Cognitive Control as a Gated Cortical Network

Jared Sylvester, Jim Reggia and Scott Weems

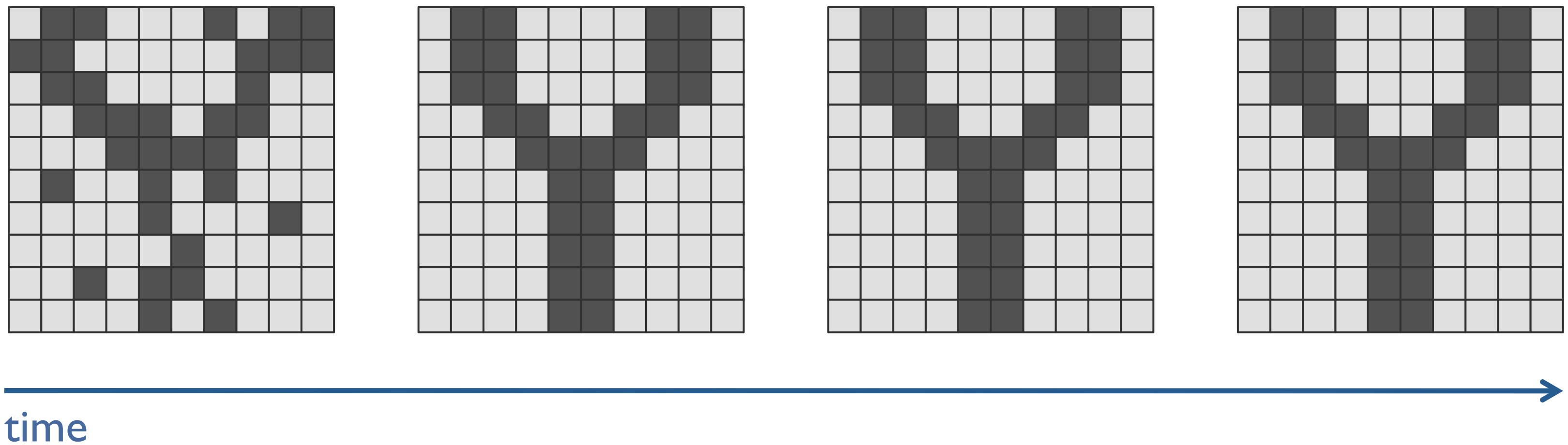
Depts. of Computer Science & Psychology, UMIACS, CASL
University of Maryland, College Park, MD

6 Nov 2011

Goal

- Generalizable model of cognitive control
  - Learned, not hard-wired into architecture
- Base behavior on memory contents
- Two type of memory/learning:
  - Memory of perceptual stimuli
  - Memory of task procedures
- Biological inspiration of our approach:
  - Network of regions, recurrent attractor nets, gating, distributed representations, Hebbian learning

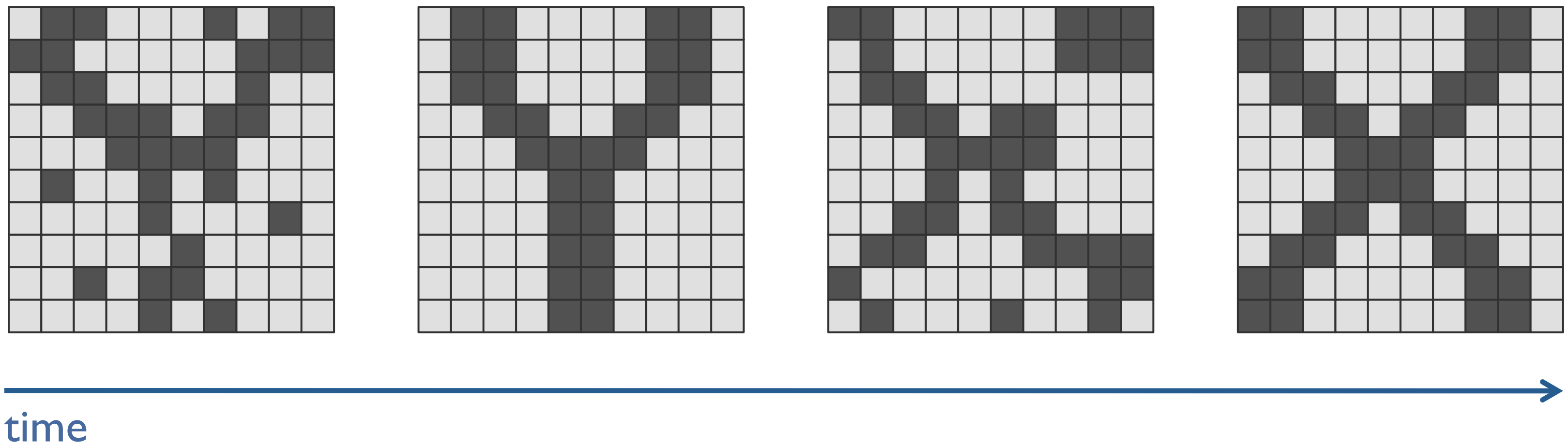
Attractor Net Memories



$$w_{ij}^{t} = w_{ij}^{t-1} + \frac{1}{N} a_i^t a_j^t (1 - \delta_{ij})$$

- Stored patterns are attractors
  - Form auto-associative memory
- But fixed-point attractors
  - Network gets “stuck” in attractor basin

Sequentially Visit Attractor States



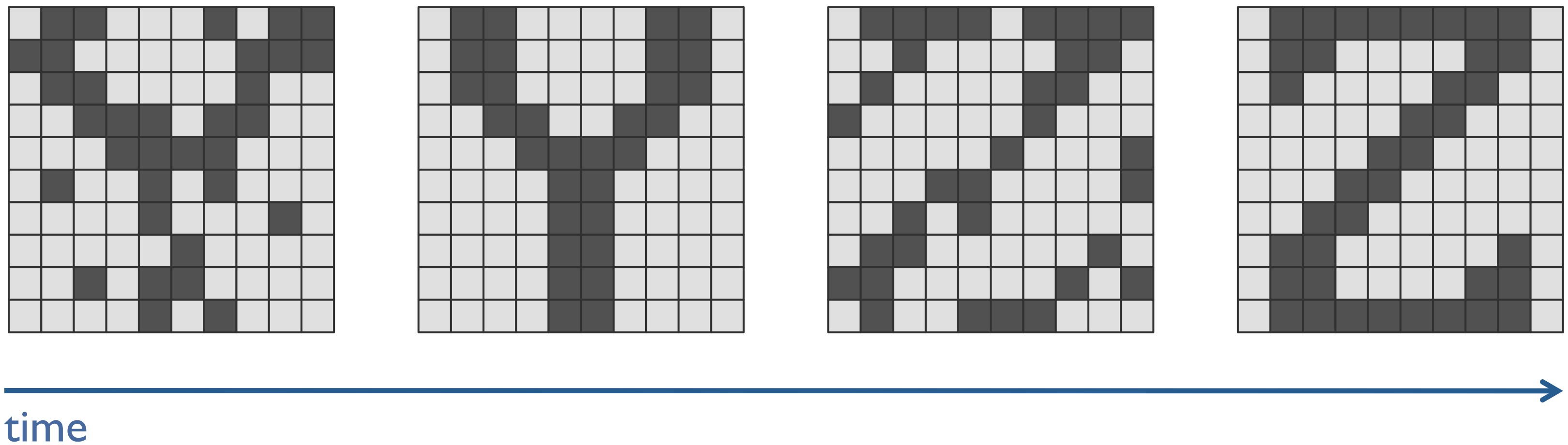
- Dynamic thresholds
  - Increase when node's state remains unchanged
  - Harder for node to stay in the same state

For details, see:

Reggia, Sylvester, Weems & Bunting.
"A simple oscillatory short-term memory model." BICA 2009.

Winder, Reggia, Weems & Bunting.
"An oscillatory Hebbian network model of short-term memory." *Neural Computation*, 2009.

Visiting Attractors in Order



- Asymmetric weights
  - Correlate activity with other nodes' *previous* activity

$$v_{ij}^{t} = (1 - k_D)v_{ij}^{t-1} + \frac{1}{N}a_i^t a_j^{t-1}$$

  - Network transitions between attractors in order

For details, see:

Sylvester, Reggia, Weems & Bunting. "A temporally asymmetric Hebbian network for sequential working memory." ICCM 2010.

Adding Cognitive Control

- Modeled Running Memory Span task
  - Can match human behavioral results
  - But all control was exogenous
- For internal control, use multiple networks
  - Network of attractor networks
  - Controlled by gating
  - Learn processing of sequences

Control Mechanism

- Built around attractor networks
- Trained prior to task beginning
- Directs the model by operating gates
- Core is “instruction memory”

Control Mechanism

- Built around attractor networks
- Trained prior to task beginning
- Directs the model by operating gates
- Core is "instruction memory"
  - Stores sequence of steps to do subtasks
  - Multiple sequences stored simultaneously
  - Divided into cue & response sections

Instruction Memory

Distributed ‘cue’ pattern

Make tea

Instruction Memory

Distributed 'cue' pattern

Make tea

Distributed 'response' patterns

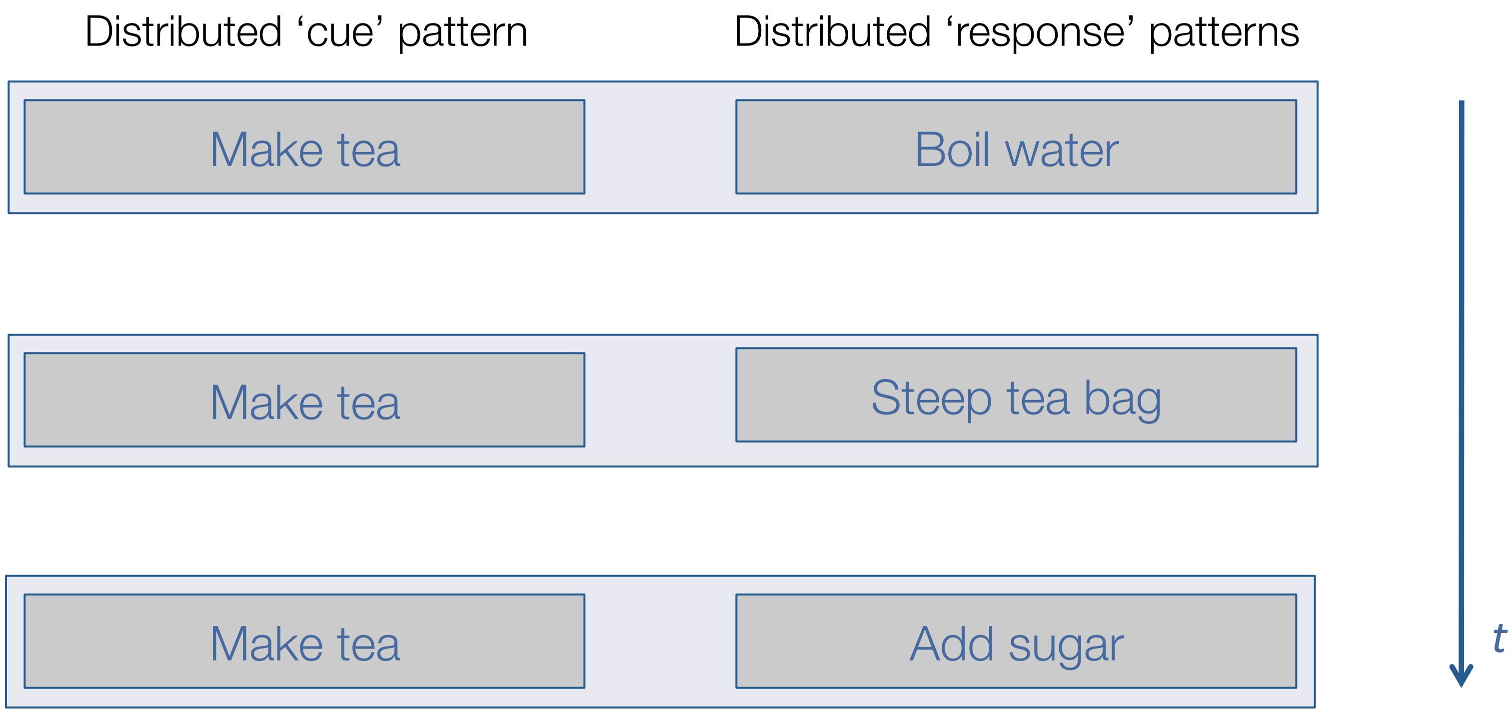
Boil water

Steep tea bag

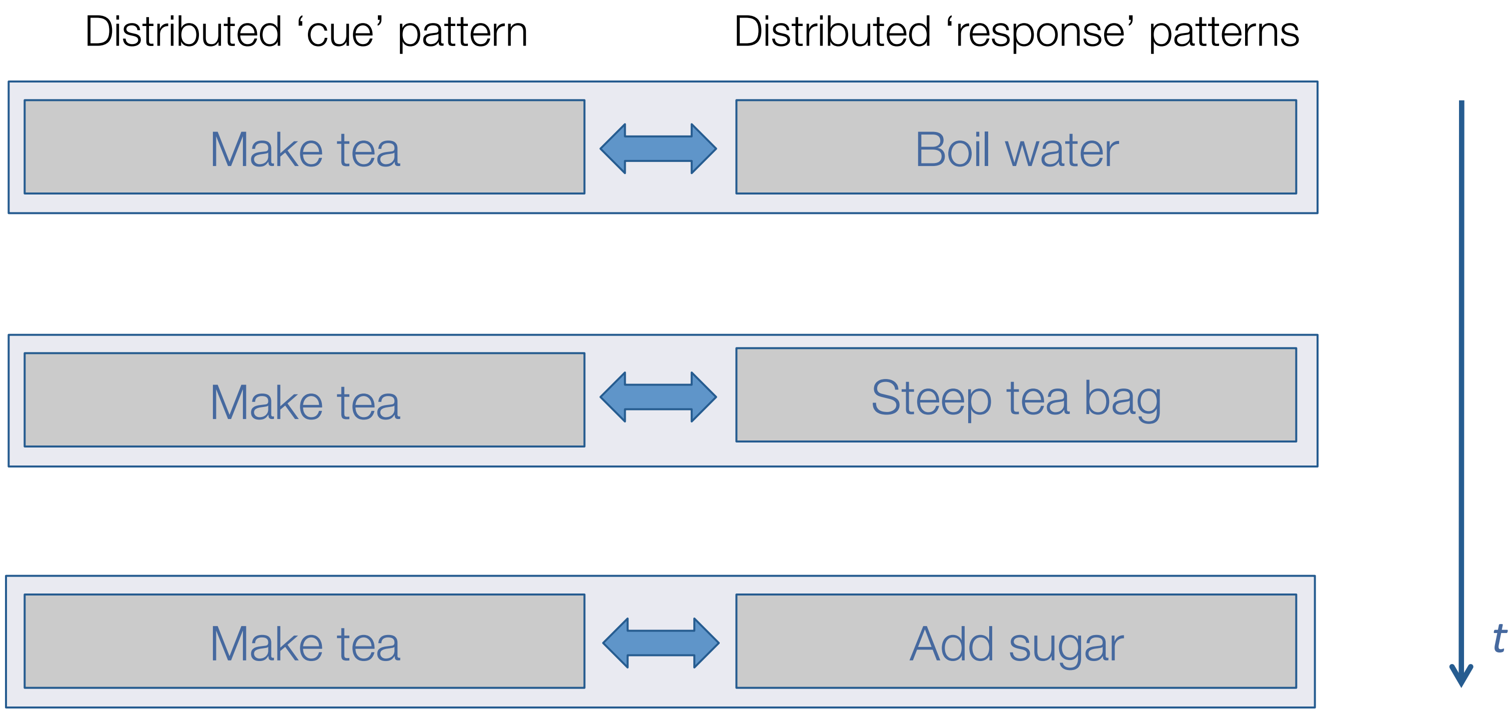
Add sugar

t

Instruction Memory

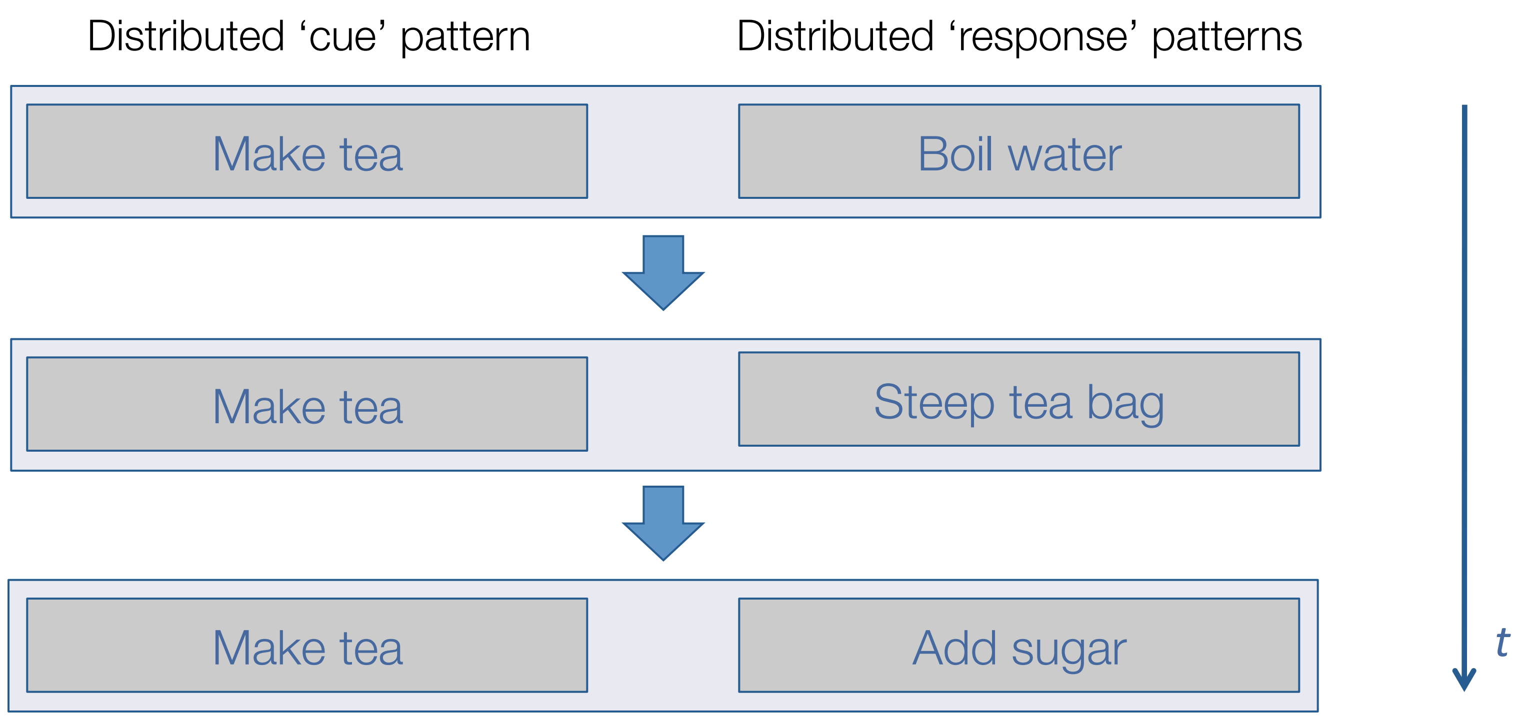


Instruction Memory



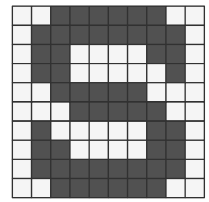
$$w_{ij}^{t} = (1 - k_{\mathrm{CTRL}})w_{ij}^{t-1} + \frac{1}{N} a_i^t a_j^t (1 - \delta_{ij})$$

Instruction Memory

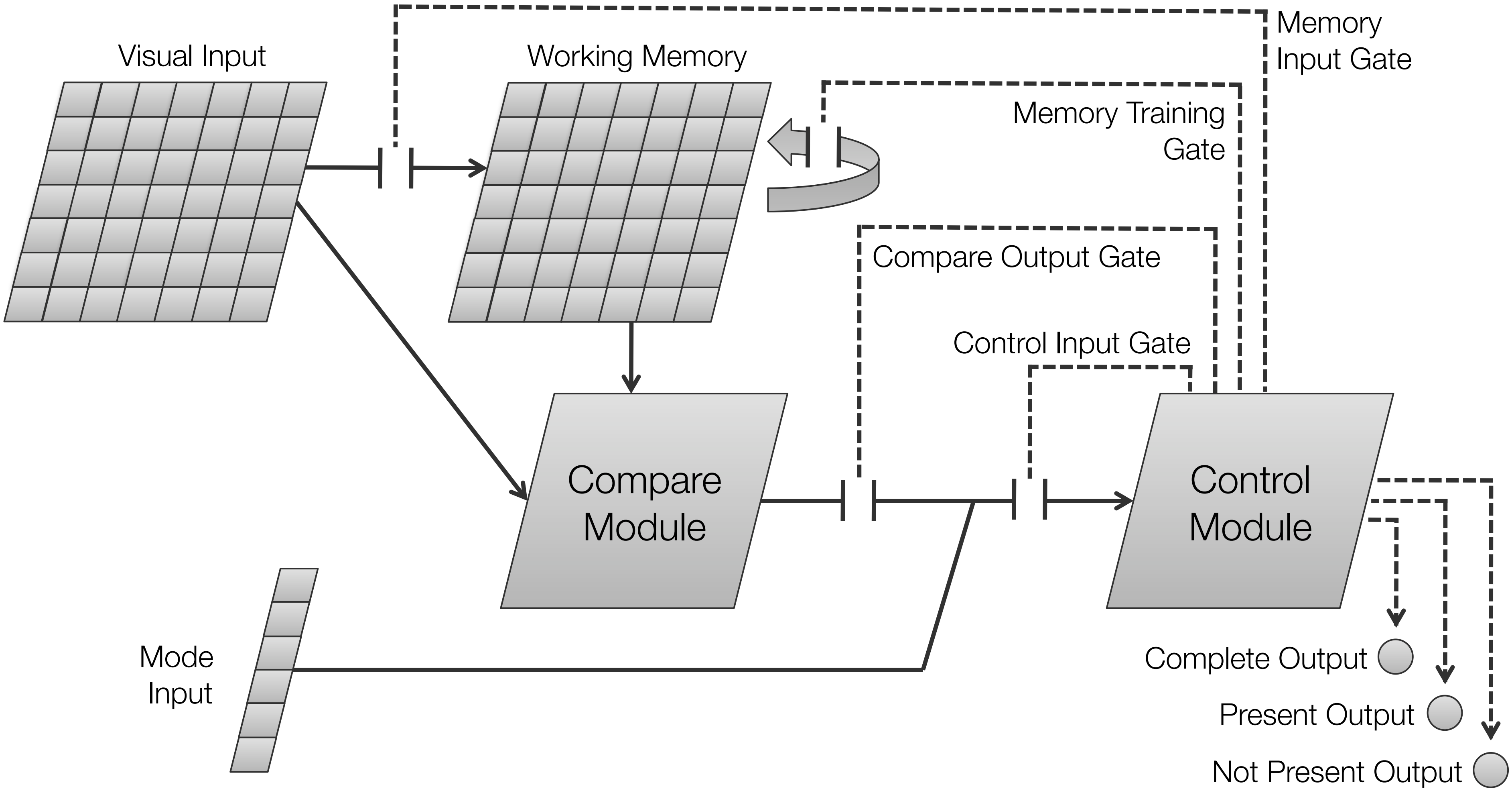


$$v_{ij}^{t} = (1 - k_{\mathrm{CTRL}})v_{ij}^{t-1} + \frac{1}{N} a_i^t a_j^{t-1}$$

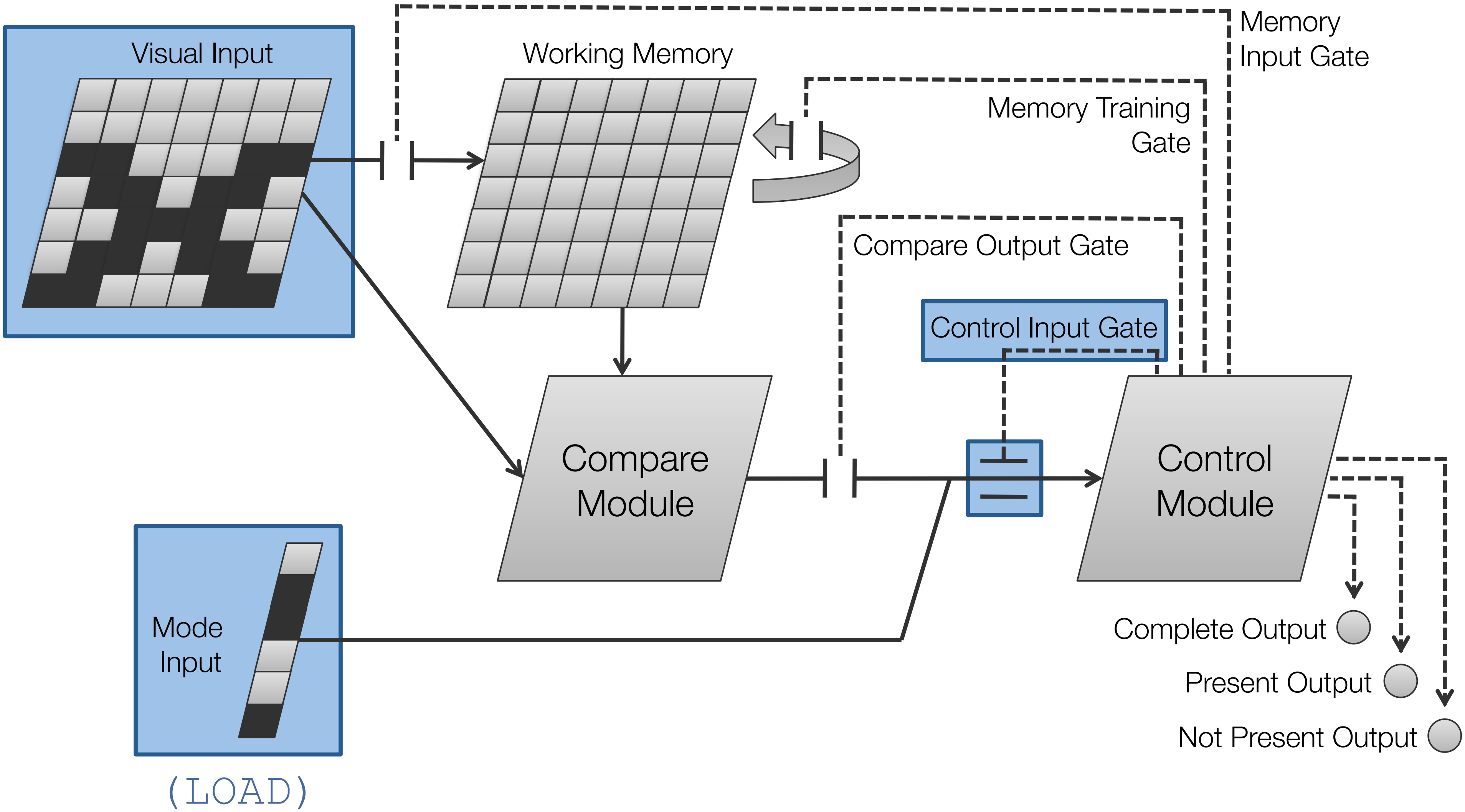
Task: “Store/Recognize”

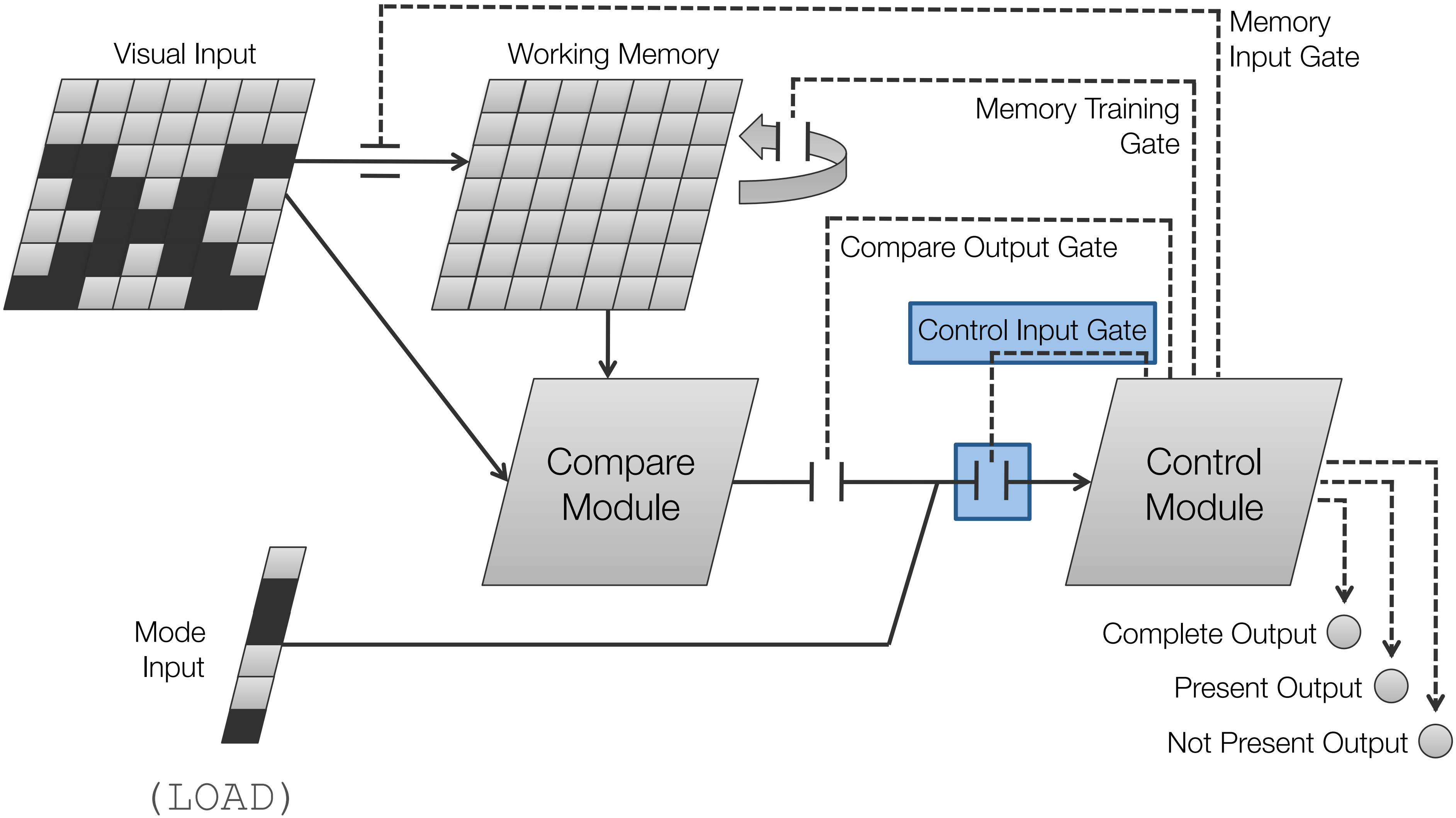
- Visual stimulus 
- Mode input 
  - “load” – add visual stimulus to W.M.; output “complete” when done
  - “evaluate” – is visual stimulus in W.M?
    - If so, output “present.”
    - If not, add it to W.M. & output “not present”

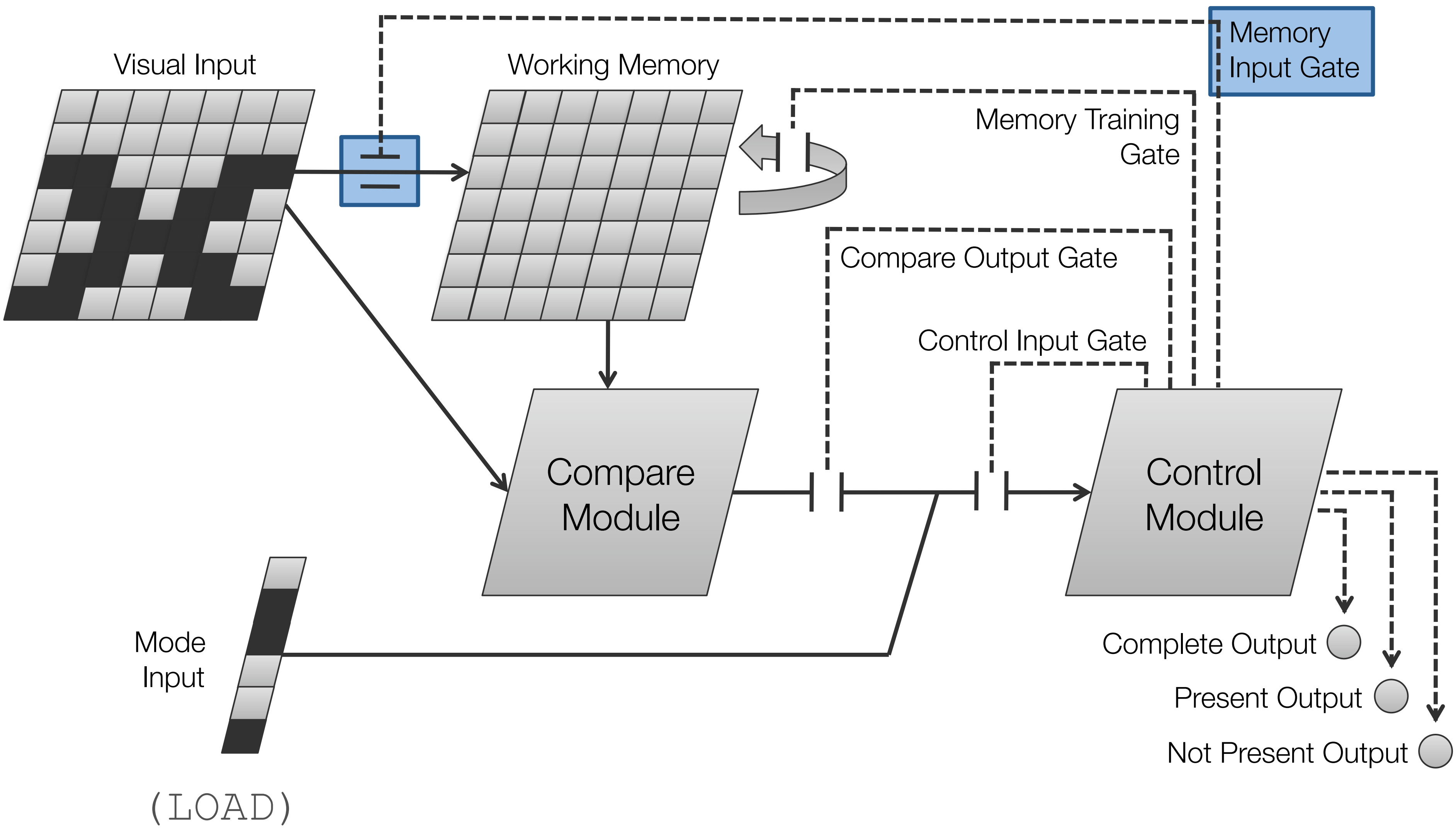
Mode input	load	load	evaluate	evaluate	evaluate	evaluate
Visual input	A	B	A	X	Y	X
Correct response	complete	complete	present	not present	not present	present

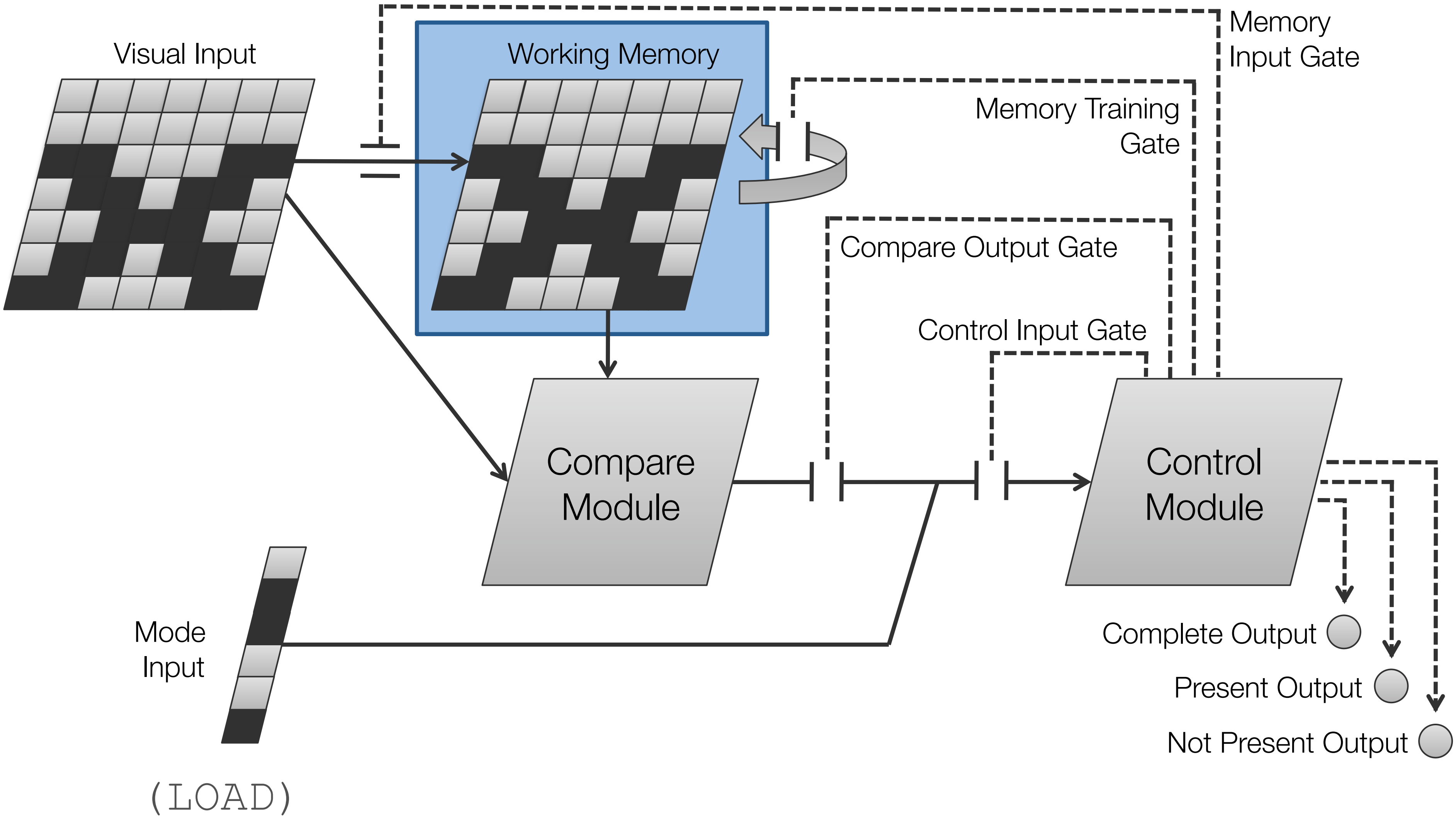


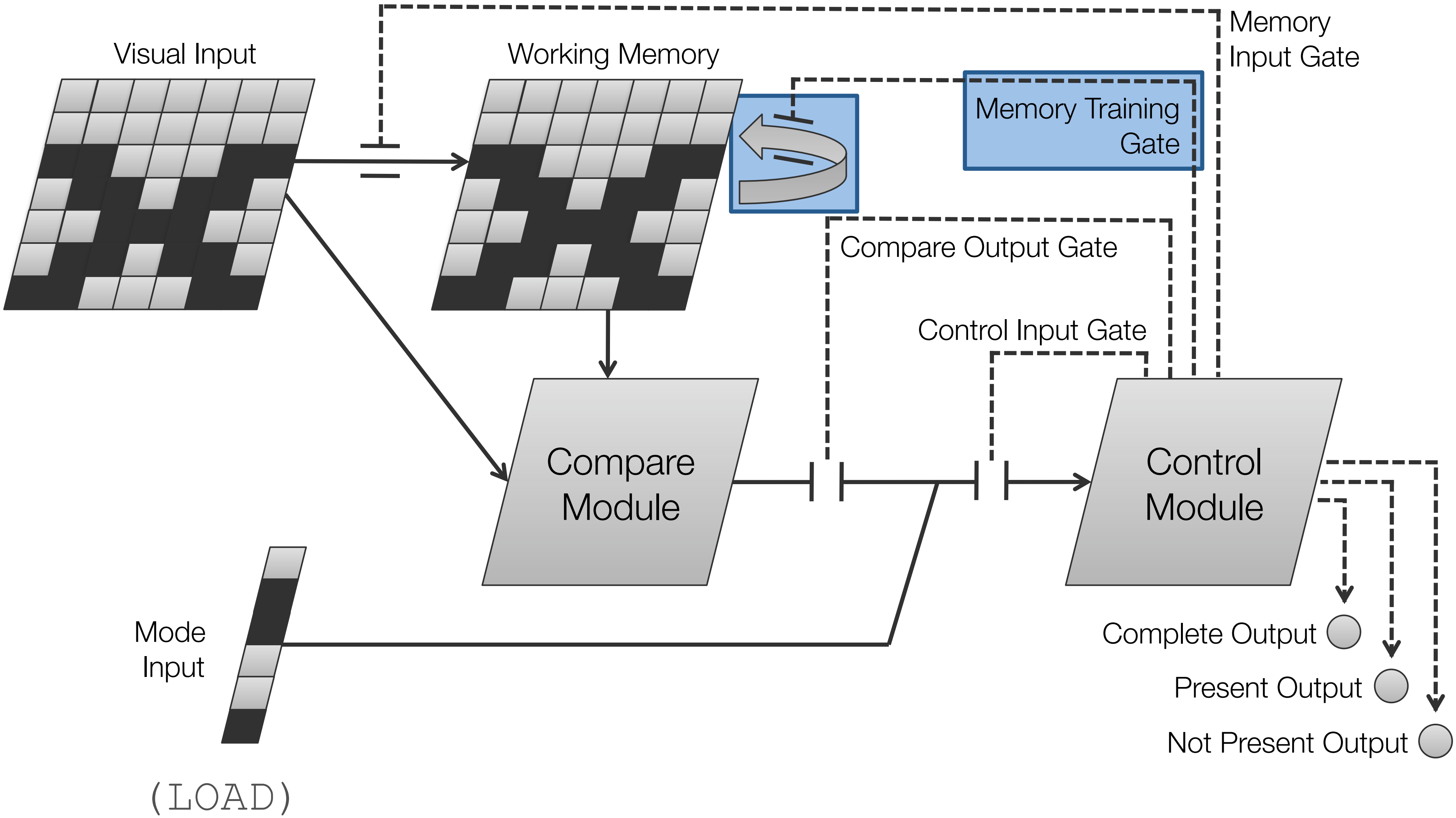
How to add item to working memory


Visual Input
Working Memory
Memory
Input Gate
Memory Training
Gate
Compare Output Gate
Control Input Gate
Compare
Module
Control
Module
Mode
Input
(LOAD)
Complete Output
Present Output
Not Present Output

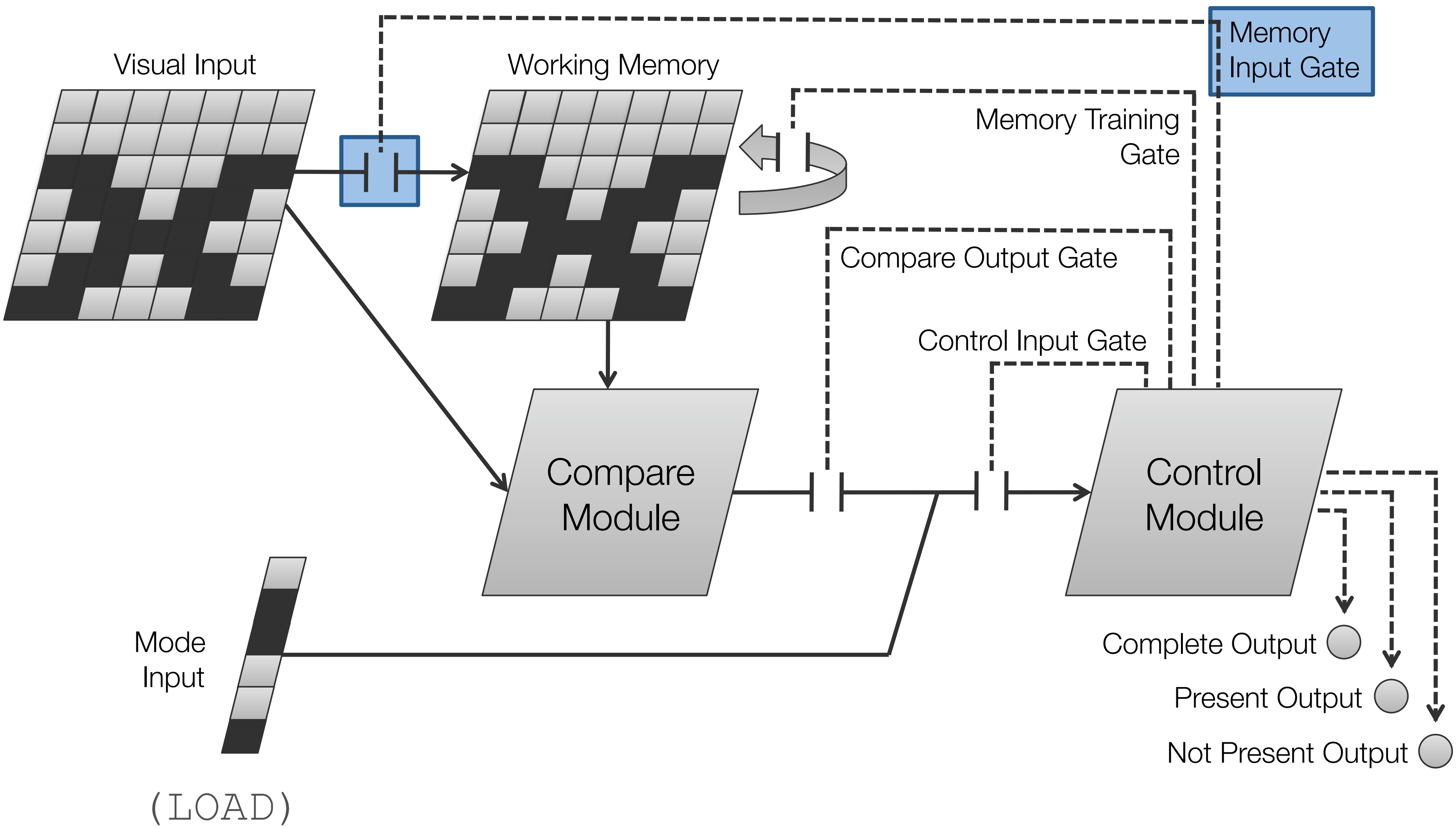

Visual Input
Working Memory
Memory
Input Gate
Memory Training
Gate
Compare Output Gate
Control Input Gate
Compare
Module
Control
Module
Mode
Input
(LOAD)
Complete Output
Present Output
Not Present Output

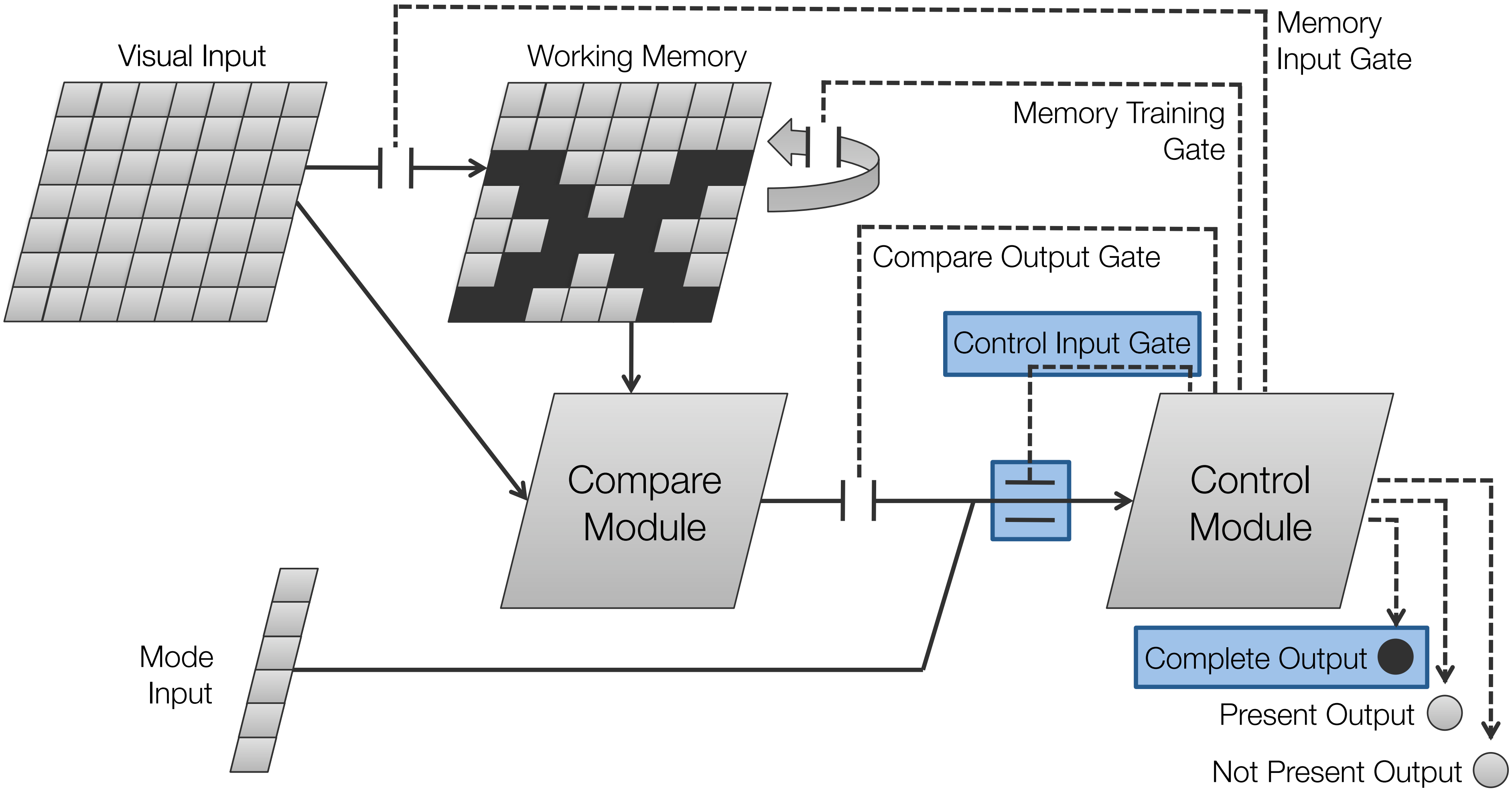

Visual Input
Working Memory
Memory Input Gate
Memory Training Gate
Compare Output Gate
Control Input Gate
Compare Module
Control Module
Mode Input
(LOAD)
Complete Output
Present Output
Not Present Output


Visual Input
Working Memory
Memory Input Gate
Memory Training Gate
Compare Output Gate
Control Input Gate
Compare Module
Control Module
Mode Input
(LOAD)
Complete Output
Present Output
Not Present Output

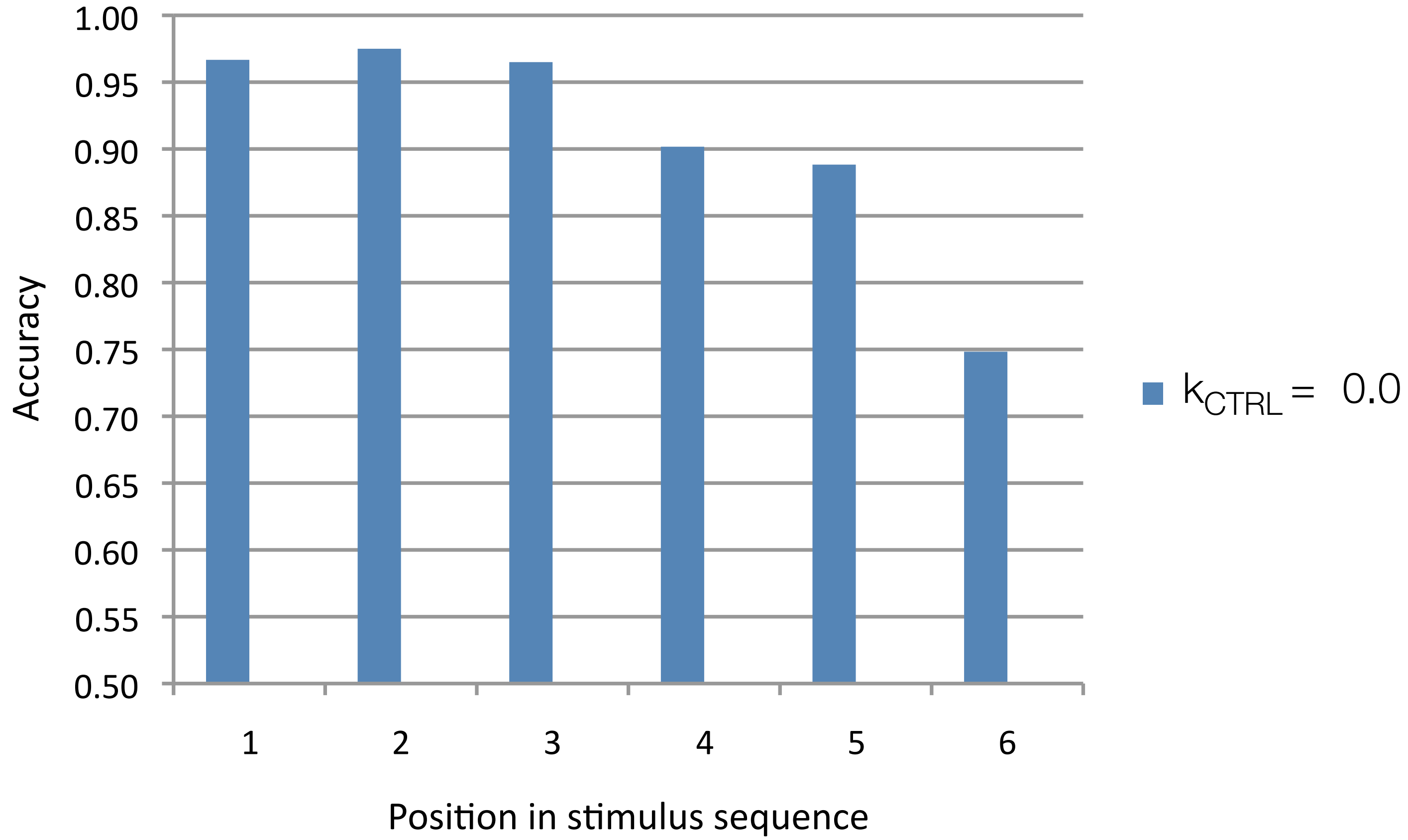


$$w_{ij}^{t} = (1 - k_{\mathrm{WM}})w_{ij}^{t-1} + \frac{1}{N}a_i^t a_j^t(1 - \delta_{ij})$$


Memory Input Gate
Visual Input
Working Memory
Memory Training Gate
Compare Output Gate
Control Input Gate
Compare Module
Control Module
Mode Input
(LOAD)
Complete Output
Present Output
Not Present Output


Visual Input
Working Memory
Memory Input Gate
Memory Training Gate
Compare Output Gate
Control Input Gate
Compare Module
Control Module
Mode Input
Complete Output
Present Output
Not Present Output

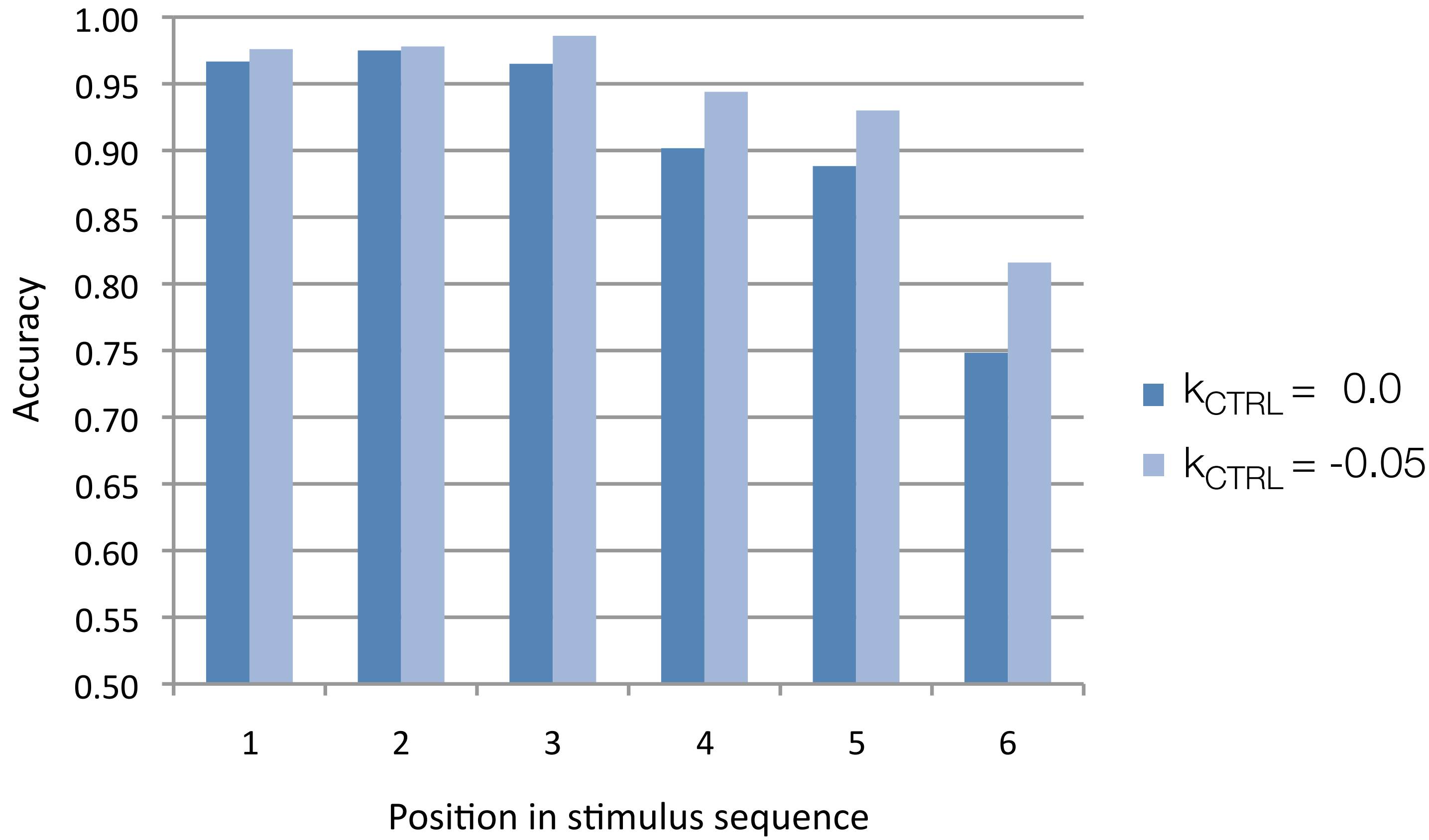
Accuracy with Varying Instruction Memory Decay



k_{CTRL} is the “decay” rate in the controller’s instruction memory layer

Sylvester J, Reggia J, Weems S. Cognitive Control as a Gated Cortical Net, Proc. 2nd Int’l Conf. on Biologically-Inspired Cognitive Architectures, IOP Press, 2011, in press.

Accuracy with Varying Instruction Memory Decay



k_{CTRL} is the "decay" rate in the controller's instruction memory layer

Negative decay (i.e. gain) performs better

Sylvester J, Reggia J, Weems S. Cognitive Control as a Gated Cortical Net, Proc. 2nd Int'l Conf. on Biologically-Inspired Cognitive Architectures, IOP Press, 2011, in press.

Conclusion

- Modeled proof-of-concept task
- Behavior determined by control module's memory contents
  - i.e., learned, not hard-wired
  - *n*-Back model done
- Biologically plausible
  - Network of regions, recurrent attractor nets, gating, distributed representations, Hebbian learning