



**HAL**  
open science

# Learning of local predictable representations in partially learnable environments

Mathieu Lefort, Alexander Gepperth

► **To cite this version:**

Mathieu Lefort, Alexander Gepperth. Learning of local predictable representations in partially learnable environments. The International Joint Conference on Neural Networks (IJCNN), Jul 2015, Kilarney, Ireland. hal-01205611

**HAL Id: hal-01205611**

**<https://hal.science/hal-01205611v1>**

Submitted on 25 Sep 2015

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Learning of local predictable representations in partially learnable environments

Mathieu Lefort and Alexander Gepperth

**Abstract**—PROPRE is a generic and cortically inspired framework that provides online input/output relationship learning. The input data flow is projected on a self-organizing map that provides an internal representation of the current stimulus. From this representation, the system predicts the value of the output target. A predictability measure, based on the monitoring of the prediction quality, modulates the projection learning so that to favor learning of representations that are helpful to predict the output. In this article, we study PROPRE when the input/output relationship is only defined in a small subspace of the input space, that we define as a partially learnable environment. This problem, which is not typical of the machine learning field, is however crucial for the robotic developmental field. Indeed, robots face high dimensional sensory-motor environments where large areas of these sensory-motor spaces are not learnable since a motor action cannot have a consequence on every perception each time. We show that the use of the predictability measure in PROPRE leads to an autonomous gathering of local representations where the input data are related to the output value, thus providing good classification performance as the system will learn the input/output function only where it is defined.

## I. INTRODUCTION

According to sensory-motor theories, sensory-motor regularities that emerge from the interaction between an agent and its surrounding environment are a key point for this agent to build perceptions of the environment [16]. Moreover, they seem to play a crucial role for humans as various experiments in psychophysics show that stimuli consistent with usual environmental interactions improve learning and detection of events compared to unisensory or inconsistent stimuli (see [3], [19], [22] for example). Hence, online and progressive construction of sensory-motor representations is one of the current challenge of the developmental robotics field [10], [17], [20] that targets the conception of robots that are able to learn to interact with an unknown environment in an autonomous and lifelong open-ended manner.

As an agent catches its internal state and the one of its surrounding environment through multiple sensors, sensory-motor representations have to be learned from the fusion of these various sensor data flows. Multimodal data fusion is a very complex task that raises challenging problems such as what, when and how to fuse data [1], even if databases are well prepared as in the machine learning field. Additionally, when learning multimodal data in the developmental robotic field, one has to consider that large areas of the sensory-motor space will never be learnable [2]. One example among

many others is that it is hopeless to try to predict the olfactory consequence of an eye movement. In this article, we consider the problem of learning an input/output relationship where the input/output function is well defined, even if noisy, in some restricted areas of the input space - that we define as learnable areas - but is not existing in most of the space - that we defined as unlearnable areas. Learning in such partially learnable environments is a very challenging and crucial problem for the developmental robotic field.

To tackle this problem, we use an architecture that is cortically inspired. At a macroscopic level of description, the cortex is structured in interconnected functional areas at various levels, from low level sensory areas (V1, A1, ...) to high level areas in the frontal cortex. At a mesoscopic point of view, structure and function of these areas seem to be generic [8], [9], [15]. Especially, self-organization (i.e. close neurons of a cortical area are sensible to close stimuli) is a widely spread computational paradigm that is mainly observed in low level sensory areas [4], [21], [23].

PROPRE (that stands for PROjection-PREdiction) is a generic architecture for multimodal representations learning [5]. Each data flow is projected on a self-organizing map, providing an internal representation of each current stimulus. From each representation, the system tries to predict the ones of all other data flows. A correct prediction can only be obtained if corresponding stimuli are correlated. A predictability measure, that depends on the ability of a representation to predict the other ones, modulates the corresponding projection learning so that to favor learning of predictable multimodal stimuli at the system level. In previous works, we already show that this predictability modulation provides better multimodal representations for real data classification [14], where all data were labeled, and that these representations are able to adapt to changes in the input distribution [13]. This article studies the PROPRE architecture facing the complex problem of partially learnable environments.

In the next section we introduce the PROPRE paradigm. In section III, we present the two partially learnable environments that we use. The first one is derived from the MNIST database [12] where we artificially corrupt some digit categories. The second one is a robotic arm direct model with a restricted visual field. We show that PROPRE provides local representations gathered in the learnable areas of the environments. This provides a simple way to discriminate between learnable and unlearnable parts of the environment and improves the performance of our model that learns the input/output relationship only where it exists. We discuss possible implications and extensions of our work in section IV.

---

Mathieu Lefort and Alexander Gepperth are with the UIIS division, ENSTA ParisTech, 828 boulevard des Maréchaux, 91762 Palaiseau, France. Mathieu Lefort and Alexander Gepperth are members of Inria FLOWERS, Inria Bordeaux Sud-Ouest, 200 avenue de la vieille tour, 33405 Talence, France (email: {mathieu.lefort, alexander.gepperth}@ensta-paristech.fr).

## II. PROPPE

### A. Main paradigms

PROPPE is a generic neural paradigm for online learning of representations from multiple data flows. Its processing is based on the coupling of projection and prediction and its architecture is structured in three modules (see figure 1). The projection module uses the self-organizing map paradigm (SOM) to autonomously obtain a dedicated non linear topological projection of each unimodal input space. It thus provides a spatial representation of the current stimulus of each data flow. In the prediction module, each representation is used to linearly predict representations of all other data flows. Each prediction can only be accurate if the corresponding modal stimuli - through their low level internal representations - are correlated. A predictability module quantifies the quality of each prediction that represents the interestingness of the learned unimodal representation as it indirectly reflects the correlation between the multimodal stimuli. This predictability measure is used to modulate the corresponding projection learning so that to favor the representations of stimuli correlated across modalities thus providing an hybrid framework combining generative and discriminative learning. For more details about the PROPPE paradigm, please refer to [5].

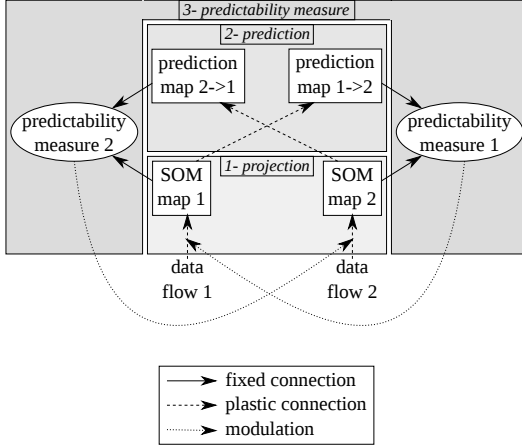


Fig. 1. PROPPE architecture is structured in three interacting modules. A projection module learns a low dimensional representation of each modal stimulus. A prediction module provides, from each unimodal representation, a prediction of all other representations. A predictability measure quantifies the quality of each prediction and modulates the corresponding projection learning.

In the multimodal architecture, the target data flow is the learned representations of other data flows. However, any target value can be used to learn any arbitrary input/output function (see figure 2) as we already did in [13] for pedestrian pose orientation classification. In this article, we study this data flow processing when learning a partially learnable environment, i.e. that the output value does not depend on the input value for large areas of the input space. In this context, the predictability measure plays a crucial role for identifying learnable areas from the environment.

From a computational point of view, the reception of a stimulus in the model leads to one transmission and one learning step so that the model provides online learning (i.e.

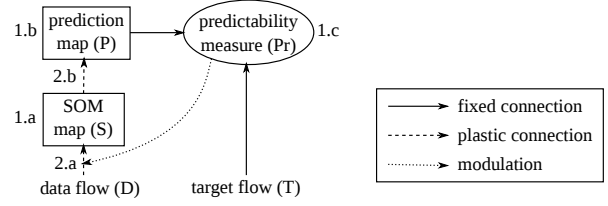


Fig. 2. One data flow processing in PROPPE. The system learns internal representations of the data flow that help the learning of the input ( $D$ )/output ( $T$ ) relationship.

the stimulus is represented and learned at the same time). Technically speaking, the transmission stage consists on the feed-forward evaluation of each module activity (equations 1.a-b-c in figure 2). Then, the learning stage updates the weights of the plastic connections linking the modules (equations 2.a-b). The equations are detailed in the three next sections describing respectively each of the three modules of PROPPE.

### B. Projection

The projection step consists on a classical Kohonen self-organizing map (SOM) [11] except for two points (see equation 2.a). Firstly, the neighborhood radius and the learning rate are constant over time, instead of decreasing, in order to have an online adaptation of the prototypes. Secondly, the prototype learning is modulated by the predictability measure to favor learning of representations that can predict the target value, which is the main feature of PROPPE.

In practice,  $S$  is a discrete bi-dimensional square grid of neurons that receives the input data flow  $D$  (see figure 2). The prototype at time  $t$  of a unit  $x$  in the map is denoted  $w_{SD}(x, t)$ . These prototypes are updated with the following equation:

$$\Delta w_{SD}(x, t) = \eta Pr(t) e^{-\frac{\|x - x^*(t)\|_2^2}{2\sigma^2}} (D(t) - w_{SD}(x, t)) \quad (2.a)$$

with  $\eta$  the constant learning rate,  $\sigma$  the constant variance of the Gaussian neighborhood radius,  $\|\cdot\|_2$  an euclidean distance in the map,  $Pr(t)$  the predictability measure (see section II-D) and  $x^*(t)$  the best matching unit defined as the unit whose prototype is the closest to the current input  $D(t)$ , i.e.  $x^*(t) = \arg \min_x \|w_{SD}(x, t) - D(t)\|_2$ , with  $\|\cdot\|_2$  an euclidean distance on the input space.

The activity of any unit  $x$  of the map is defined as

$$S(x, t) = \begin{cases} 1 & \text{if } x = x^*(t) \\ 0 & \text{otherwise} \end{cases} \quad (1.a)$$

This simple map activation function provides a prediction that only depends on the best matching unit (see section II-C), so that there is no interference between units in the predictability measure computation (see section II-D). This allows to more precisely observe the influence of the predictability measure on the unit prototypes learning in partially learnable environments, which is the aim of this article. However, in order to obtain a better classification performance, other activation functions can be used [7].

### C. Prediction

The projection activity  $\mathbf{S}(t)$  is used to compute a prediction  $\mathbf{P}(t)$  of the current value of the target data flow  $\mathbf{T}(t)$  at each time  $t$ . The activity of a unit  $x$  in  $P$  is computed as a weighted sum of the  $S$  activity:

$$P(x, t) = \sum_y w_{PS}(x, y, t) S(y, t) \quad (1.b)$$

with  $w_{PS}(x, y, t)$  the weight between the unit  $y$  in  $S$  to the unit  $x$  in  $P$ .

The weights of the connection between  $S$  and  $P$  are learned with a classical stochastic gradient descent implementation of a linear regression. This algorithm minimizes the mean square error between the prediction  $\mathbf{P}(t)$  and the current target value  $\mathbf{T}(t)$ . Thus, the weights are updated with the following equation:

$$\Delta w_{PS}(x, y, t) = \eta' S(y, t) (T(x, t) - P(x, t)) \quad (2.b)$$

with  $\eta'$  the constant learning rate.

### D. Predictability measure

The predictability measure modulates the projection learning depending on the quality  $Q(t)$  of the prediction  $\mathbf{P}(t)$  with respect to the true target value  $\mathbf{T}(t)$ . It is defined as

$$Pr(t) = \begin{cases} Q(t) - \theta(x^*(t), t) & \text{if } Q(t) > \theta(x^*(t), t) \\ 0 & \text{otherwise} \end{cases} \quad (1.c)$$

$$Q(t) = \frac{P(z^*, t)}{\sum_z P(z, t)} \quad \text{with } z^* = \arg \max_z T(z, t)$$

$$\theta(x, t) = \begin{cases} (1 - \tau)\theta(x, t - 1) + \tau Q(t) & \text{if } x = x^*(t) \\ \theta(x, t - 1) & \text{otherwise} \end{cases}$$

In our experiments, the target value represents the category of the input, if any, so that the index of the highest target value corresponds to the input category, when it is defined (see section III). Thus, the quality of the prediction  $Q(t)$  represents the percentage of prediction of the right category. Please note that this measure can also be consistent with a regression learning task if the target value is encoded with a spatial coding, i.e. that the real value is represented by the spatial position of an activity in a map. This is precisely the case when PROPRES is used for multimodal learning [14], as the target value is the representation of other data flows (see section II-A), which is the projection activity, here consisting on the best matching unit activated (see section II-B).

$\theta(x, t)$  is a sliding threshold that is computed as the average prediction quality of a unit when it is the best matching unit (i.e. when  $x = x^*(t)$ ). Thus, it represents the average prediction quality in a local area of the input space, corresponding to the Voronoi cell of the unit. This sliding threshold is computed as an online average with a temporal horizon  $\tau$ .

The predictability measure is equal to the difference between the current prediction quality and the average one of

the current best matching unit<sup>1</sup>. Thus, the current stimulus is only learned if it provides a prediction locally more accurate than the average and the learning strength depends directly on this accuracy difference to the local average.

## III. EXPERIMENTS AND RESULTS

For all our experiments, in order to quantify the influence of the predictability modulation on the system learning, we compare the PROPRES architecture described in section II with the same one without the modulation influence (corresponding to  $\forall t, Pr(t) = 1$  in equation 2.a) that we denote SOM+LR in the following of the article. Unless specified otherwise, in section III-A (respectively III-B), all experiments are done with a  $10 \times 10$  (respectively  $20 \times 20$ ) SOM map and training occurs during 20 millions (respectively 100 millions) of time steps, where an input/output couple of values is randomly picked in the train dataset at each time step. Each initial connection weight is uniformly randomly chosen in  $[0, 1[$ ,  $\tau$ ,  $\eta$  and  $\eta'$  are set to 0.01.

### A. MNIST database with corrupted digit categories

1) *Protocol*: We define several datasets derived from the classical machine learning MNIST database [12]. The handwritten digits pictures are used as the input data flow. The target value, which corresponds to the digit category, is defined for every picture in the MNIST database. In order to get partially learnable environments, we corrupt this target value for some digit categories. In practice, the target value is computed with the following steps (see figure 3):

- first, we corrupt some digits (1, 2, 4, 5 and 7 in our experiments)<sup>2</sup> by not providing the category to the system when a corresponding picture is presented as input
- second, distractors, i.e. other random categories, are added to the category vector (the number of distractors can be different for uncorrupted and corrupted digit categories)
- third, uniform noise is added to the category vector (the noise amplitude can be different for uncorrupted and corrupted digit categories)

So around half of the input space is unlearnable as the output does not depend on the input for 5 digit categories.

For our experiments we define 4 databases with various distractor numbers and noise amplitude, each time having around 90% of training data and 10% of test data (only containing uncorrupted digit categories)<sup>3</sup>:

<sup>1</sup>This is a slight adaptation of our previous algorithm, where predictability value was binary (learning on or off) [14], that allows more precise modulation.

<sup>2</sup>We initially choose 0, 3, 6, 8 and 9 as uncorrupted digit categories as the digits are visually similar so that having more prototypes for representing these digits may improve more clearly the classification performance. However, tests on other configurations show that the results are qualitatively independent of this ad hoc choice.

<sup>3</sup>In practice, we split the MNIST database in 80% of training data and 20% of test data. However, as we only test our model on uncorrupted digit categories, we only keep around half of the initial testing dataset.

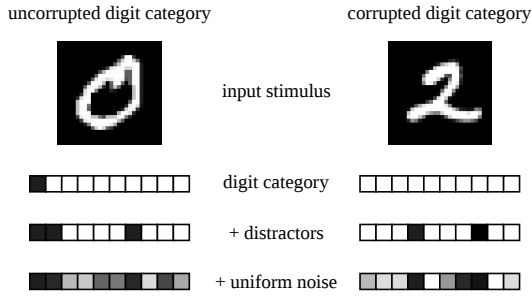


Fig. 3. Input data are the handwritten digits of the MNIST database (top). For uncorrupted digit categories - 0, 3, 6, 8 and 9 in our experiments - (left column), the digit category is represented as in the MNIST database by a 1 at the right position in the category vector, but is missing for the other corrupted digit categories (right column). Some distractors, i.e. other values at 1, are added at random positions in the vector. Then, some uniform random noise with a fixed amplitude is added to all values of the vector, providing the output target value of the system (bottom).

- setup 1: 1 distractor is added to corrupted digit categories
- setup 2: random noise with amplitude 1 is added to corrupted digit categories
- setup 3: random noise with amplitude 1 is added to all digit categories
- setup 4: 1 (respectively 2) distractor(s) is (are) added to uncorrupted (respectively corrupted) digit categories and random noise with amplitude 0.1 is added to all digit categories

Thus, setups 1 and 2 correspond to various kind of perturbations when the input/output relationship is not defined, whereas setups 3 and 4 also include perturbations of the input/output function where defined. Moreover, we define a fifth database as a baseline, where training data only contain uncorrupted digit categories without noise and distractors, denoted learnable environment in the following.

2) *Learned representations:* In figure 4, we present one example of the learned self-organizations by the projection map in the PROPRES and the SOM+LR architectures, that is representative of all four partially learnable setups. Without modulation, the projection map represents all the input space data, i.e. all 10 digits, in a self-organized way, which is the expected behavior of a Kohonen SOM. However, we can clearly observe that, with the predictability modulation, the SOM only represents digits from uncorrupted categories. Thus, thanks to the predictability measure modulation of the projection learning, the system autonomously learned to map the input space area where an input/output function is defined.

3) *Classification performances:* In the previous section, we show that the modulation introduced in PROPRES leads to representations that are gathered in the learnable area of the input space. As these representations are used as input of the prediction module, having a better mapping of the learnable area of the input space leads to a better classification performance of the system, which is shown in figure 5. Indeed, PROPRES is more efficient than SOM+LR in all setups, including the dataset where the input/output

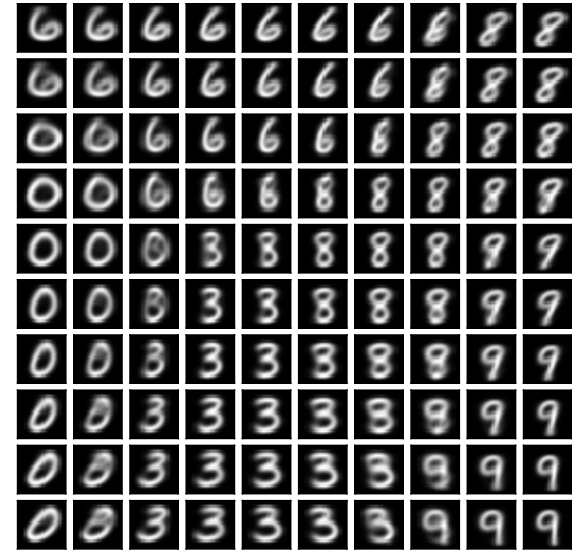
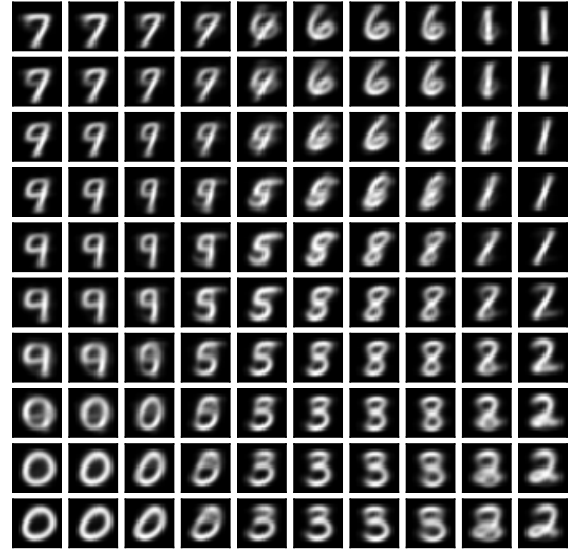


Fig. 4. Top (respectively bottom): prototypes learned by the projection map when no modulation (respectively predictability modulation) is applied. Each square represents the prototype of the unit at the corresponding position in the  $10 \times 10$  SOM.

function is defined everywhere, confirming our previous results on other benchmarks [13]. Moreover, the PROPRES performance on the partially learnable environments are close or even slightly better than the one of SOM+LR on the learnable environment. Thus, on this protocol, additionally to autonomously avoid unlearnable part of the environment, the predictability modulation is also able to improve the representation of the learnable area of the environment to get a better performance.

#### B. Motor direct model with restricted visual field

1) *Protocol:* We also tested our PROPRES architecture with some simulated robotic data, which is the targeted field of application of our model. We define several datasets corresponding to a motor direct model of a robotic planar arm with a restricted visual field (see figure 6). Thus, the

	PROPRE	SOM+LR
setup 1	8.56%	12%
setup 2	8.46%	13.9%
setup 3	9.63%	14.8%
setup 4	11.3%	14.2%
learnable environment	6.4%	9.2%

Fig. 5. Final classification errors on the different setups (refer to section III-A1 for details) depending on the use of the predictability modulation.

robot is only able to see its hand in a limited area of its reachable space. The visual input of the robot is a simple  $5 \times 10$  pixel matrix.

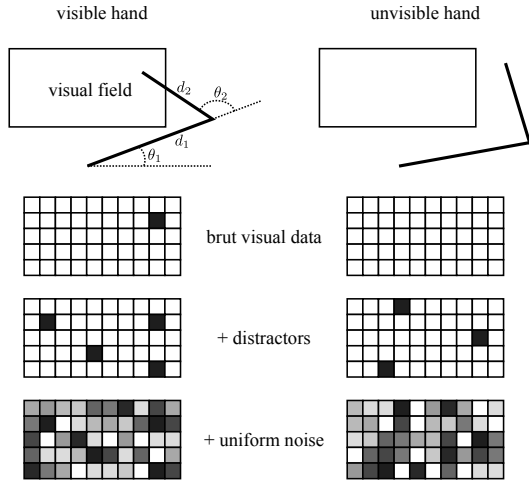


Fig. 6. A robot moves its arm ( $d_1 = 3$  and  $d_2 = 2.5$ ) in a plan depending on an input joint motor command (top). It can only see its hand in a restricted visual field. If so (left), the corresponding pixel of the visual data is set to 1, otherwise no activity is provided (right). Some distractors and some uniform random noise with a fixed amplitude are added to the visual target data of the system (bottom).

Datasets are generated as following:

- an input motor command corresponding to  $(\theta_1, \theta_2)$  is randomly chosen in  $[0, \pi[ \times [0, \pi[$  with an uniform distribution
- if the hand is visible, a 1 is put at the corresponding position in the  $5 \times 10$  pixel matrix
- some distractors, i.e. 1 values, are put at random position in the pixel matrix (the number of distractors can be different if the hand is visible or not)
- the visual target value is obtained by adding some random uniform noise with a fixed amplitude to the pixel matrix (noise amplitude can be different if the hand is visible or not)

Finally, the visual data only depends on the input motor command in around 20% of the reachable space, where the hand is visible. By the way, some part of the visual field are not reachable.

Similarly to what we did with the MNIST database, we generate 4 different datasets with various distractor number and noise amplitude, each time having 90% of training

data and 10% of testing data (only containing motor inputs providing a visible hand position):

- setup 1: 1 distractor is added when the hand is not visible
- setup 2: random noise with amplitude 1 is added when the hand is not visible
- setup 3: random noise with amplitude 1 is added whenever the hand is visible or not
- setup 4: 1 (respectively 2) distractors are added when the hand is visible (respectively not visible) and random noise with amplitude 0.1 is added whenever the hand is visible or not

Thus, setups 1 and 2 correspond to various kind of perturbations when the input/output relationship is not defined, whereas setups 3 and 4 also include perturbation of the input/output function where defined. Moreover, we define a fifth database as a baseline, where training data contain only motor commands leading to a visible hand, denoted learnable environment in the following.

2) *Learned representations*: We can clearly see in figure 7 the same effect that we observed with the MNIST database. Indeed, without modulation of the projection learning, the prototypes represent all the input space in a generative way whereas, with the modulation by the predictability measure, prototypes are gathered mainly in the learnable area of the input space. It has to be noted that, with all initial values between 0 and 1, prototypes are initialized far away from the learnable area of the input space.

Another way to study this gathering of prototypes in the learnable area of the input space is to use the distance to the best matching unit to build a binary classifier between learnable and unlearnable areas of the environment. Thus, once the model has converged, we classify the input data of a testing dataset (here including also data from the unlearnable area) as coming from the unlearnable area (respectively learnable area) if the distance between the input and the best matching unit is above (respectively below) some threshold. The corresponding ROC curves of these binary classifiers when using prototypes learned by SOM+LR and PROPRE are represented in figure 8.

We can clearly see that the classifier based on the prototypes of PROPRE is largely better than the one built with the prototypes learned without the predictability modulation. This confirms that PROPRE can autonomously learn local representations in the learnable area of the environment.

3) *Classification performances*: We also trained a multi-layer perceptron (MLP) on all setups to have a baseline of performances obtained by a standard machine learning discriminative algorithm. The hidden layer of the MLP was set to 400 units so that the internal representation in all models have the same size. Performances obtained by the various models on the different setups are shown on figure 9. A prediction error is counted when the maximum of the prediction does not correspond to the real position of the hand in the visual field.

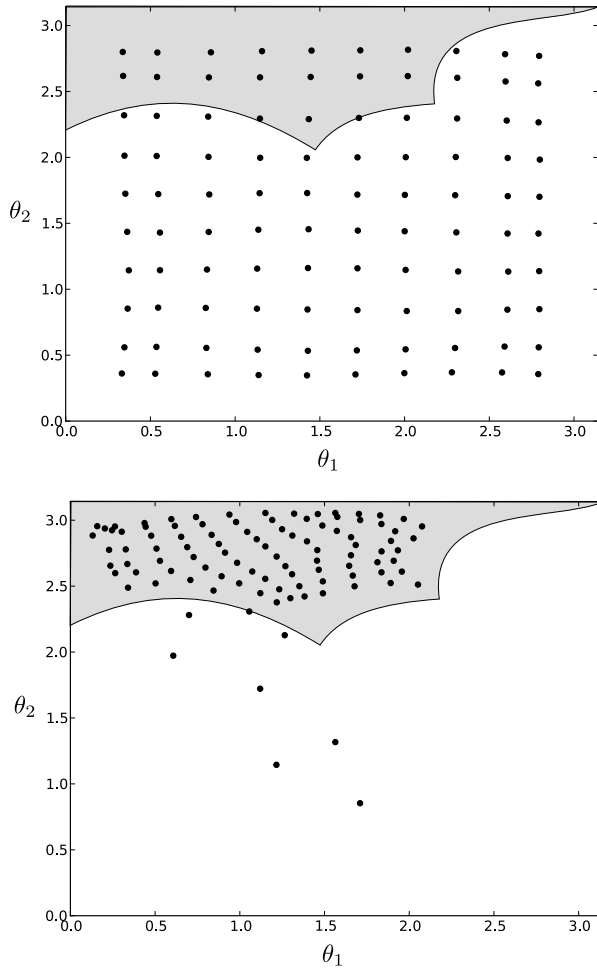


Fig. 7. Top (respectively bottom): examples of prototypes learned by a  $10 \times 10$  self-organizing map, represented in the motor input space, when no modulation (respectively predictability modulation) is applied (same qualitative results are obtained on all four partially learnable setups). The gray area represents the motor space leading to a visible hand, i.e. the learnable area of the input space.

When trained only on the learnable area of the input space, all models (SOM+LR, PROPPE and MLP) achieve logically their best classification performance. In this case, the best performance is obtained by the MLP, as expected, as it is a discriminative learning algorithm. By the way, PROPPE performance is, once again, better than the one of SOM+LR. The more interesting results concern the 4 setups where a large area of the input space is unlearnable, that does not fit with classical machine learning framework. In these cases, the SOM+LR performances decrease dramatically, as the SOM represents the entire input space (see previous section). Moreover, the MLP performances also decrease and are worst than the one of PROPPE. By the way, Hartono and colleagues proposed a model whose equations are close to the ones of PROPPE but are derived from a stochastic gradient descent so that the model is a discriminative learning algorithm [6]. Preliminary results seem to indicate that this model has the same qualitative behavior as a MLP, i.e. that its performance is deeply impacted by unlearnable areas. This illustrates the main interesting monitoring property of the PROPPE predictability module to detect and gather

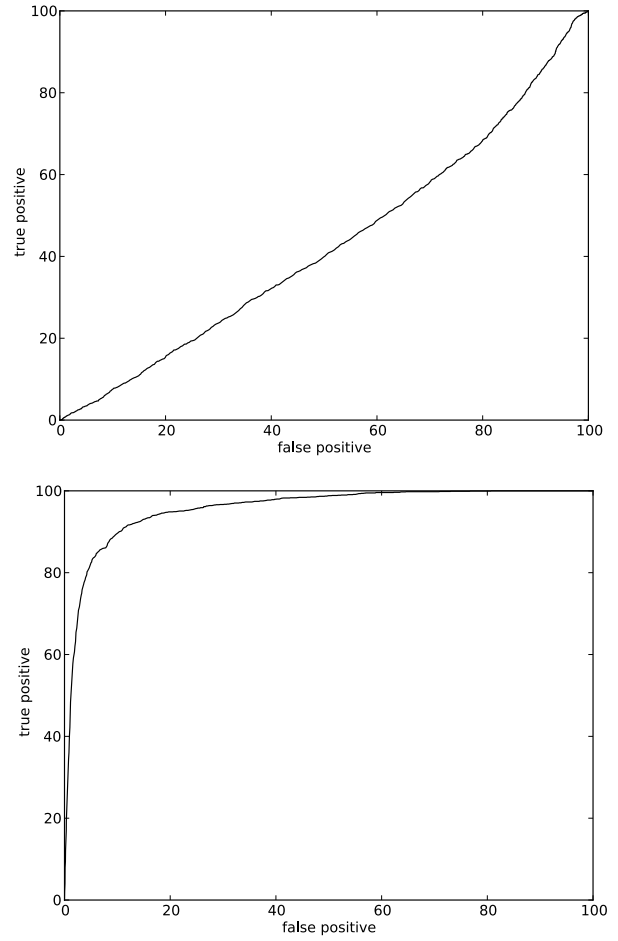


Fig. 8. Top (respectively bottom): ROC curves of a classifier that classifies data as coming from the unlearnable or learnable area of the input space depending on the distance to the best matching unit of the SOM learned with SOM+LR (respectively PROPPE).

representations in the learnable area of the input space. Thus, PROPPE only tries to regress the input/output function in the learnable area and is quite insensible to the existence of unlearnable areas contrary to classical discriminative learning approaches.

4) *Robustness to instabilities in the learnable area:* In all previously presented experiments, the input/output function, even if noisy, was always present in the learnable area of the input space and never defined elsewhere. This is precisely what distinguishes learnable from unlearnable areas. In this section, we study the robustness of PROPPE learning to some missing target values in the learnable area to be more representative of real data. For that purpose, we take the setup 2 (see section III-B1) and, with some probability, suppress the visual location of the hand in the pixel matrix when the hand should be visible. This may correspond to the occlusion of the hand behind an object for example. Performances obtained with PROPPE and SOM+LR architectures on this setup are presented in figure 10.

We can see that, for both models, the performance decreases with the increase of the percentage of input/output relationship instability in the learnable area as the prediction step has more difficulties to get a relevant statistic in the

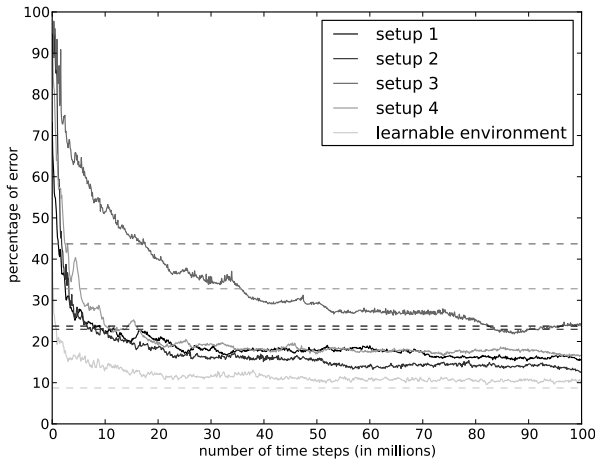
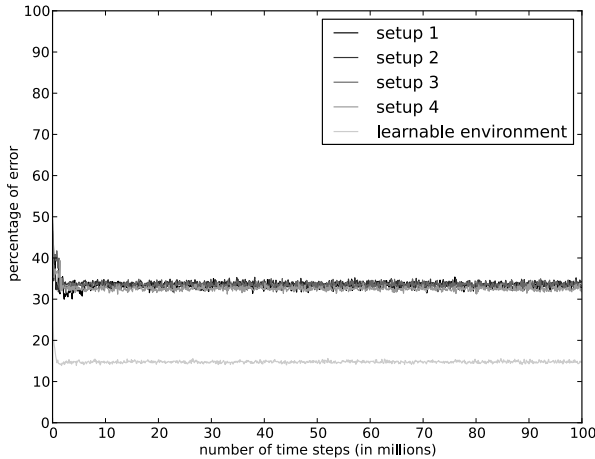


Fig. 9. Top (respectively bottom): temporal evolution of classification errors on the different setups (refer to section III-B1 for details) when no modulation (respectively predictability modulation) is used. On the bottom figure, dashed lines represent the final performance obtained by a multi-layer perceptron, the color indicates the setup on which the MLP was trained.

data. Interestingly, PROPRES performance is only gradually degraded with the instability percentage and always achieves a better performance than SOM+LR in all cases. By the way, even with up to 30% of instability, PROPRES performance remains better than the one of SOM+LR on the data without any instability.

#### IV. DISCUSSION AND PERSPECTIVES

PROPRES is a generic and modular paradigm whose architecture and processing are inspired by the ones of the cortex. It provides an input/output relationship online learning based on the combination of projection and prediction, that is well suited for multimodal learning. The input data flow is projected on a learned self-organizing map that provides a low dimensional representation of the current input. This internal representation is used to predict the value of the output target. The main originality of PROPRES is the use of a predictability measure, that depends on the monitoring of the prediction quality, to modulate the projection learning. This predictability modulation leads to learning of representations that are locally useful for the global output target prediction.

PROPRES was already successfully applied to unimodal

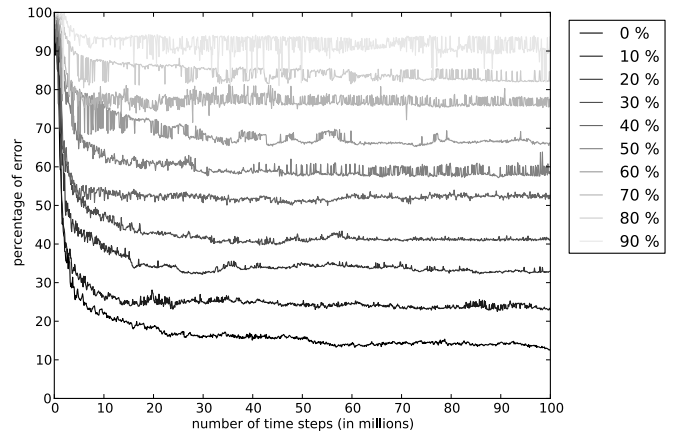
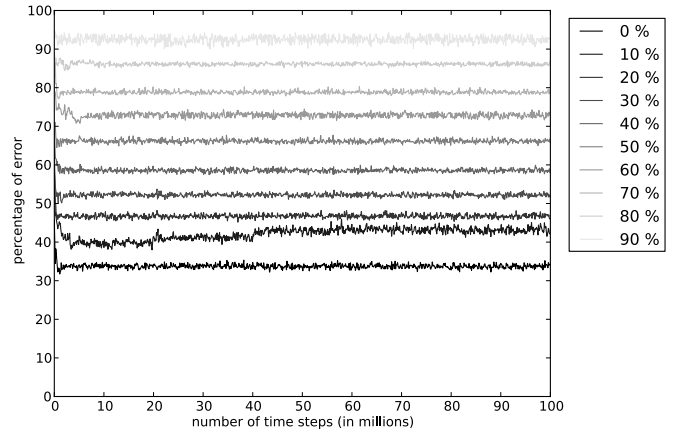


Fig. 10. Top (respectively bottom): temporal evolution of the performance of SOM+LR (respectively PROPRES) when learning on setup 2 depending on the percentage of input data in the learnable area of the environment whose target value is missing (refer to the text for more details).

and multimodal learning in real data classification tasks [13], [14], where all data are labeled. In this article, we study the PROPRES paradigm when learning environments that include a large unlearnable area in the input space, i.e. where the output value does not depend on the input. This kind of problem is not typical of the machine learning field, where benchmarks assume an hidden and noisy input/output function everywhere. However, this very challenging problem is crucial for the robotic developmental field. Indeed, robots face high dimensional sensory-motor environments, where large part of these environments cannot be predicted at any time of the development and is thus hopeless to try to learn [2].

We construct multiple benchmarks of partially learnable environments with various random uniform noise and distractors in the output target value. Firstly, we use the MNIST database where we artificially suppress the category for half of the digits. Secondly, we simulate a robotic arm in a plan, whose hand can be seen in a restricted visual field only covering around 20% of the reachable space.

The modulation mechanism introduced in PROPRES leads to a gathering of the local representations in the learnable areas of the input space. So, as PROPRES regresses the output



value from these internal representations, it only learns the input/output relationship where it is defined. Thus, when some areas of the input space are unlearnable, PROPRES performances are better than the ones achieved by the same architecture without predictability modulation and are also better than the ones of a multi-layer perceptron, a standard discriminative learning algorithm, in the robotic setup. On the contrary, the MLP obtains the best performance when the input/output relationship is defined on all the input space. This validates the important role of the predictability module to be able to learn in partially learnable environments by detecting learnable areas from unlearnable ones. This property may also be used to build a classifier that discriminates inputs from the learnable areas in the input space based on the distance between the input and the best matching unit as we illustrate it on the direct arm model learning. By the way, PROPRES performance is only slightly degraded by adding noise to the input/output function where it exists and is gradually degraded when the input/output function is missing for some increasing percentage of inputs of the learnable area, which are both interesting robustness properties.

The successful application of PROPRES to learning in partially learnable environments, lies on the monitoring provided by the predictability modulation. It is based on the prediction quality measure that is defined as the percentage of prediction of the right category. As mentioned in section II-D, it can also be applied to a regression task as we already did it in [14] for multimodal learning. Thus, a direct perspective of our work is to study PROPRES facing real multimodal data with a partial relationship between the modalities, which is particularly relevant as each sensor usually has its own characteristics. Moreover, preliminary results indicate that this predictability measure can be extended to a multiple categories representation, with some constraints, even if more work is needed to conclude on this aspect.

The discrimination between learnable and unlearnable areas of the environment, that is autonomously learned by PROPRES, can be very interesting from a developmental robotics perspective. Even if we show in this article that PROPRES can deal with partially learnable environments, exploration of highly dimensional sensory-motor environments cannot be extensive within a live time. Thus, active exploration mechanisms have to be used for guiding exploration towards interesting sensory-motor areas, that are defined as the one leading to competence progress in the artificial intrinsic motivation framework [18] for example. It may be interesting to study the coupling of PROPRES with such intrinsic motivation framework as the autonomous detection of learnable areas of the input space must provide a great help for active exploration.

#### ACKNOWLEDGMENT

The authors want to thank Thomas Hecht for his useful remarks and reviews.

#### REFERENCES

[1] P. K. Atrey, M. A. Hossain, A. El Saddik, and M. S. Kankanhalli, "Multimodal fusion for multimedia analysis: a survey," *Multimedia systems*, vol. 16, no. 6, pp. 345–379, 2010.

[2] A. Baranès and P.-Y. Oudeyer, "R-iac: Robust intrinsically motivated exploration and active learning," *Autonomous Mental Development, IEEE Transactions on*, vol. 1, no. 3, pp. 155–169, 2009.

[3] I. Bernstein, M. Clark, and B. Edelstein, "Effects of an auditory signal on visual reaction time," *Journal of Experimental Psychology*, vol. 80, no. 3p1, p. 567, 1969.

[4] W. Bosking, Y. Zhang, B. Schofield, and D. Fitzpatrick, "Orientation selectivity and the arrangement of horizontal connections in tree shrew striate cortex," *The Journal of Neuroscience*, vol. 17, no. 6, p. 2112, 1997.

[5] A. Gepperth, "Efficient online bootstrapping of sensory representations," *Neural Networks*, 2012.

[6] P. Hartono and T. Trappenberg, "Classificability-regulated self-organizing map using restricted rbf," in *International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2013, pp. 1–5.

[7] T. Hecht, M. Lefort, and A. Gepperth, "Using self-organizing maps for regression: the importance of the output function," in *European Symposium on Artificial Neural Networks (ESANN)*, 2015.

[8] K. Holthoff, E. Sagnak, and O. Witte, "Functional mapping of cortical areas with optical imaging," *NeuroImage*, vol. 37, no. 2, pp. 440–448, 2007.

[9] E. Kandel, J. Schwartz, T. Jessell, S. Siegelbaum, and A. Hudspeth, *Principles of neural science*. Elsevier New York, 1991, vol. 3.

[10] S. Kirstein, H. Wersing, and E. Körner, "Towards autonomous bootstrapping for life-long learning categorization tasks," in *International Joint Conference on Neural Networks*. IEEE, 2010, pp. 1–8.

[11] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological cybernetics*, vol. 43, no. 1, pp. 59–69, 1982.

[12] Y. LeCun and C. Cortes, "The mnist database of handwritten digits," 1998.

[13] M. Lefort and A. Gepperth, "Propre: Projection and prediction for multimodal correlations learning. an application to pedestrians visual data discrimination," in *International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2014, pp. 2718–2725.

[14] M. Lefort, T. Kopinski, and A. Gepperth, "Multimodal space representation driven by self-evaluation of predictability," in *International Conference on Development and Learning and on Epigenetic Robotics (ICDL-EPIROB)*, 2014.

[15] K. Miller, D. Pinto, and D. Simons, "Processing in layer 4 of the neocortical circuit: new insights from visual and somatosensory cortex," *Current opinion in neurobiology*, vol. 11, no. 4, pp. 488–497, 2001.

[16] M. Mossio and D. Taraborelli, "Action-dependent perceptual invariants: From ecological to sensorimotor approaches," *Consciousness and cognition*, vol. 17, no. 4, pp. 1324–1340, 2008.

[17] P.-Y. Oudeyer, "Developmental robotics," *Encyclopedia of the Sciences of Learning*, 2011.

[18] P. Oudeyer, F. Kaplan, and V. Hafner, "Intrinsic motivation systems for autonomous mental development," *Evolutionary Computation, IEEE Transactions on*, vol. 11, no. 2, pp. 265–286, 2007.

[19] M. Radeau and P. Bertelson, "The after-effects of ventriloquism," *The Quarterly journal of experimental psychology*, vol. 26, no. 1, p. 63, 1974.

[20] B. Ridge, D. Skocaj, and A. Leonardis, "Self-supervised cross-modal online learning of basic object affordances for developmental robotic systems," in *Robotics and Automation (ICRA)*. IEEE, 2010, pp. 5047–5054.

[21] C. Schreiner, "Order and disorder in auditory cortical maps," *Current Opinion in Neurobiology*, vol. 5, no. 4, pp. 489–496, 1995.

[22] L. Shams and A. Seitz, "Benefits of multisensory learning," *Trends in cognitive sciences*, vol. 12, no. 11, pp. 411–417, 2008.

[23] C. Wessinger, M. Buonocore, C. Kussmaul, and G. Mangun, "Tonotopy in human auditory cortex examined with functional magnetic resonance imaging," *Human brain mapping*, vol. 5, no. 1, pp. 18–25, 1997.