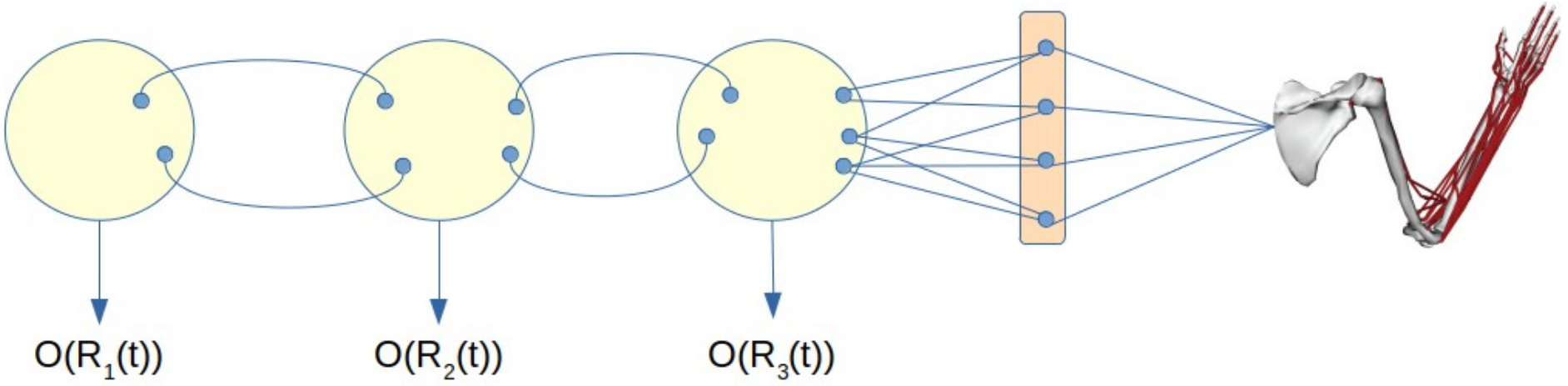
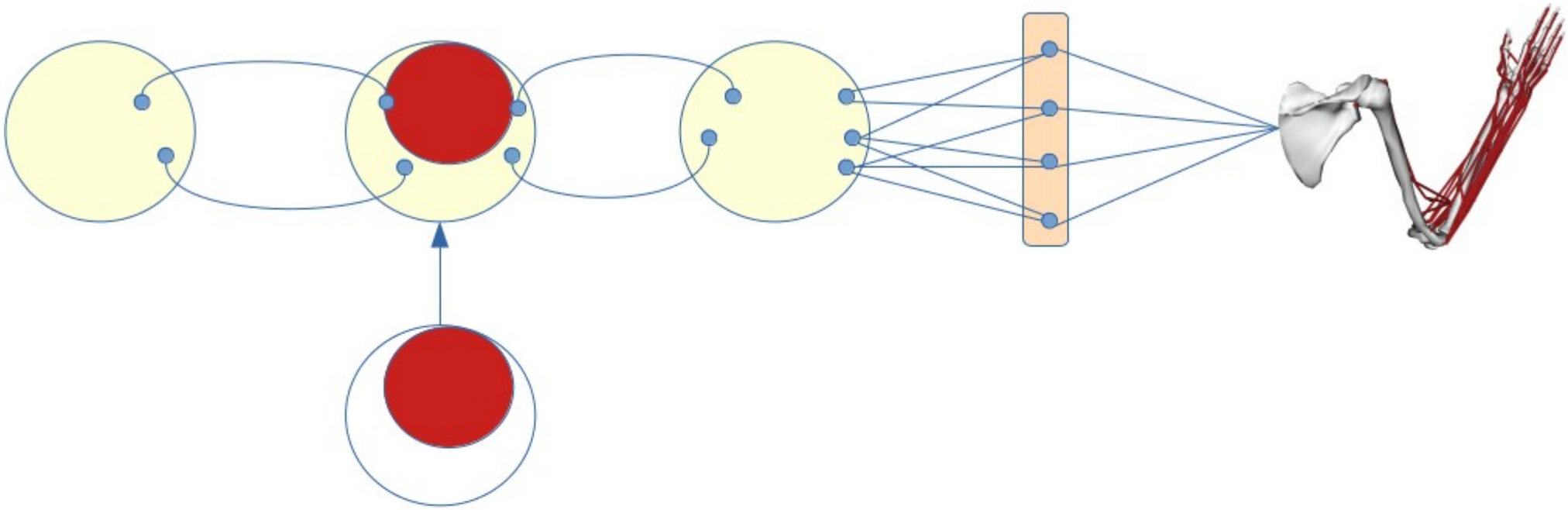
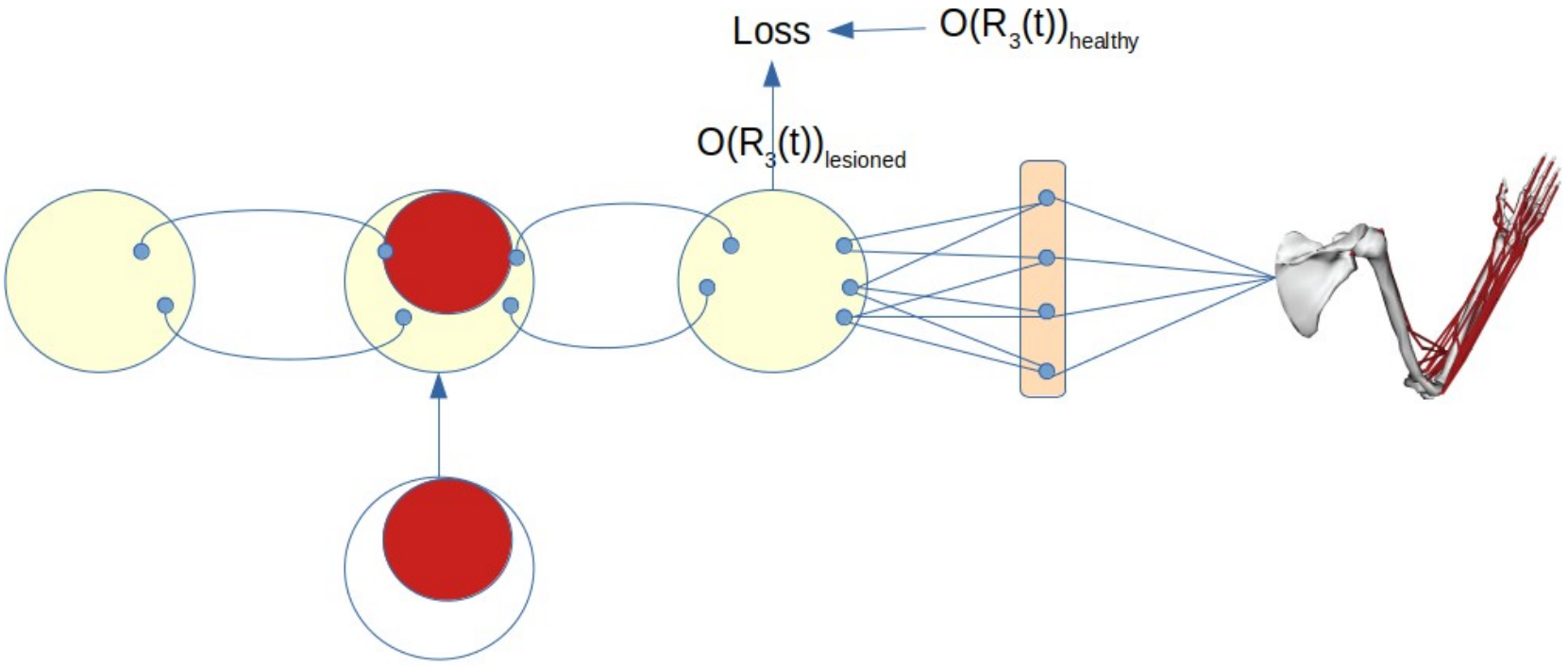


1) Train on simple grasping task
2) Record typical outputs from each module, after passing through an observability model, which is presumably dimensionality reducing (dropping out, time averaging, or spatially averaging units?)

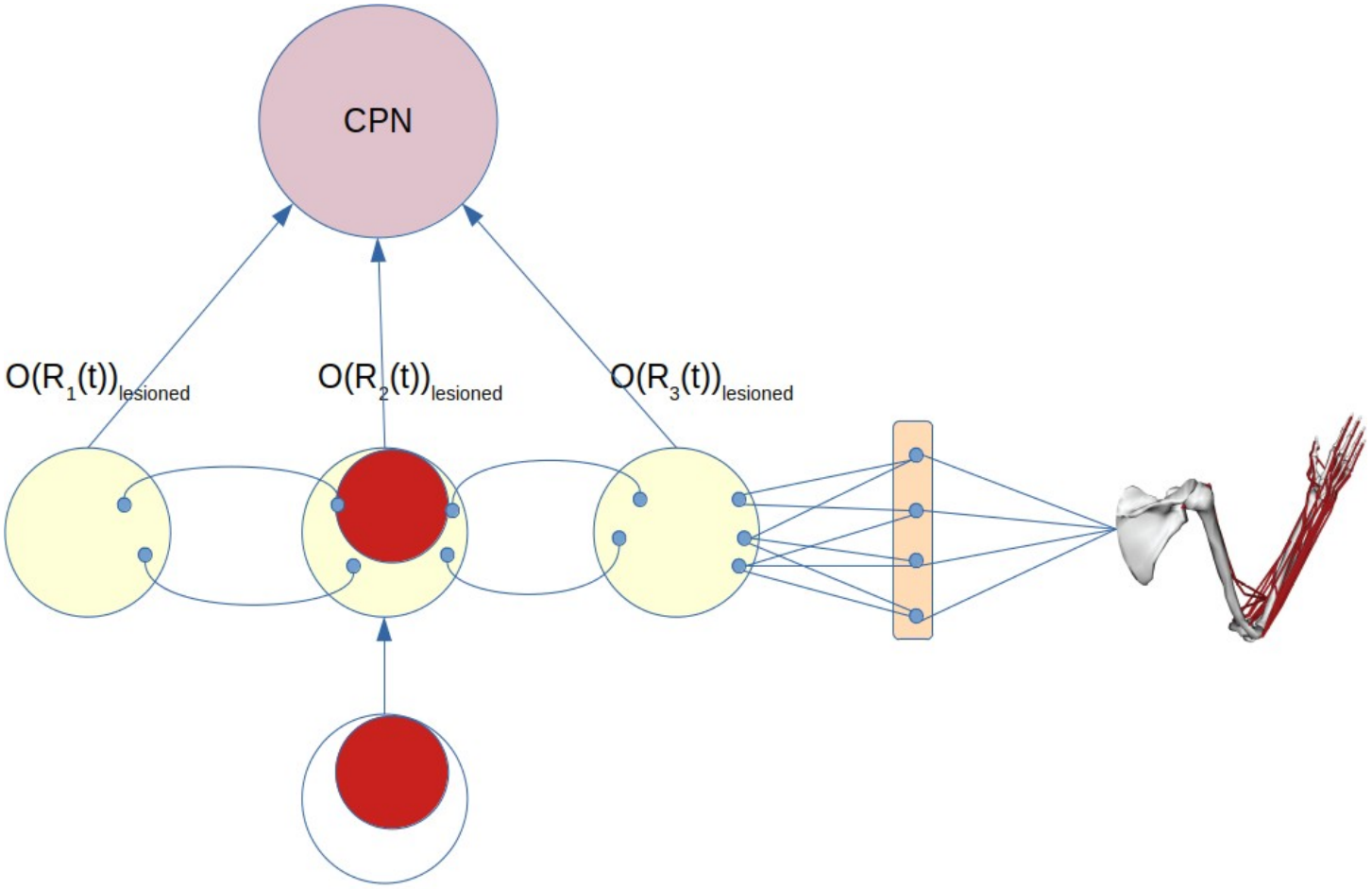


3) Apply a ‘lesion mask’ to module 2, which acts as an additional input to select neurons
    * If we can: identify the functional form of the ‘lesion effect’ s.t. the arm, or module 3 outputs, look “somehow” like a realistically lesioned brain
    * Explore both the mask values and functional forms.
    * If we are exploring just values, not functional forms, perhaps we can learn these?

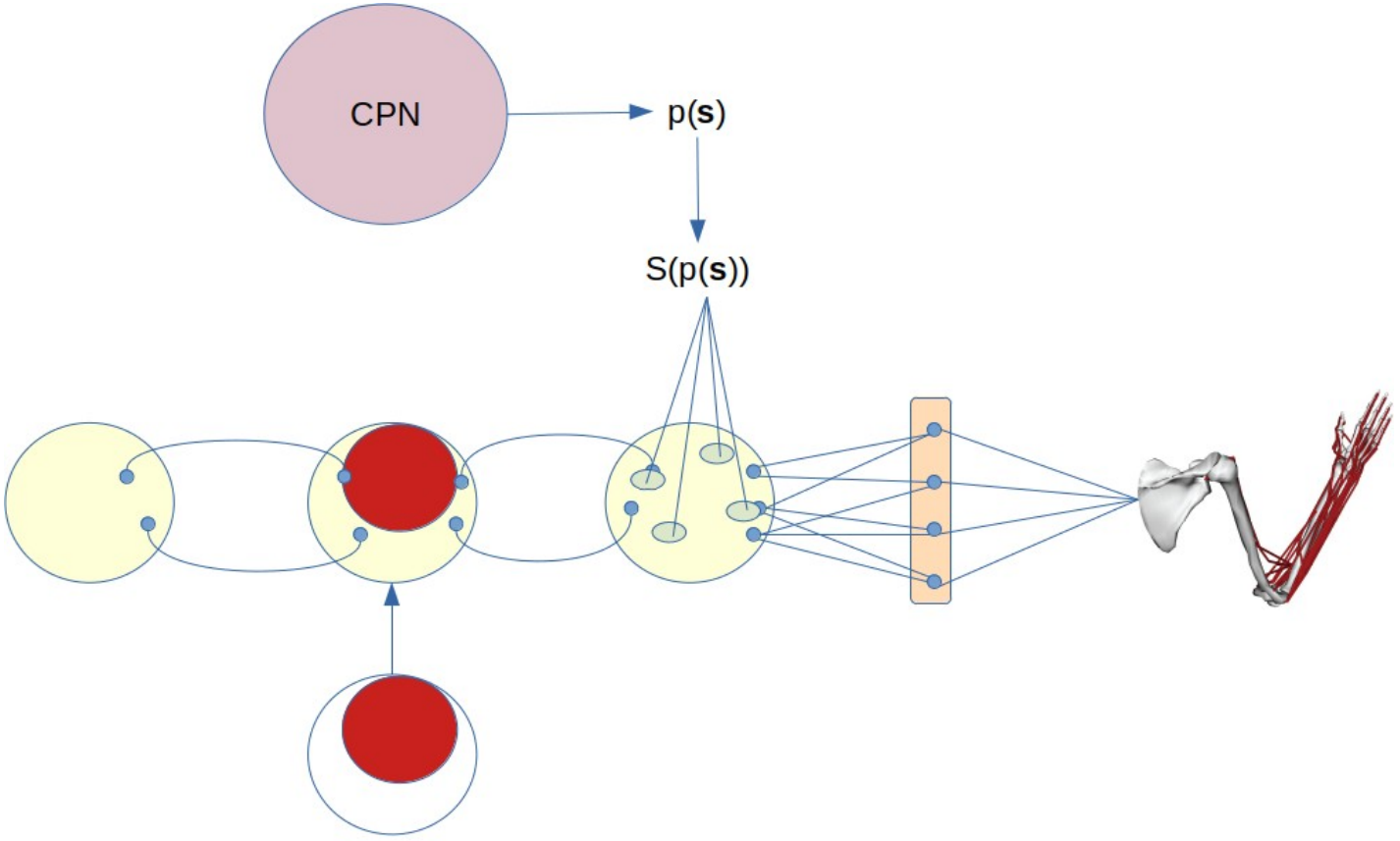


4.) We define two losses:

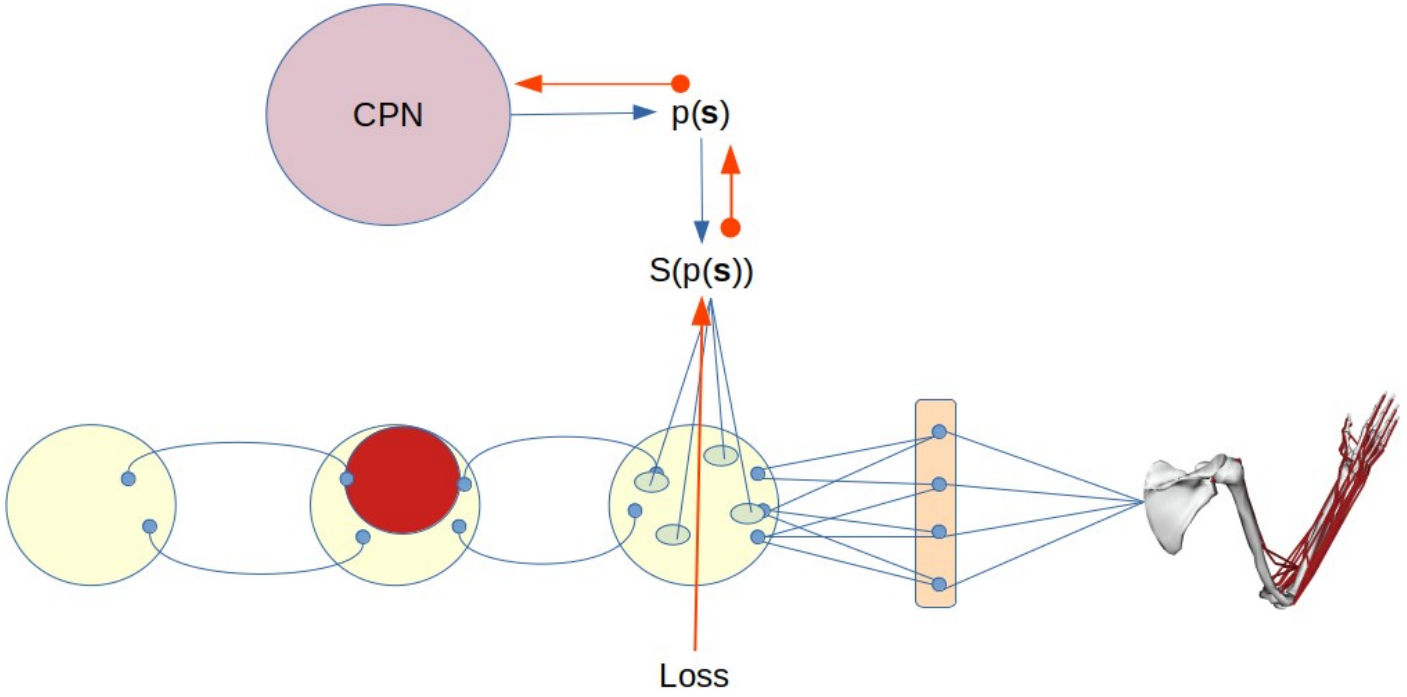
1) Based on the module 3 observed brain state, compared to healthy, within each class
   * O Must be differentiable
2) Based on arm movement, perhaps the simple MSE Loss from Michaels, for simplicity. (Not depicted above)



5.) CPN receives data from every brain area, subject to the same observability model. These three together form the “state”, as the term is used in DPG.



5.) The CPN's policy maps brain state at time 't' to p(**s**), which serves as input to a stimulation function.
6.) The stimulation function S, differentiable, is fixed, and maps **s** onto inputs to module 3's neurons. This should be fixed to some reasonable model of stimulation, e.g. that several electrodes are inserted, each with a 'blast area' of surrounding neurons.



7.) Loss is backpropagated through module 3, to CPN. Modules are not updated. This forms our training. Examine relation to DDPG: we have “examples” in the form of healthy activity for each class, but we don’t know how stimulation relates. Does that mean we make an actor/critic model?
8.) We compare the resulting arm movement after CPN is trained, to measure outcome, but we are not optimizing for it directly. Thus this is a brain-to-brain coprocessor.