Learning Simultaneous Sensory and Motor Representations- (SESEMO)

Berkeley Kids

***Abstract*—**

Redundancy reduction [1], edge detection [3] and hierarchical representations [5] have been the main stay for lot of work in computer vision and vision neuroscience to represent sensory data. But, these representations do not directly lend themselves to action. It has been argued elsewhere [6] that what the brain probably codes for are an organism's sensorimotor contingencies.

Work by Philipona and O'Regan [7, 8] showed that the dimensionality of the manifold of compensable actions is smaller than the dimensionality of the sensory or motor spaces. How then can we go about building a model that models the manifold of the sensory space but with knowledge of it's relationship with the motor system?

In this work we wish to explore this problem by introducing two ideas

- Learning a basis space that represent motor actions
- Joint estimation of both sensory and motor representations
- Representations that are flexible

In over actuated motor systems (high degrees of Freedom) training a controls model becomes challenging. While, there are many approaches of training such a model, we are interested in a framework that can help learn the dynamics of the many control systems an agent may have for many tasks. We propose that this can be achieved when a global objective is minimized simultaneously that learns both sensory and motor systems. An added advantage of such representations is that, once learnt, the problem is no longer solving an optimization problem but inferring coefficients on a learnt basis. This mimics biological plasticity [4] more closely and we argue that for a robotic agent, a hierarchical control system with some low-level reflex circuits that are modulated by high level control systems that are task dependent can effect a more versatile system.

We take the specific case of percept stabilization as an example for the agent to have an invariant representation of an image (with or without objects). Through this example, we hope to motivate that a new class of models can be imagined for learning sensorimotor representations.

I. Model

I is the Image that falls on the retina (say). From this image, we learn our sensory representation **S**. From **S**, we infer our representation α. We then apply a matrix transformation $\mathcal{G}$ that takes us from the sensory space to the motor actuator space. Here, we learn a motor representation **M** and the coefficients β are not inferred but are a transformation based on the sensory estimate. The transformation $\mathcal{F}$ lets us go from the Motor actuator space to transforming our self, so that we can compute an error estimate.

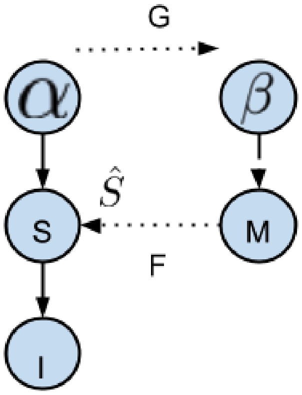

Fig. 1. A loosely defined graphical model that describes the problem sensorimotor representations.

In our case, we define the objective to be tracking a point light source in a virtual world. We are given a camera plane and two motors that control the camera. Further, we parameterize the motor space. That is, to move the camera in the x-y plane all we have to do is apply varying powers to the two different motors. In stead of solving for the appropriate amount of power at each time step, we put forward an idea of motor representation for controlling the camera. This motor representation is a collection of tuples that specify the power to the two different motors.

$$\min_{S,\alpha^t} \|I - S\alpha^t\|_2 + \lambda\|\alpha^t\|_1 \tag{1}$$

$$\min_{G,M} \|dx - (center + \mathcal{F}\beta_{t+1}^T M)\|_2 + \|G\|_2 \tag{2}$$

$$\textit{where} \tag{3}$$

$$\beta_{t+1} = \alpha_t G \tag{4}$$

The way we go about solving the various parameters is by solving the problem in stages. The interesting idea with this model is that the sensory representation can be learnt independently as well with little loss of generality. For the rest of this work, we will not describe talking about learning the Sensory representations and focus on trying to learn the Motor basis with coefficients that are not inferred but are transformed (G) versions of inferred sensory coefficients. For our experiments we also fix our Sensory basis (S) to be a bunch of vectors that point in eight different directions that the light source may have traveled from the previous frame.

An important point to note is that our model is not completely without supervision. For the model to learn the consequences of it's motor actions we use a teacher signal (dx) that tells the system how the motor actions effect the state of the system. Another point of interest is that we begin our system with the agent not knowing the dynamics of it's control system. That is, it does not know what forward/backward/left/right are but by minimizing a global objective function it learns how to control it's motor to attain specific trajectories.

The data in this case are 10000 samples of a point light source moving according to the following equation

$$r = a + bcos(k\theta), where\ \theta = [-\pi, \pi] \tag{5}$$

$$x = rcos(\theta) \tag{6}$$

$$y = rsin(\theta) \tag{7}$$

$$\tag{8}$$

To solve (equation 2.2) we take our current estimate of α and multiply it with matrix G (which is either learnt or fixed as a random matrix). This is our estimate of β. We then solve for M,G using BFGS as the solver. The model learns pretty quickly and takes only a few 100 samples to converge. We then test our model on a different value of k in the above equation for about 100 samples.

The results of our experiments are summarized in the table below. Row 1 describes the experiments with learning only the Motor basis. Row 2 describes experiments with learning both the motor basis M and the sensorimotor transform G. Row 3 describes experiments with learning both the Motor basis (M) and the sensorimotor transform G but with an extra constraint on minimizing the norm of G. From our experiments, this was the best model. Systemic errors occurred during the sharp turns that we see in the data, we hypothesize that with a more sophisticated planning component in the motor planning part of the model, this can be overcome. The motor space learnt tiles the set of directions we would want the camera to move in, increasing the over completeness resulted in a more granular representation of the 3D space.

A Comparison of Models

Model	Euclidean Error
M	68.01
M,G	13.95
M,$\|\|G\|\|_2$	0.58

TABLE I
THE TABLE DEPICTS THE ERRORS WE SEE WHILE LEARNING VARIATIONS OF THE MODEL.

II. Discussion

We present preliminary work in trying to explore a model that tries to learn a motor representation to track an object based on its sensory representations. We show that a simple model can be learnt without having to learn the effects of the motor actions on the sensory space directly, which is a high dimensional, non-linear and complex space. We argue that by posing problems in this way with a global objective, we can explore various parts of the control space effectively because it is bounded by the task and the sensory representation. Similarly, the sensory representation that is learnt (not done in this work) is also directed towards action as opposed to a representation that is purely for redundancy reduction (compression) or to ask questions such as object recognition.

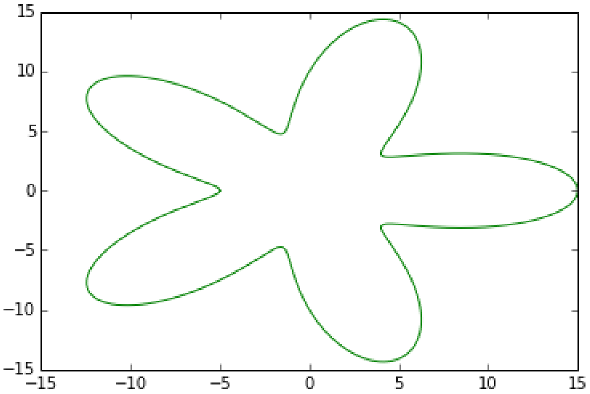

Fig. 2. A loosely defined graphical model that describes the problem sensorimotor representations.

We note that, our model as it stands, is still nascent. Extensions of this model would be along the following lines

- Sensory representations are spatio-temporal filters along the lines of Cadieu and Olshausen [2]
- One can then repose the problem of object tracking as minimizing motion energy. That is, we want to minimize the derivative of the phase coefficients of the model, so that a constant action can be chosen so as to stabilize the percept
- Reposing the problem in the above way also changes the role of $\mathcal{F}$. $\mathcal{F}$ is no longer a transformation of the self but a prediction of the percept from the motor system i.e. $\mathcal{P}(\alpha^{t+1}|\beta^{t+1})$. We then compare this estimate with $\mathcal{P}(\alpha^{t+1}|\alpha^{t}, S)$.
- A model such as the above is a loopy graphical model which can be quite challenging to train. But, a model such as the above is also biologically plausible with sensory systems making motor predictions and vice-versa. Further, a representational motor space can be more efficient to work with once trained.

III. Acknowledgments

Discussions with Jitendra Malik and Tony Bell motivated a lot of this work as well. Special thanks to Pavan Ramkumar, Northwestern University for helping clarify a lot of ideas through discussions. Special thanks to Bruno Olshausen for urging us to think about sensorimotor representations. We would also like to thank our respective funding agencies - Fulbright Scholar Program. NGA. NIH. UC Berkeley.

All the code for this project can be found on our github[1] link.

References

[1] Horace B Barlow. Possible principles underlying the transformation of sensory messages. *Sensory communication*, pages 217–234, 1961.

[2] Charles F Cadieu and Bruno A Olshausen. Learning intermediate-level representations of form and motion from natural movies. *Neural computation*, 24(4):827–866, 2012.

[3] David H Hubel and Torsten N Wiesel. Receptive fields and functional architecture of monkey striate cortex. *The Journal of physiology*, 195(1):215–243, 1968.

[4] Eric R Kandel, James H Schwartz, Thomas M Jessell, et al. *Principles of neural science*, volume 4. McGraw-Hill New York, 2000.

[5] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, volume 1, page 4, 2012.

[6] J Kevin O'Regan and Alva Noë. A sensorimotor account of vision and visual consciousness. *Behavioral and brain sciences*, 24(05):939–973, 2001.

[7] David Philipona, J Kevin O'Regan, and J-P Nadal. Is there something out there? inferring space from sensorimotor dependencies. *Neural computation*, 15(9):2029–2049, 2003.

[8] David Philipona, Jk O'regan, J-P Nadal, and Olivier Coenen. Perception of the structure of the physical world using unknown multimodal sensors and effectors. In *Advances in neural information processing systems*, page None, 2003.

[1] github.com/rctn/sesemo