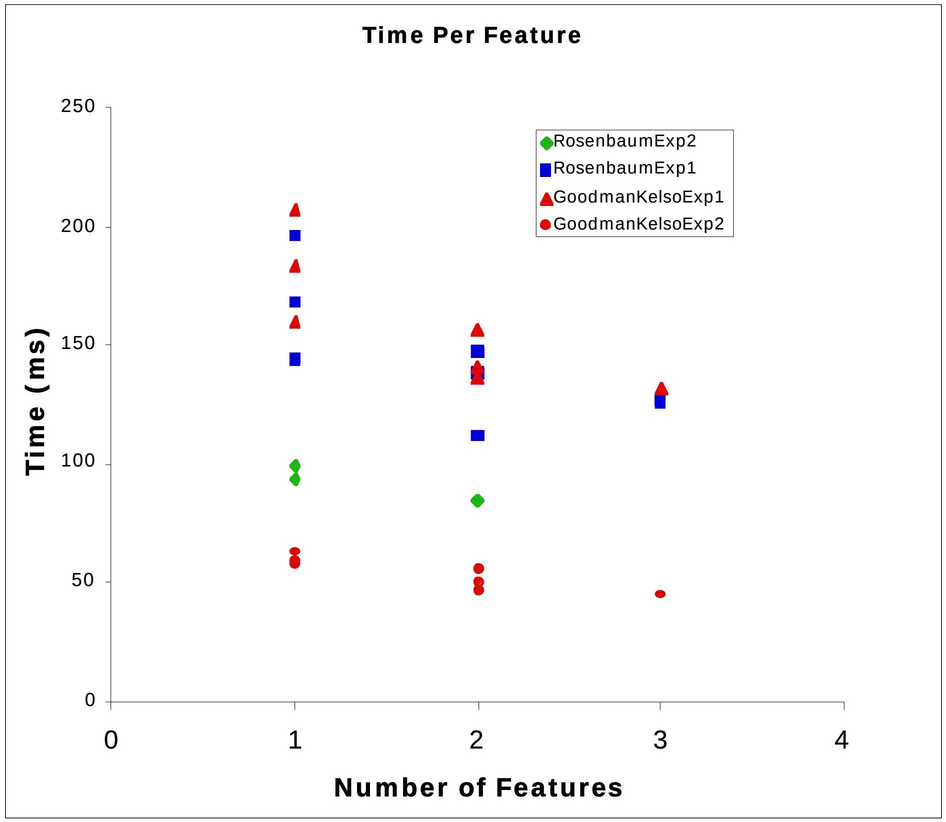


Light panel mounted vertically above horizontal button board.

Rosenbaum vs. Goodman & Kelso Results



Per-Feature Time Depends on Compatibility

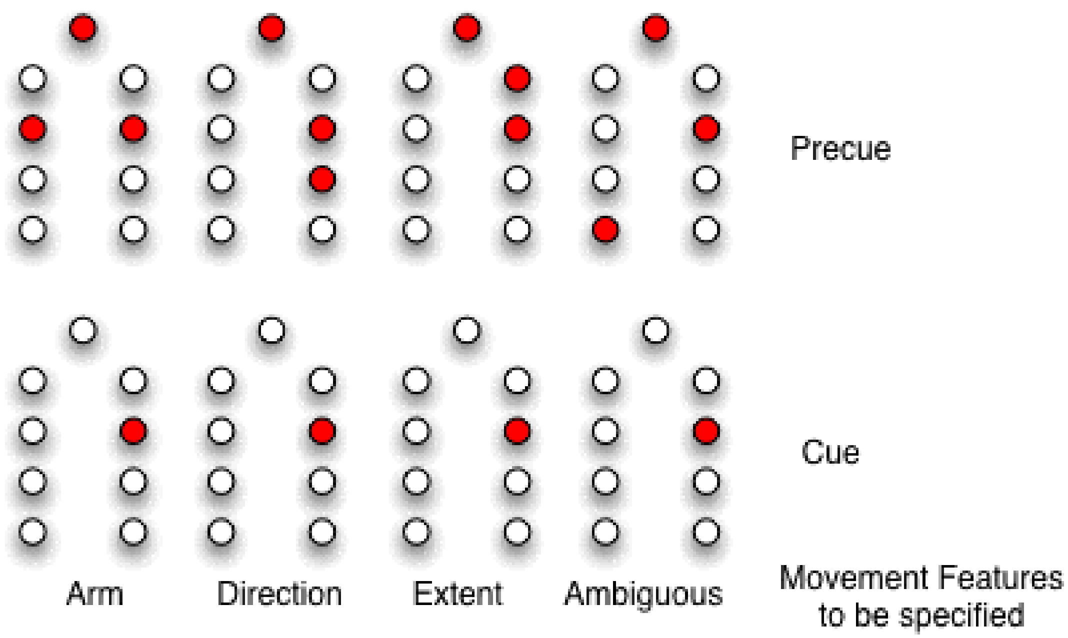


Not supposed to happen! Optimistic interpretation:
50 ms per feature is actual motor time; rest is response selection.

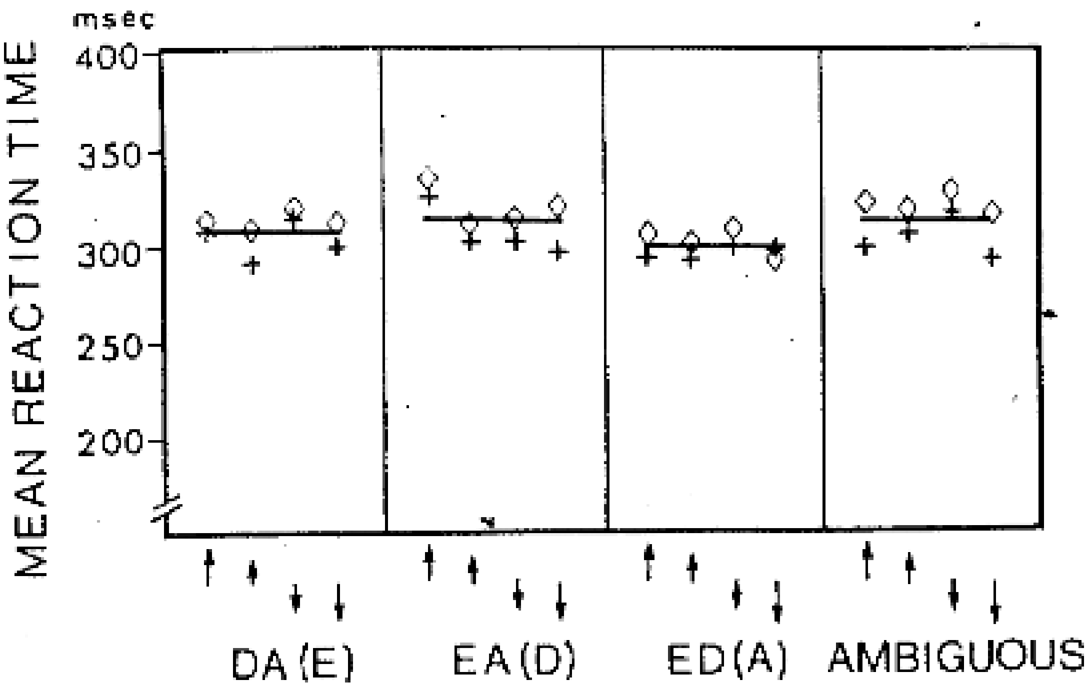
However, S-R compatibility isn't perfect in the task; could the feature time actually be minimum response selection effects?

G&K: Hold Number of Alternatives Constant

Include ambiguous precue: cue both values for the same feature:
Ambiguous features, but specifies possible targets.



Number of Alternatives Dominates



RTs are the same - the set of possible targets is relevant, not the movement features.

This is now looking more like a Hick's Law paradigm, isn't it?

Hypothesis: Rosenbaum's precuing paradigm is actually a "dynamic Hick's Law" paradigm.

Direct correspondence between:

- Number of features that not precued.
- Number of alternatives specified by the precue.

On each trial, specify the number of responses by specifying the possible targets by means of feature precues.

When the stimulus is presented, the response is thus one of the possible targets.

Hick’s Law: RT is linear with log(2) of number of alternatives, with the slope depending on the S-R translation required.

RT for 2, 4, 8 alternatives is thus linear under Hick’s Law.
Ancient result (1952).

Goodman & Kelso compatible task might still require some S-R translation for response selection.

Making Aimed Movements More S-R Compatible

Old result: If the stimulus directly identifies the response (e.g. name digits), then number of alternatives irrelevant, Hick's Law slope is 0.

What happens in aimed movements?

Make stimulus location identical with target of movement - can't get any more spatially compatible than that!

Dassonville, et al. 1999.

Make joystick movements to visible targets with different mappings:

Most compatible: Move directly to target.

Least compatible: Move to target at 90° counter-clockwise from position given by symbolic code (e.g. NW).

Varied number of alternative targets (2, 4, 8).

In highly compatible mapping, no effect at all of number of alternatives on RT. Hick's Law slope is zero.

Latency: about 300 ms.

If S-R pair repeated, no effect on highly compatible mapping, but substantial effect on incompatible mappings.

Suggests that repetition benefit is due to remembering results of response selection, not previous motor features.

Wright (2004): No Hick's Law Effects for Pointing Movements

Stimuli - a set of circles arranged in an arc.

Response - point with stylus to the circle that turns white.

Compared 2 versus 6 alternatives.

Results: Mean RT for N=2 is 254 ms, for N=6 is 257 ms. No Hick's Law effects.

Little or no effect of repetitions - suggests no remembered response selection results.

Compared onset stimuli with equiluminant color-change stimuli: No difference.

Argues against some kind of automatic onset-triggered movement.

Conclusion for Aimed Manual Movements

Motor feature programming effects are most likely a result of translations due to poor S-R compatibility - actually a Hick's Law effect.

Once a visual target has been identified, an aimed movement can be quickly launched to it without any S-R translation or motor feature programming delays.

Eye Movement Programming

Eye Movements: Abrams & Jonides (1988)
Abrams & Jonides Results
Compatible Task, Only Two-alternative Cues
Crawford & Mueller (1990)
Results
Response Selection Effects in Saccades?
Conclusion for Eye Movements

Eye Movements: Abrams & Jonides (1988)

Classic follow-up to Rosenbaum.

But with saccades instead of arm movements.

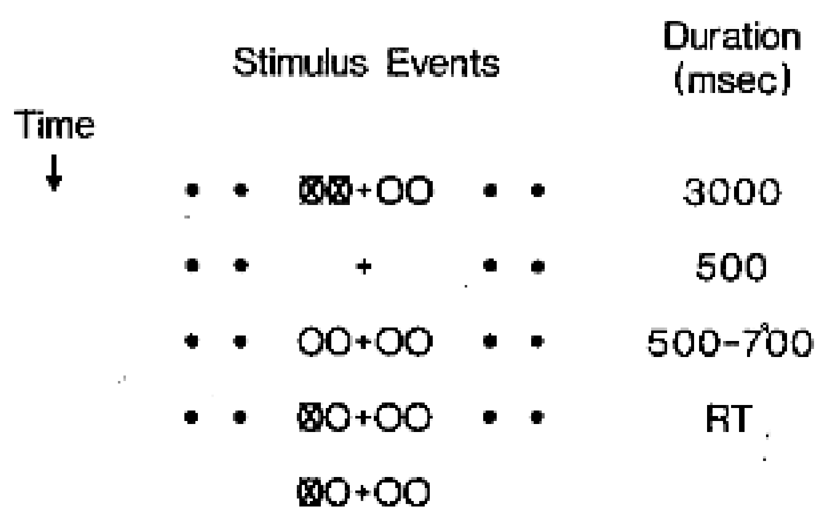
Paradigm:

Anti-saccade: Cue is opposite direction of required movement.

- So cue stimuli are mirror-imaged from required response.

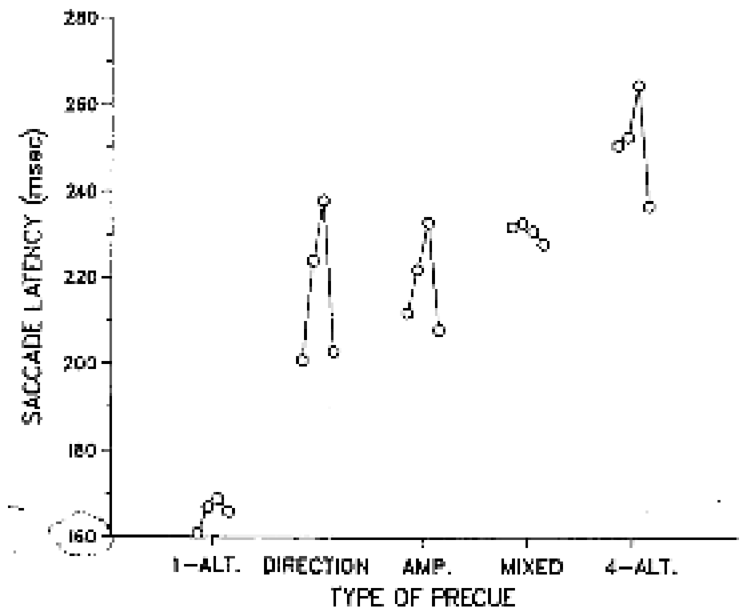
Compare cues of different types:

- Direction, amplitude, ambiguous, uninformative.



Targets are removed as soon as eye movement started.

Abrams & Jonides Results



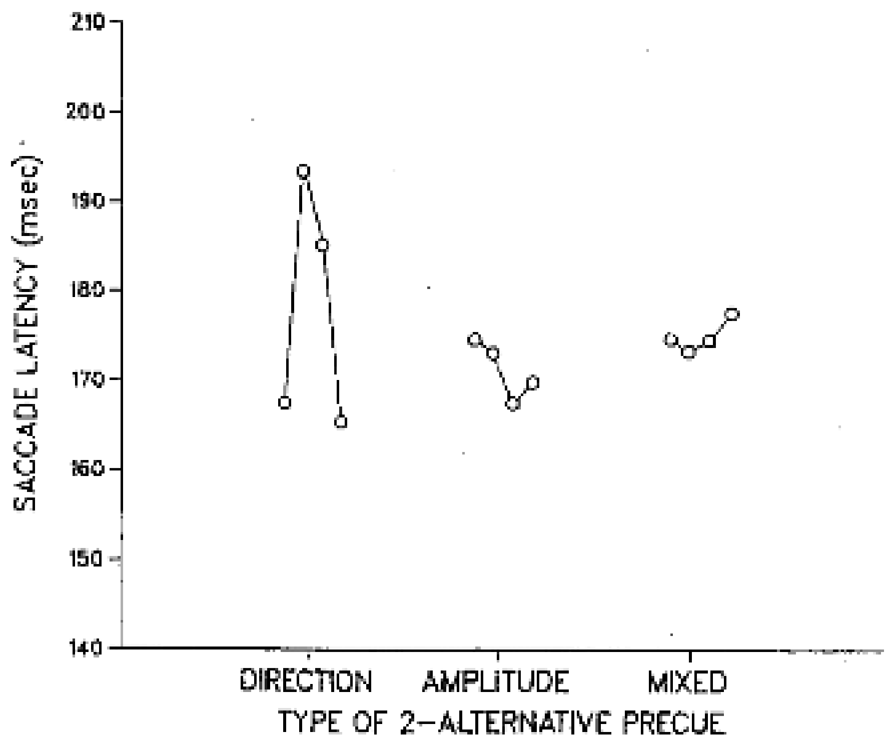
Some movement-specific effects, but 50 ms/feature benefit.
Mixed (ambiguous) cue was 13 ms slower, which was significant.
But should be a lot slower if uninformative.
But anti-saccades mean poor S-R compatibility.

Compatible Task, Only Two-alternative Cues

Cues made compatible with responses.

Mean latencies the same for ambiguous precues as other cues that identify two alternative targets.

Number of alternatives, not number of features, dominates.



Some movement specific effects.

Crawford & Mueller (1990)

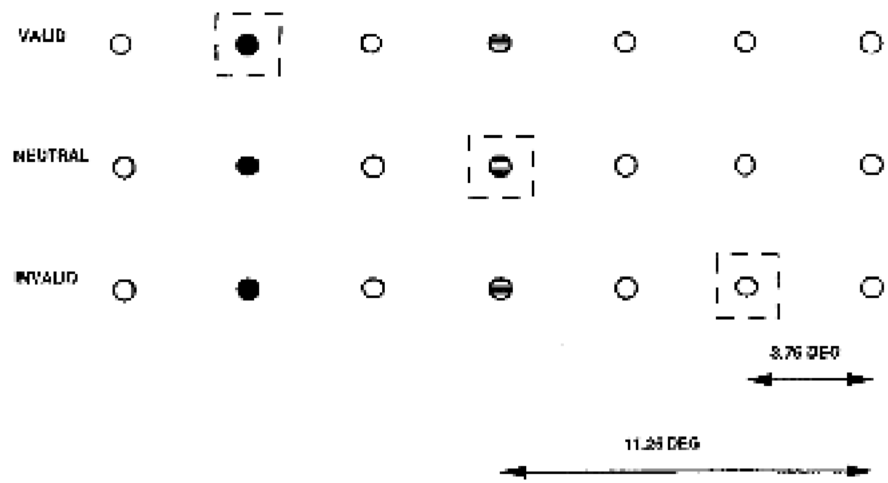
Totally compatible precues and cues.

Cues and targets are lights at the actual target locations.

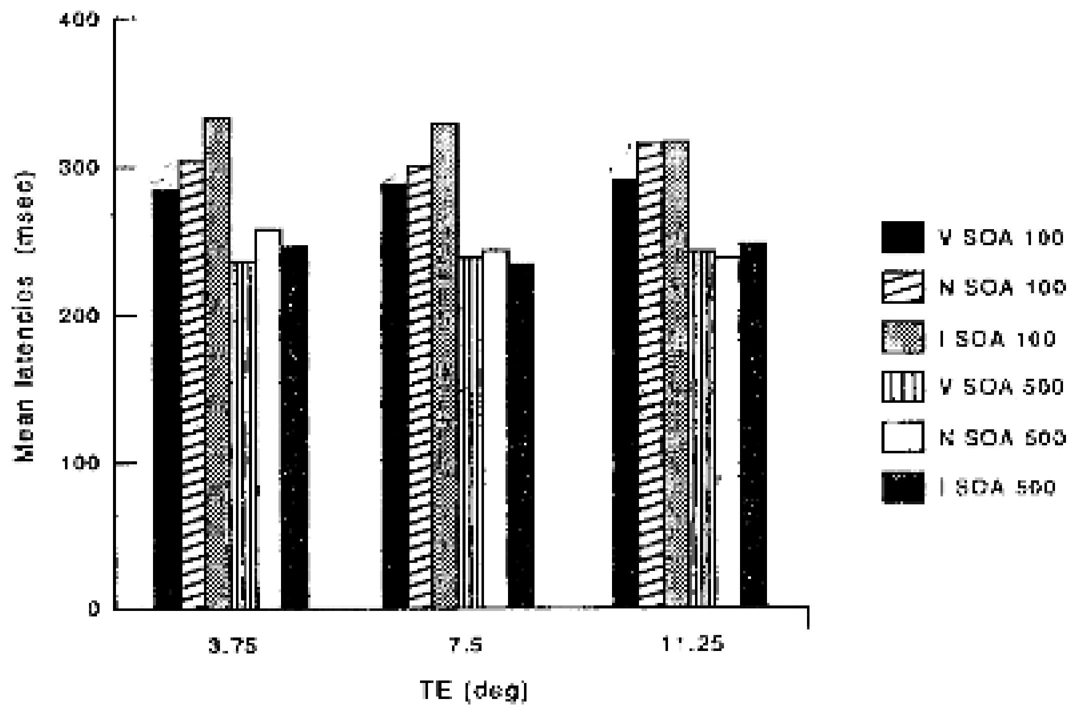
Precue: background lighting a possible target location.

- Valid, invalid; different positions.

Eye movement made to onset light.



Results



At SOA 500 ms, precuing is no benefit, latencies equally fast.
About 250 ms.
Small benefit at SOA 100 ms.

Not expected given motor feature precuing concept.

Due to high S-R compatibility?

Response Selection Effects in Saccades?

Yes, Hick's Law effects if incompatible mapping:
Lee et al. (2005): saccades to remembered locations in a circular array identified by memorized color codes (shades of Rosenbaum!).

No Hick's Law effects if stimulus is co-located with target:
Kveraga et al. (2002, 2005): zero Hick's Law effects in saccades to targets in a circle or semicircle.

- But did get Hick's effects for anti-saccades (look in opposite direction) or key presses to the same stimuli.

Are these "reflex" saccades to target onsets?
Persistent concern in this literature - onset triggers a saccade to it.
Kveraga:

- Stimulus is not full onset, but visible circle is filled in.
- Latency strongly depends on brightness of an onset stimulus, but Hick's Law effect still missing even in targets barely above threshold.

Findlay's results argue against reflex saccades.
All objects onset at once; target chosen cognitively; RT is still only 250 ms.
By usual Hick's Law coefficients, expect it to be a lot slower.

Conclusion for Eye Movements

Same pattern as with aimed manual movements:

Motor feature programming effects for eye movements are also probably Hick's Law effects.

Once a visual target has been identified, an eye movement can be quickly launched to it without any S-R translation or motor feature programming delays.

Architecture Revisions

Recap of Motor Feature Programming Problem
Hypothesis: Primacy of Spatiality in Perceptual-Motor Relations
Changes to Motor Processors
Motor Features in Other Manual Movements?
Golden Oldies: S-R Pair Spatial Effects
Our Results on Fast Learning of Individual S-R Pairs
What is Learned in a Simple Choice Task?

Recap of Motor Feature Programming Problem

Difficulty in matching times in high-speed and complex realistic visual search tasks suggests that motor processors are too slow.

A re-examination of the motor processing literature suggests that the motor precuing paradigm is actually a Hick's Law paradigm and shows effects only if S-R compatibility is moderate or poor.

Guided eye movements and aimed pointing movements have high S-R compatibility.

Therefore, evidence for motor feature programming for these movements is actually weak or ambiguous.

No justification for maintaining motor feature programming if it interferes with model fitting.

- Need to make this decision in order to model high-performance realistically complex tasks.

Hypothesis: Primacy of Spatiality in Perceptual-Motor Relations

Visual, oculomotor, manual (pointing) processors work directly in terms of spatial locations of objects.

When cognitive processor specifies target object to motor processor, motor processor initiates a movement to that location very rapidly.

Fundamental architectural feature:

Perceptual-motor system organized around an integrated spatial system that transforms visual coordinates to motor movements subcognitively and rapidly.

- Evidence that some brain areas do this, e.g. posterior parietal cortex.

Changes to Motor Processors

Current workaround is simple:

Set feature programming time parameter to zero for saccades and aimed manual movements.

Will be made permanent if conclusions still hold after further review.

Effect on previous models:

Movement preparation relevant only if it was on the critical path.

- Some models overlapped preparation with other processes.

If on the critical path, motor preparation time was substituting for some of the response selection time.

- Our claims about simultaneous response selection in dual tasks are actually *stronger* with this revision.

Motor Features in Other Manual Movements?

What about other manual movements, such as key-pressing - are there still motor feature programming effects there?
Not clear yet - have to reassess the literature.

But clear evidence of powerful spatial effects in key-pressing tasks.
Nonhomogeneity of specific S-R pairs.
Role of spatial S-R compatibility.

"Outer" keys are favored strongly over "inner" keys.
Obscure, but good, precedents in the literature for our results.

Golden Oldies: S-R Pair Spatial Effects

Alegria & Bertelson (1970)

Digit - key compatible mapping, index and little fingers over “outer” keys.
For 8 choice task, index and little fingers about 100 ms faster than others (about 500 vs 600 ms).
For 4 choice task, about 50 ms difference.

Welford (1971)

Row-of-lights to row-of-keys compatible mapping.
Similar 100 ms advantage of index and little fingers in 8 choice tasks.

- Similar reduced effects if fewer choices.
- Not biomechanical, not due to stimulus discriminability, but rather ease of identifying spatial position on "outside" of button group.

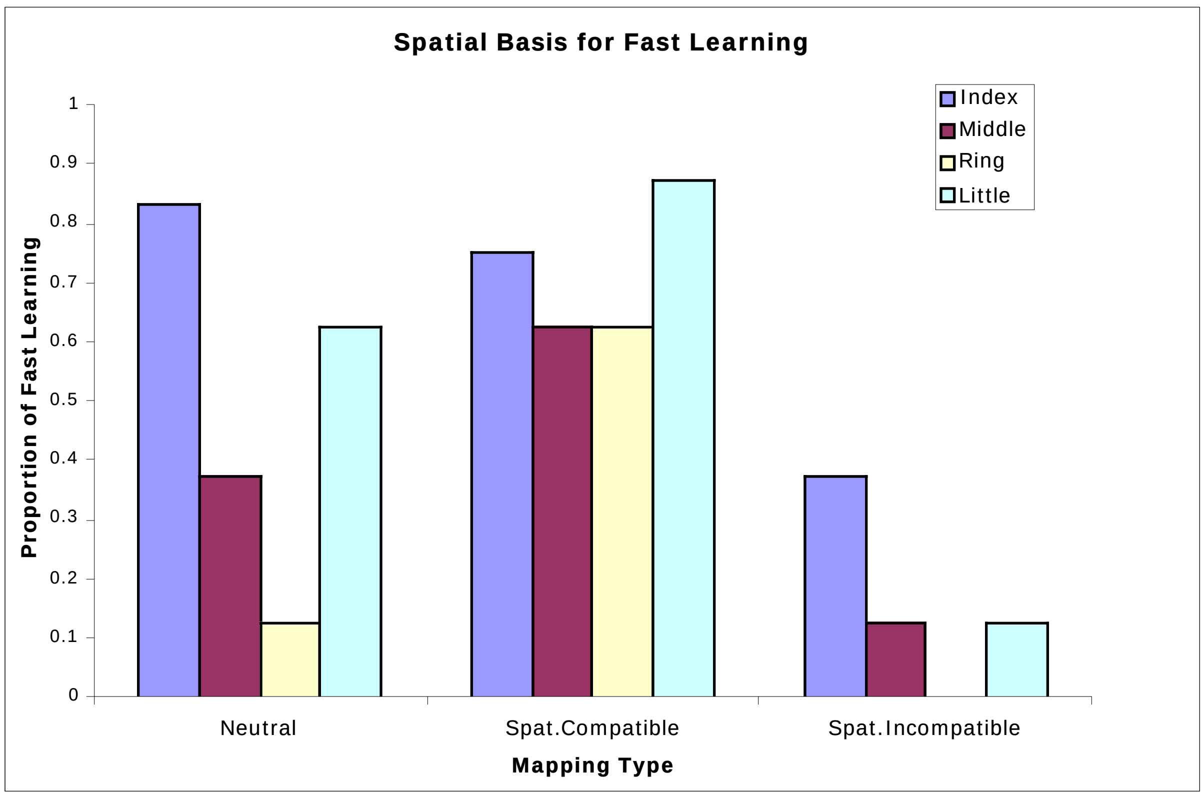
Welford concludes that response localization required:

"... choice reactions do not involve a perceptual-response task so much as a double perceptual task: first to identify the signal and then to identify the position in which a response should be made."

Our Results on Fast Learning of Individual S-R Pairs

Identify S-R pairs that show fast learning in different mappings:
Neutral, spatially compatible, spatially incompatible.
- Neutral is arbitrary - e.g. colors to keys.

"Outer" responses are often learned almost instantly.
Less if mapping is spatially incompatible.
All positions easier if spatially compatible mapping.



What is Learned in a Simple Choice Task?

Hypothetical processes in key-press choice reaction task:

Not:
Single production rule maps from the stimulus to the response finger, motor processor programs movement features.

Instead:
Possibly multiple production rules map from the stimulus to the response key *location*; motor processor quickly programs a move at that location.

Mappings to "outer" locations are faster learned and executed.
Still not clear why, but effects are too strong to ignore.
Fewer production rules involved in the mapping?

Conclusion: Spatial Primacy might well apply even in key-press tasks.
Have to re-assess motor control literature to see if there are non-spatial forms of motor feature programming.

General Conclusion

Implications for Using Empirical Literature in Architecture Development

Don’t Rely on Conventional Wisdom in the Empirical Literature

Implications for Using Empirical Literature in Architecture Development

State of the empirical literature:

Literature is severely compartmentalized:
You can't take just one research community's word for it.
E.g. visual search RT paradigms vs eye-movement paradigms.

New work is constantly appearing:
Issues that you would think were thoroughly explored decades ago are only now being looked at.
E.g. whether Hick's law applies to aimed movements to the stimulus object.

Library's bibliographic search engines help cut across community lines for both old and new work.

Don’t Rely on Conventional Wisdom in the Empirical Literature

Conventional wisdom - explanation for an interesting phenomenon that seems to work well and becomes generally accepted.

But most conventional wisdom in literature is based on verbal theorizing, with only limited quantitative analysis.

Thus conventional wisdom can be seriously wrong.

No substitute for quantitative computational modeling!

Don't believe it until you've seen broad evidence of it in more than one community and paradigm.

Don't believe it until you've modeled it.

Forces careful look at the actual task demands, possible strategies and the quantitative results.