# Notes on important references for project

## Cueva and Wei, 2018

- Title: "Emergence of grid-like representations by training recurrent neural networks to perform spatial localization
- Published on arXiv.org

### Task of interest: 2D Spatial Navigation

- Involves Entorhinal Cortex (EC)
- *Question*: How do networks of neurons in the brain solve the spatial navigation task?
    - To answer this, we need to find network models that 1) solve the spatial navigation task, 2) are biologically realistic, and 3) make predictions about single-neuron-level activity that are consistent with known experimental observations (i.e. grid cells, place cells, border cells)
    - One proposed solution: Hard-coded models
    - Problem: These models require a large amount of fine-tuning, which isn't biologically realistic, and only predict grid cells, not other types of EC activity
    - Solution proposed by paper: Train RNN to perform path integration under biologically realistic constraints, and EC responses like grid activity and place activity naturally emerge

### Motion simulation

- What do they mean by "modified Brownian motion"?
- How they choose the speed is unclear, but because many of the network units end up having speed tuning, it can't be constant
- Strategy for dealing with boundary: Resampling after colisions
    - How exactly do they do this resampling?

### Network

- Model
    - Two inputs: speed and direction
    - Two outputs: x- and y-position
    - N=100 recurrently connected neurons
    - Activation function: tanh()
    - Network has internal, i.i.d. Gaussian noise
- Training
    - Path length used for training: 500 steps
    - Initialization
    - Input weights: zero-mean Gaussian variables with variance 1 / (number of inputs)

  - Output weights: set to zero
  - Bias: set to zero
  - Recurrent weights: varied based on environment
  - Orthogonal matrix for square and triangular environments
  - Zero-mean Gaussian variables with variance 1.5 ˆ 2 / (number of units)
  - Regularization was very important for results of paper
  - Is this biologically realistic?
  - This ensures that the network encodes position, which shouldn't change if speed is zero. Any dynamics that occur while the speed is zero must only move the network through the null space of the output weights.

**Properties of trained networks**

- Spatial tuning of units
  - This is *not* the same as just considering the input weights for the unit as a linear filter; we need to actually run the network and compute each unit's mean activity as a function of space
- Speed tuning of units
  - Many cells had tuning properties relative to the animal's speed
- Error correction
  - Contrary to intuition, error does *not* seem to accumulate when network performs path integration on paths longer than training length
  - The network seems to perform error correction using boundary interactions
  - This model was proposed by Hardcastle et al., 2015

**Future directions:**

- See if this can work with a more biologically plausible learning rule
- Figure out why grid cells don't have multiple spatial scales
- Understand how the RNN works mechanistically – is it an attractor of some sort, or does it have more complex properties?
  - Tatiana and Chris's work might be able to help with this

## Sorscher et al., 2019

- Another model of grid cells, from Surya Ganguli's lab at Stanford
- Code hosted on GitHub
- Because this code uses PyTorch, it was used as the basis of our code

## Hardcastle et al., 2015

- This paper proposes model of how EC circuitry (and models of that circuitry) can use environmental boundaries for error correction