

Learning of motor maps from perception: a dimensionality reduction approach

Ankit Awasthi, Sadbodh Sharma, Amitabha Mukerjee (aawasthi, sadbodh, amit@cse.iitk.ac.in)

Department of Computer Science and Engineering, IIT Kanpur

Abstract

The role of perception in sighted infant motor development is well-established, but what are the processes by which an infant manages to handle the complex high-dimensional visual input? Clearly, the input has to be modeled in terms of low-dimensional codes so that plans may be made in a more abstract space. While a number of computational studies have investigated the question of motor control, the question of how the input dimensionality is reduced for motor control purposes remains unexplored. In this work we propose a mapping where starting from eye-centered input, we organize the perceptual images in a lower-dimensional space so that perceptually similar arm poses remain closer. In low-noise situations, we find that the dimensionality of this discovered lower-dimensional embedding matches the degrees-of-freedom of the motion. We further show how complex reaching and obstacle avoidance motions may be learned on this lower-dimensional motor space. The computational study suggests a possible mechanism for models in psychology that argue for high orders of dimensionality reduction in moving from task space to specific action.

Keywords: motor development, perception-action models, dimensionality reduction, reach learning

Introduction

That the apparently random arm movements by neonates may have a role in terms of learning visuo-motor control is a position that has gained strength in recent years (A. Van der Meer, Weel, & Lee, 1995; Adolph & Berger, 2006). Visual control of arm movement is evident almost immediately after birth, when neonates appear to struggle with artificial handicaps to keep their arm in the field of view, such as when small weights are tied to their arms, or when they appear to be purposefully keeping their arms within a narrow beam of light (A. van der Meer, 1997).

The emphasis on motor development over the last few decades, arising from the linkage between perception and action that has driven new theories of dynamic cognition (Thelen, Smith, Lewkowicz, & Lickliter, 1994), has further highlighted this connection. Indeed, there is evidence to suggest that the very representation of action has some relation to mental imagery which may provide the prospective structure for organizing motions (Caeyenberghs, Wilson, Van Roon, Swinnen, & Smits-Engelsman, 2009). Studies on monkeys report that a large majority of neurons implicated in the reach planning area in the posterior parietal cortex, encode location in eye-centred coordinates (Batista, Buneo, Snyder, & Andersen, 1999). In addition, nearly half the neurons in the ventral premotor area - thought to be implicated in visual grasp planning - are responsive to eye-centered image data (Mushiake, Tanatsugu, & Tanji, 1997). It is thought that the eye-centric neural computations work together with other body-centric and arm-centric systems (Graziano, 2011) but much of the planning is thought to invoke eye-centered or visual images of the arm.

However, the exact mechanisms by which the complex high-dimensional visual stimulus is used for motor control of the arm, remain unclear. While there have been many computational studies of different aspects of motor development, several involving robots with embodied cameras (Beltrán-González, 2005; Hoffmann, Schenck, & Moller, 2005), such works usually address the question of managing the complexity and dimensionality reduction only peripherally (Jordan & Wolpert, 1999). A particularly interesting approach was the modeling of reaching and grasping via neural networks (Kuperstein, 1991), and a related attempt to map the workspace using self-organizing maps (Saxon & Mukerjee, 1990). However, these works do not deal directly with image models.

On the other hand, another body of work in robot motion planning attempts to construct paths through a workspace that are optimal in various senses of reducing path length, kinematic smoothness, dynamic smoothness, energy costs, etc. A class of efficient methods are based on sampling the motion space, which may also be rather high-dimensional, though of the order of tens, not thousands as in the image space. Such stochastic approaches include probabilistic roadmaps or branching trees of possible paths through the free space. However, these algorithms, some of which also have cognitive ramifications, work primarily with inputs defined directly in the workspace, and do not work on the image space as the input (Choset, 2005).

The objective of this work is then to try to develop a model that would reflect motor properties in an embedding derived purely from the visual space. Thus, we consider the mechanisms by which the system may observe certain similarities between perceived arm configurations to construct a local neighbourhood of visual configurations. These local neighbourhoods are presumed to lie on a low-dimensional surface in the very high-dimensional image space. In the ISOMAP algorithm being used here (Tenenbaum, Silva, & Langford, 2000), these local neighbourhoods are assumed to be linear in the dimensionality of the manifold, and they are then "composed" to construct the global relations between distant arm configurations. For the purposes of the demonstration we have used a simulated two degree of freedom arm.

Reach planning and obstacle shifts

The input consists of a large number of images showing the arm in random poses in its workspace. These images are then mapped to a lower-dimensional manifold. We shall use the term *visuo-motor map* and *low-dimensional embedding* for this image to low-dimensional map. Note that the variables in the low-dimensional map are just mathematical abstractions that emerge from the computation and are not based on

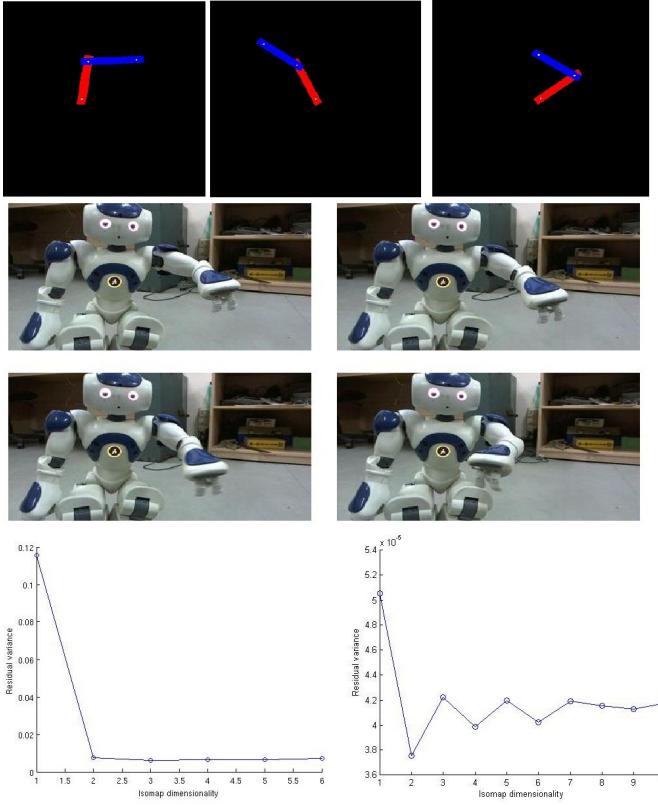


Figure 1: Image data from (a) two-degree of freedom planar arm, (b) humanoid robot nao moving one arm along a linear trajectory. The residual variance error in the lower-dimensional space after ISOMAP, versus the dimensionality of the embedding is shown below. Left: planar arm (2-DOF). Right: humanoid robot nao moving arm in straight line (1-DOF); errors are noisy, $\tilde{10}^{-5}$.

prior knowledge about any physical variable. Thus, we use no knowledge of the kinematics or geometry of the arm, except the initial image data. Despite this paucity of supervisory knowledge, we are able to identify the number of degrees of freedom (two) of the motion system, and also a set of (two) parameters that can be controlled to move the arm to different configurations within its workspace. We also show that these parameters can be easily mapped to physical variables such as joint angles.

The visuo-motor map is obtained as a graph embedded in the lower-dimensional space, each node is mapped to an image in the much higher-dimensional image space. The structure of the graph is shared between both the original image space as well as the lower-dimensional embedding space. In the following, we shall consider a) constructing such a map (with no prior knowledge about the physics of the arm), b) using it to map a path from a source to a target in the absence of obstacles, and c) map obstacles that may impinge on the path, and plan trajectories that may avoid it. A further aspect is that with increasingly dense samples in the image space,

the models become increasingly accurate.

We demonstrate the dimensionality reduction process - part (a) - on three domains - images of a simulated two-link planar arm moving to various positions in its workspace, a humanoid robot moving its arm along a straight line, and two coordinated planar arms moving a box. A two-dimensional interpolation is shown to work well for the first situation (two degree of freedom in the arm). For the other two situations, we find that one dimension is enough. For motion planning (part b), and obstacle avoidance (c), we restrict ourselves to only the first simulated arm - the two degree of freedom simulation of the upper and lower limbs.

For motion planning, both initial and desired configurations, as well as obstacles, if any, are known only in the visual space. The initial and target images are mapped to the manifold embedding using the out-of-sample mapping - i.e. they are interpolated based on the “nearest” images and the same interpolation is applied to the embedded points. Now, a mapping to the goal can be discovered in terms of edges in the graph, which actually constitute a roadmap, but in the visuomotor space and not in the usual configuration space of robot motion planning. The problem of having the visuomotor map represent obstacles is quite challenging, since some obstacles (such as the own body) may be permanent, but in most situations other constraints would arise in the workspace. As the obstacle positions shift with respect to the arm, must it re-learn the visuo-motor embedding every time? This is indeed what happens to most robot-motion planning algorithms, and handling dynamic obstacles remains a considerable challenge in robotics. In our case, we assume that our map is a base map of the maximal free space, constructed in the absence of all movable obstacles. This results in an embedding from which nodes can be deleted if a obstacle is introduced to a region of the image space which was otherwise occupied by an arm in one of the data points. Such situations are handled by marking as *blocked* the non-free nodes of the graph in the lower-dimensional space. To start with therefore, we consider full rotations for each of the two joints of our simple 2-DOF arm, and images are randomly sampled across this entire 360×360 degree rotation range.

As obstacles are introduced into the workspace, the system marks those nodes of the graph that visually overlap with the obstacle as *blocked*, and motions are restricted to the remaining “free” parts. Of course, this method assumes that if two nodes are in free space, the path between them is also free, which is not true for the non-linear mapping space. However, considering how densely sampled the image space of a limb in daily use is for an organism, we may argue that in real situations, the visuomotor map will have a dense mapping so that this assumption is far more likely to hold.

An additional characteristic of the learning process reflects increasing accuracy in the visuo-motor map. As the system encounters more and more motions in the visual space, the graph becomes more dense, and the accuracy of plans improves. This may also contribute to the observations of in-

creasing fluency in infant motions, though much of it is also due to development in the musculatory system.

Dimensionality Reduction

We consider a computational system exposed to a large number of images reflecting different motor configurations of its arms. At the same time, it has muscle sensations of how the arm motion is executed. In this work, we focus primarily on the visual input space. We first observe that the data there, captured here in terms of a set of large images, constitute an extremely high dimensional image space. Every pixel may be considered to be independent, so that a 800×800 image (figure. 1(a)), is a point in a 64000 dimensional space. That is each image is one point out of N^{64000} , where N is the number of values a pixel can take. Clearly, the system needs to reduce the complexity of this data drastically. In the biological system, this is achieved by a combination of neural data compaction processes starting in the retina, as well as a number of learned responses that eventually result in responses in the pre-motor areas, primarily in the posterior-parietal cortex (PPC) (Batista et al., 1999). Here we consider a computational model for constructing a lower-dimensional manifold from the image data.

Many approaches have been tried for dimensionality reduction. One class of approaches assume that the underlying manifold from which the observations are drawn is linear. Linear dimensionality reduction methods include linear methods such as Principal Component Analysis or Multi-dimensional Scaling (Bishop, 2006). On the other hand, non-linear dimensionality reduction assumes that the data is drawn from a manifold - a surface that is mappable everywhere to a disk in a Euclidean space of much lower-dimension. Such NLDR algorithms include Isomap, Locally-Linear Embedding (LLE), Local Tangent Space Alignment, etc. (Lee & Verleysen, 2007). In this work, we have used the ISOMAP algorithm, which attempts to preserve geodesic distances in the lower-dimensional mapping.

Isomap on the image space

To see how the ISOMAP algorithm is applied to our image space, we observe that a continuous motion will result in a sequence of images that lie along a one-dimensional curve in the image space. This is because images from successive instants in time are likely to be very similar, and extending this similarity locally would result in a 1-manifold. This applies to complex real images (figure. 1b) Now, let us consider a two-degree of freedom arm. For the two-degree of freedom, it turns out that all such one-dimensional trajectories in the image space are found to lie on a surface, which is everywhere mappable to a disk in the Euclidean plane (R^2). Thus, all these images lie on a curve 2-dimensional manifold in the image space (figure. 1). Distances between configurations can be thought of as distances between the shortest paths (geodesics) that lie on this manifold. The Isomap essentially solves an optimization problem where the lower-dimensional points are to

be chosen so as to minimize the error in these geodesic distances. This is the *residual error* which is reported in the graphs in figure 1c. If there is a sharp drop in the error at some value of the dimension used, then one assumes that this dimension is a good estimate for the dimensionality of the underlying manifold. In some cases (e.g. for the Nao robot images), all the errors are less than 10^{-5} - this implies that the error is already very low at dimension one, so $d=1$ is a good estimate for its dimensionality.

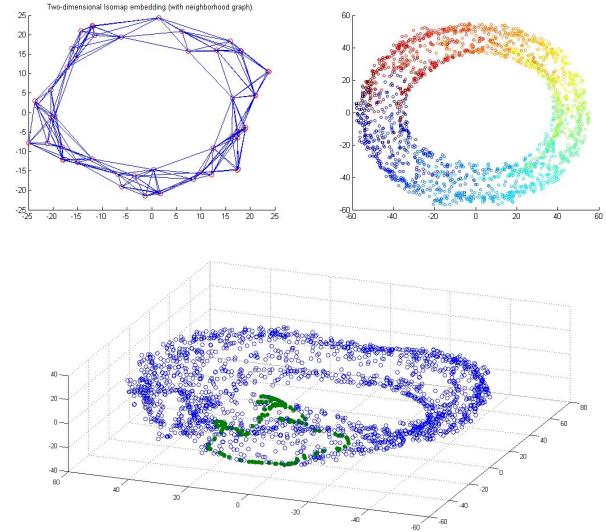


Figure 2: Two-dimensional embeddings generated by ISOMAP. Top Left: map from 50 robot images. Top Right: map from 2000 images showing variation of θ_1 along the manifold. Bottom: map in 3d showing a cross-section with θ_1 kept constant and θ_2 varying. Note that the topology of the embedding actually reflects the topology of the θ_1, θ_2 space - in that the θ_1 variation occurs along the ring (top right figure), while the θ_2 variation is approximately radial (bottom). Note that the 50-case also captures the cyclic nature of θ_1 , though the resolution is poor. This suggests the torus topology of the θ_1, θ_2 space.

Out of sample points

A significant difficulty with non-linear dimensionality approaches is that these models are not very good for mapping novel (unseen) data points. In order to overcome this very significant difficulty, some approaches such as the Locality Preserving Projections (LPP) (Lee & Verleysen, 2007). Here a linear mapping is constructed based on a quadratic function that attempts to preserve some aspects of the non-linear data. However, the inverse mapping (from lower dimension to the original space) in such systems is poor, and we have restricted ourselves to true non-linear models.

In order to solve the out of sample interpolation problem, we have borrowed an idea from Locally-Linear Embedding, where a local approximation can be constructed for novel

points based on a weighted linear approximation in the image space. Now, the same weights are used in the lower-dimensional space to obtain a mapping for the point in that space. Indeed, in the LLE, the fact that these weights remain the same in the lower-dimensional space is the constraint that is the basis for its optimization. The underlying assumption is that locally, the nearest neighbours around the novel point may be approximated by a linear surface in both the image space and the target space. While the Isomap algorithm is premised on a somewhat different (global geodesic) constraining, the locally linear approximation is not too far off for the isomap, especially if the sampling is dense. Thus, this method is here adopted to our isomap data; though it is only approximate, the method appears to work well in practice.

Thus, the objective of this work is to use visual input that is as general as possible, but to show that for certain functional situations, it can lead to much reduced dimensionalities. Thus, here, the very fact that it reflects different configurations of the same physical arm restricts its variation in the image space. Although we have experimented with several other dimensionality reduction techniques, we report here only the results based on ISOMAP. The approach is to first reduce the dimensions of the visual input and then use this compact representation to do path planning in the presence of obstacles. Also, we show that a simple mapping can be learnt between the low dimensional embedding generated by ISOMAP and the motor signals and thus effectively eliminating the problem mentioned above.

Visuo-Motor embedding

We conducted the following experiments on 100×100 images of a robot. For our experiments, we assumed that the robot is free to move all around - that is there are no restrictions on θ_1 and θ_2 , i.e., $0 \leq \theta_1 \leq 360$, $0 \leq \theta_2 \leq 360$. The lower dimensionality embedding generated by the Isomap, as shown in figure 3, is the visuo-motor embedding since it captures motor signals using visual data and both can be arrived at using the embedding.

Mapping Visuo-Motor Embedding to Motor Signals

In this experiment, we illustrate that the ISOMAP not only reduces the number of dimensionality optimally, but this reduction is not arbitrary and geometry of the embedding captures important insights into parameterization of the data - θ_1, θ_2 (figure 3). In order to further substantiate this point we show that there exists simple mapping from the embedding to motor signals - (θ_1, θ_2) . We used a 1-hidden layer (10 hidden units) feed forward neural network with linear output to learn the mapping with small errors. Moreover the parameters governing the data - θ_1, θ_2 , can be seen as motor signals given to manipulate the robot. Note that, once a path is found in the embedding space, such a mapping is required to realize it in real i.e., generating the actual motor signals to traverse the path. While there exists a simple mapping from the embedding space to the motor signals, it is very hard to

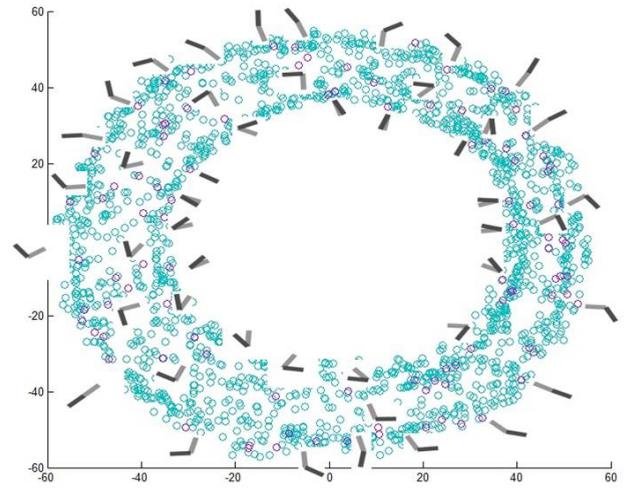


Figure 3: ISOMAP embedding (torus) with points with a sampling of images. The base angle (what we call θ_1) changes along the circumference of the torus and θ_2 along the radius of the torus.

learn a mapping directly from the sensory input (image space) to motor signals (θ space) (see figure 4)

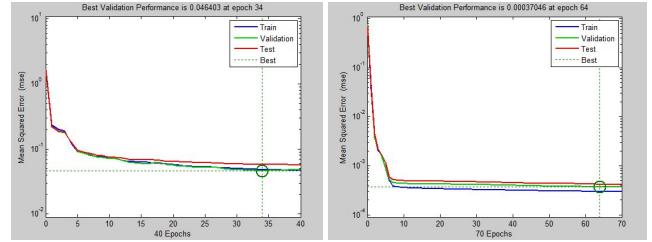


Figure 4: Comparison of the performance of neural network when learnt from high dimensional images (left) and low dimensional embedding (right) to θ_1, θ_2 for the two-link manipulator.

Mapping Obstacles to Visuo-Motor Embedding

Given an obstacle, we mark all the points which are blocked by the obstacle as colliding and others as collision free configurations (this can be checked very easily in the image space) resulting a set of collision free points in which the path can be planned. An important advantage of this method is that it avoids recomputation of the embedding and obstacle mapping for each obstacle separately. The set of collision free points can be easily updated in the event of a moving obstacle (figure 5), new obstacles appearing in the workspace, and old obstacles going away. This also preserves the topology of the embedding.

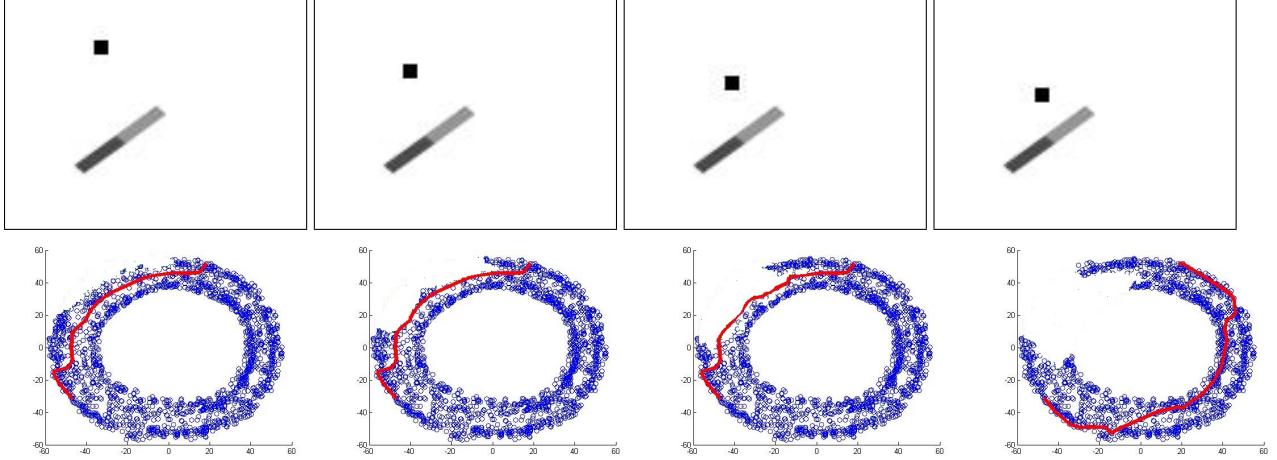


Figure 5: Robot and obstacle configurations and corresponding mapping to embedding space. The set of collision free points can be easily updated for a moving obstacle as shown. Moreover, if a previously free path becomes obstructed due to movement of an obstacle or appearance of a new obstacle, a new path can be easily be planned with the updated set of collision free points. Blue points here are the set of collision free points and green points the set of colliding points. Red color shows the points along the path from the initial configuration to final configuration

Path Planning in Visuo-Motor Embedding

Once we have the embedding points and a set of collision free points we can plan a path between any pair of start and end points, using any of the graph search algorithms. In case, the start or end point is not in the embedding, a nearby point is found in the embedding and then path planning is done between those points. One of the important observations is that a shortest path query in the graph, results in a path which avoids any redundant movements which is consistent with how a human would do if presented with same planning task. In other words, the shortest path in the embedding space corresponds to least changes to parameters - θ_1, θ_2 . (see figure 6) Moreover, as an obstacle changes its position in the workspace, the obstructed points can be removed as visually perceived and a new path can be planned avoiding the obstacle.

Learning a representation?

We have shown that this process, starting from a set of images and no other information, is able to discover a number of facts. We present these facts, which we call a *naive representation*, presenting in parentheses what a robotics text may call an analog for these.

- a two dimensional manifold obtains then best reduction in residual error. (Robotics: it has two degrees of freedom)
- the structure of the low-dimensional 2-D manifold is that of a torus, so it is cyclic in both dimensions, i.e. it is a as $S^1 \times S^1$ topology. (R: it is an $S^1 \times S^1$ topology)
- if we associate two variables with these 2 DOFs, we may have one go around the torus in the main direction, and the

other along the thickness, then these capture some aspects of the variation (R: θ_1, θ_2)

- any node in the graph associated with two variables is mapped to the image space, and any image used in the original input is mapped to a node. (R: forward and inverse kinematics)
- the space of these variables is connected with edges that denote nearest neighbours. (R: there is a roadmap)
- given any images for the start and end positions, it is possible to find a path connecting them using these edges. (R: roadmap-based path planning)
- given an obstacle, the system can determine which configurations hit the obstacle in the image space. The corresponding nodes are removed in the manifold graph. (R: the C-space map of an obstacle).
- given an obstacle and a start and goal image, a path can be found via those edges not incident on any of the obstacle nodes. (R: C-space obstacle avoidance).

The above analogies are of course very coarse. Only the topology is really preserved; none of the metric properties are guaranteed to hold; thus the path obtained may not be very short. Nonetheless, considering that no prior knowledge of any kind was used in constructing this motor map, the above naive representation is surely an impressive analogy to present models of robotics.

Note however, that the system, in using these routines, need not be “conscious” that it is using such a representation - it just has to be able to use it effectively.

A final note on the precision of the process. We note that the space need not be sampled uniformly, as was done here. In many situations, certain configurations would be occurring more frequently, and there would higher accuracies would hold in these part of the space (as in the rough diagram of the 50-torus earlier). This is also true of animals and humans, where more frequently executed actions are much more precise.

Conclusion

In this work, we have demonstrated that although visual input is extremely high-dimensional, the data lies on a nonlinear manifold that is much lower dimensional. The dimensionality of this manifold, and its structure, reveals much about the problem domain and may be called a *naive representation*. We also propose an approximate mechanism for handling out-of-sample data in the NLDR algorithm - first, by constructing the manifold for the entire workspace and then deleting points, as opposed to trying to construct a new manifold for every workspace. This (figure) also explains the fact that during the early stages of motor learning, the resolution of the map is poor, and this may also contribute to some aspects of the jerky movements demonstrated in early infancy. As the system populates the visual space with more data points, the resulting surface becomes smoother and permits fluent motions.

Is there cognitive evidence that such a method is actually implemented in the cortex? It is perhaps too early to comment on this. We can only point to a clear role for eye-centered coordinates in the reach planning process, and suggest that such a representation, involving “nearby” images from the image space, may constitute at least a plausible model for part of the computation involved in reach planning.

References

- Adolph, K., & Berger, S. (2006). Motor development. In *Handbook of child psychology*.
- Batista, A., Buneo, C., Snyder, L., & Andersen, R. (1999). Reach plans in eye-centered coordinates. *Science*, 285(5425), 257.
- Beltrán-González, C. (2005). *Toward predictive robotics: the role of vision and prediction on the development of active systems*. Unpublished doctoral dissertation.
- Bishop, C. (2006). *Pattern recognition and machine learning*. Springer.
- Caeyenberghs, K., Wilson, P., Van Roon, D., Swinnen, S., & Smits-Engelsman, B. (2009). Increasing convergence between imagined and executed movement across development: evidence for the emergence of movement representations. *Developmental science*, 12(3), 474–483.
- Choset, H. (2005). *Principles of robot motion: theory, algorithms, and implementation*. The MIT Press.
- Graziano, M. (2011). Is reaching eye-centered, body-centered, hand-centered, or a combination? *Reviews in the Neurosciences*, 12(2), 175–186.
- Hoffmann, H., Schenck, W., & Moller, R. (2005). Learning visuomotor transformations for gaze-control and grasping. *Biological Cybernetics*, 93(2), 119–130.
- Jordan, M., & Wolpert, D. (1999). Computational motor control. *The cognitive neurosciences*, 601.
- Kuperstein, M. (1991). Infant neural controller for adaptive sensory-motor coordination. *Neural Networks*, 4(2), 131–145.
- Lee, J., & Verleysen, M. (2007). *Nonlinear dimensionality reduction*. Springer.
- Meer, A. van der. (1997). Keeping the arm in the limelight: advanced visual control of arm movements in neonates. *European Journal of Paediatric Neurology*, 1(4), 103–108.
- Meer, A. Van der, Weel, F. Van der, & Lee, D. (1995). The functional significance of arm movements in neonates. *Science*, 267(5198), 693.
- Meer, v. d. W. F. R. . L. D. N. van der, A. L. H. (1996). Lifting weights in neonates: Developing visual control of reaching. *Scandinavian journal of psychology*, 37(4), 424–436.
- Mushiake, H., Tanatsugu, Y., & Tanji, J. (1997). Neuronal activity in the ventral part of premotor cortex during target-reach movement is modulated by direction of gaze. *Journal of neurophysiology*, 78(1), 567–571.
- Oztop, E., Bradley, N., & Arbib, M. (2004). Infant grasp learning: a computational model. *Experimental Brain Research*, 158(4), 480–503.
- Saxon, J., & Mukerjee, A. (1990). Learning the motion map of a robot arm with neural networks. In *Neural networks, 1990., 1990 ijcn international joint conference on* (pp. 777–782).
- Tenenbaum, J. B., Silva, V. de, & Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500), 2319.
- Thelen, E. (2000). Motor development as foundation and future of developmental psychology. *International Journal of Behavioral Development*, 24(4), 385–397.
- Thelen, E., Smith, L., Lewkowicz, D., & Lickliter, R. (1994). *A dynamic systems approach to the development of cognition and action*. MIT Press.
- Von Hofsten, C. (2004). An action perspective on motor development. *Trends in cognitive sciences*, 8(6), 266–272.