Forward Models and the Prediction of Undesired Situations

No Author Given

No Institute Given

Abstract. Using forward models as a basic cognitive tool, the cornerstone of the research presented in this paper is the importance of prediction and action as part of the perceptual process of a cognitive system. An artificial agent equipped with a forward model is let to interact with its environment in order to learn the prediction of undesired situations. The forward models is implemented as an artificial neural network trained with data coming form a simulated agent. The network is tested and then implemented on-line on the simulated agent to solve an obstalce avoiding task while seeking a light source. The trained system learns to successfully predict a multimodal sensory representation formed by visual and tactile stimuli. The results presented here are very encouraging and represent the starting point for more research on the use and advantages that cognitive models can provide on artificial autonomous agents.

1 Introduction

The main objective of the research reported in this paper is the development of artificial agents capable of interacting with their environment making use of cognitive models. We are interested in studying these models to find out their capabilities to provide agents with the necessary tools to interact with their world.

The cornerstone of this research is the importance of predictions and actions as part of the perceptual process of the cognitive system. The cognitive model is learned and tested through interaction of the agent with its surroundings. The qualitative testing of the decisions the agent takes is based on the needs of the agent to act within its own world. For these reasons, we argue that the agent is grounded in its environment [9].

A widely accepted view of cognition explains behaviour as a product of a direct, unidirectional line of information processing. Sensory inputs create a sensory representation and according to this a motor action is performed, actions are regarded as reactions, responses to stimuli. Most of the observed behaviour is considered a consequence of an innate stimulus-response mechanism that is available to the individual [8]. Known as the *information processing metaphor*, this framework thinks of the perception processes as modules that receive, modify and then pass the information available from the environment.

A novel approach to perception considers sensory input and action (motor output) as part of the same cognitive process.

In the field of cognitive psychology these ideas have recently received much attention. A general framework is based on the ideomotor views of action control. These views stress the role that internal states such as goals or intended actions play as to the realisation of actions, disregarding to different extents the external sensory conditions.

Only in recent times the idea that the anticipation of actions and/or sensory states can influence behaviour has been appreciated. Anticipations are now seen to play an important role on the coordination, planning and realisation of behaviour [1]. The linear information processing approach has given way to new frameworks according to which the direction of information flow is not anymore a one-way path.

Among others, the ecologist views, championed by Gibson [3], suggest a direct link between action and perception. At the centre of this view is the importance played by the body of the agent and its dynamic relation with the environment. A more radical and innovative framework has been proposed by Hommel et. al [5]: The Theory of Event Coding (TEC) which is *"... based on the central notion that perception, attention, intention and action share, or operate on, a common representational domain"*[5].

As Gibson, TEC links perception and action functionally. It is this link and its coordination that provides the basis for adaptive behaviour [5].

Within this framework, sensory representations are also considered as consequences of actions. Any action realised by an agent on the environment has effects (action effects) and are the main reasons for behaviour. Representations that code for the environmental and bodily consequences of a movement become associated to motor representations coding for that actual movement [6]. The planning and control of actions becomes anticipatory when it is driven by the desired sensory situations or desired action effects.

1.1 Forward Models

A computational equivalent proposal are forward models. Mainly used in the field of motor control, a forward model is an internal model which incorporates knowledge about sensory changes produced by self-generated actions of an agent. Given a sensory situation S_t and a motor command M_t(intended or actual action) the forward model predicts the next sensory situation S_{t+1}.

Forward models provide an alternative to the classical approaches mentioned on Section **??**. Möller [7] suggested forward models as a possibility to integrate visual perception and action generation.

In the realm of artificial autonomous agents anticipation and forward models can be used as a base for coherent behaviour. Autonomous agents interact with their environment in a direct way. A basic need for them to deal with their world is to predict the events happening. An anticipatory agent learning and using a forward model should be able then to have sufficient information to form planning strategies avoiding undesired situations and reacting timely to the hazards of its environment. Very interesting results have been presented by Dearden et. al. [2]: a robot learns a forward model that successfully imitates actions presented to its visual system. Other implementations of cognitive approaches are discussed in Section 3.

The work presented in this paper attempts to provide an artificial agent with the necessary tools to predict undesired situations. The prediction of the agent is based on the inputs to the forward model, this prediction is characterized by the association of visual and tactile stimuli. This association can be considered an event composed by the motor command and the sensory situations (actual and desired) [5].

2 Experiments

The forward model is obtained by trainning an artificial neural network with data coming from a simulated agent, the network is then tested on trajectories not seen during training. The whole system is implemented on the artificial agent to solve an obstacle avoiding task while seeking for a light source.

2.1 Environment and Data Collection

Using a robot simulator (Left figure on Fig. 1) a robot is placed in an arena with obstacles varying in size from 25-50 pixels. In the figure the robot is moving forward from left to right.

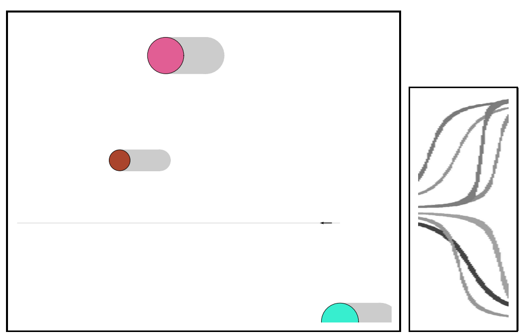

Fig. 1. *Virtual World of the robot and the recording of a single trajectory.*

The robot has a diameter of 30 pixels and is equipped with an omni-directional linear black and white camera and a simulated frontal bumper. The robot moves in a straight line through the arena, taking snapshots of the environment. Every 20 pixels a new image is recorded.the figure on teh right of Figure **??** shows an 80 steps or 1600 pixels trajectory.

In the image of the trajectory the y axis represents the spatial dimension of the image, this is, the 360 degrees of the robot view and the x axis represents the time dimension. The front of the robot is located in the middle of the image in the spatial dimension. In the first snapshot, the obstacles can be seen close together and of a relatively small size. As the robot moves forward, the obstacles grow and move away from the centre of the robot until they are at the back of it, (ie. in the far right and far left of the image).

Obstacles are randomly placed to the right, left and front of the robot trajectory. The task of the robot is to predict whether it can perform a collision free trajectory of 1600 pixels. Figure 2 shows two trajectories where a collision occurs obliging it to stop (for convinience the trajectories are rotated 90 degrees). As the robot aproaches obstacles these grow in the image view, the obstaces are located in the central area of the image as this area represents the front of the robot.

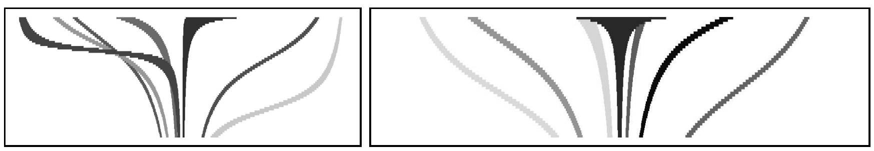

Fig. 2. *Trajectory of robot with a crash at the left. Trajectory length: 62 steps. And trajectory of robot with a crash at the front. Trajectory length: 50 steps.*

2.2 Data Preprocessing and Forward Model Implementation

Before being used in the system the visual information is preprocessed. Originally, the images coming from the robot camera have a size of 1000 x 1 pixels, individual trajectories have different lengths as once the robot encounters an obstacle it stops recording images. The trajectories are preprocessed as follows:

– As the system requires to predict collision on the front of the robot, the visual information at the front is the relevant one for this task. A section equivalent to 90 degrees (250 pixels) at the front of the robot is extracted from the whole image.
– A butterworth one dimensional low-pass filter is applied to the spatial dimension of the images. This works by applying a Fourier transform to the one dimensional image. The amplitudes of the Fourier coefficients are multiplied by a factor:

$$f = \frac{1}{1 + (\frac{k}{w})^{2n}} \tag{1}$$

with the values $w = 0.25$ for the width and $n = 0.1$ for the order. The inverse Fourier transform is then applied to the result. This is done in order to get rid of high frequency redundant information.

– Foveal mapping in the spatial domain. Foveal mapping is basically a weighted subsampling of the image. The farther pixels are from the centre of the image the more they are subsampled, the pixels at the centre remaining nearly unchanged. For the subsampling an averaging mask was used.

The effects of applying the preprocessing algorithms to the image shown in figure 1 can be seen in figure 3.

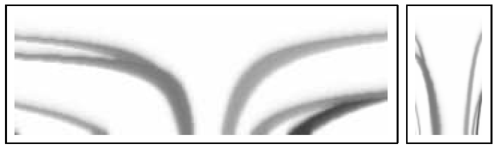

Fig. 3. *Frontal* 90^o *after filtering (left image) and after fovealisation (right image).*

A system is needed that is able to predict visual information as well as the simulated bumper states. This system can be implemented as a forward model of the form seen in Figure 4, where the current sensory situation is composed by the visual images V at times t, $t+1$ and $t+2$. This form of the input is expected to provide the model with the necessary information to learn the temporal structure of the data. The output of the forward model is the visual scene and the bumper state for time $t+3$: V_{t+3} and B_{t+3} respectively.

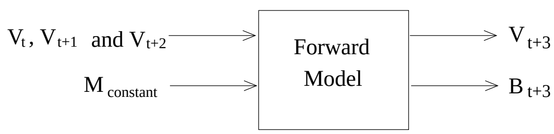

Fig. 4. *Particular Forward Model*

The system performs a local symmetrical prediction. This is, every predictor or forward model takes its input from a section of the sensory input and predicts only the central pixel of that section. On the edges of the image this is not possible, so the input window is shifted in order to use the available information.

Given that the final size of the images is 50 pixels, the system consists of 50 networks, multi-layer perceptrons trained with Resilient Back-propagation. Each network has a size of 45 input units (15 pixels for each time step), 10 hidden units and 2 output units, one for the predicted pixel and one for the predicted bumper value (0 for no collision and 1 for collision).

Training patterns for each network are prepared consisting of several images (46000 patterns) with a mixture of different collisions and collision-free trajectories. It is important to note that each one of the networks gets the same bumper value assigned so that when there is a collision *all* networks should have an activation value of 1.

2.3 Testing the Forward Model

After training the networks for 6000 cycles of batch training (all the training patterns are presented once before changes to the network are performed) the testing is done with trajectories that were unseen by the network during training. The networks are expected to perform two kinds of prediction. First a one step prediction (OSP), this is, given the values of V_t, V_{t+1} and V_{t+2} the values of V_{t+3} and B_{t+3} are predicted. This is the standard network output. Second a *long term prediction (LTP)* which consists of using the predicted visual data back as input to the system. This prediction compares to an internal simulation of the events.

The OSP bumper values are not binary, instead the activation increases as the robot approaches an obstacle. More importantly, the networks that increase activity are those on the side of the robot where the obstacle is approaching or where significant changes on the visual field occur. A threshold is implemented during OSP to indicate a collision, if 5 or more neurons show an activation higher than 0.5 it means there is a collision.

The system should have a necessity to trigger LTP and this is also defined as a threshold. When 3 or more bumper output neurons present an activation greater than 0.3 during OSP, an internal simulation of the rest of the trajectory starts. Although the threshold might seem low it guarantees that there is no false predictions. The system was tested on 30 different trajectories. It never failed to trigger LTP when there was in reality a (future) collision on the trajectory. Likewise, it never triggered the LTP when presented with collision-free trajectories.

The visual output a typical testing runs can be seen in Fig 5 which shows a trajectory with a collision on the left side of the robot. The system triggers LTP 2 steps before it actually happens, in the robot *world* this means 40 pixels or more than once its own size of 30 pixels.

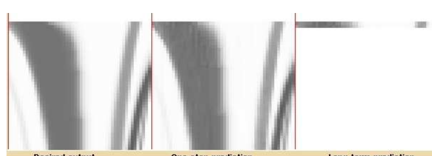

Fig. 5. *System evaluation during a collision on the left side of the robot*

Similar behaviours are observed during different trajectories. The system triggers LTP when a collision is likely to occur, most importantly during LPT the activation of the bumper units signals the presence of a crash.

The activation of the whole array of bumper output neurons for the last step of the LTP for these tests is presented in Fig. 6. It can be seen that the output neurons presenting higher activation are those located on the side of the robot where the collision is going to occur. In the case of the frontal collision, the activation is higher on the networks where changes in the visual field occur.

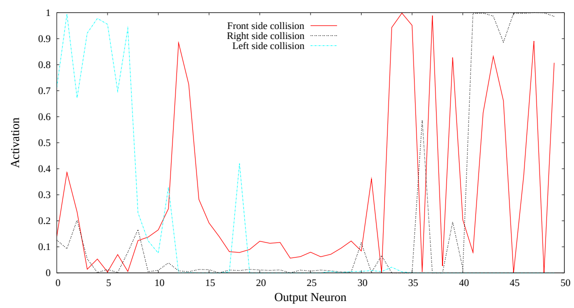

Fig. 6. *Activation of the output neurons for bumper states at the last step of LTP*

It is worth noting that Fig. 2 show trajectories and collisions before preprocessing. Fig 5 shows a trajectory with a collision after preprocesing, the fovealization process has the effect of *stretching* the central region of the real image. It is for this reason that in Fig. 5 the obstacle seems to be very far from the center of the image when in fact it is not. The existance of a crash is coded in the data, the system however, detects the crash correctly just due to the activation of the bumber output neurons.

2.4 Error Measure

A Sobolev norm is applied to the visual data, such that:

$$E_S = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{x_i})^2 + \frac{1}{n-1}\sum_{i=1}^{n-1}(y_i - \hat{y_i})^2 \tag{2}$$

where

$$\begin{aligned} y_i &= x_i - x_{i+1} \\ \hat{y_i} &= \hat{x_i} - \hat{x_{i+1}} \end{aligned} \tag{3}$$

The first term of Eq. 2 is the *Sum Squared Error (SSE)* between the real data x and the predicted data $\hat{x}$. The second term provides a measure not only of the difference between the two sets of data but of the posible existence of oscillation. The figure on the left on Fig. 7 shows the E_S for several trayectories during 40 time steps of One Step Prediction (OSP). During this kind of prediction the error is neglectabl. The figure on the right shows the second term of Eq. 2 for the same period of time and the same trajectories, showing that the oscilations between real data and prediction are minimal. The curves on the right stick to the general shape of their counterparts on the left.

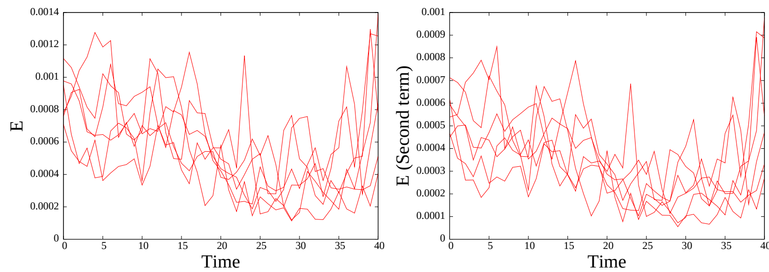

Fig. 7. *E_S and the second term of Eq. 2 during 40 steps of OSP*

Fig. 8 shows the same two errors during Long Term Predictions on different trayectories during 40 time steps. In these case the magnitude of the errors is more dramatic, however the behaviour of the curves for the second term of Eq. 2 is the same, showing low oscilation between real data and the predicted one.

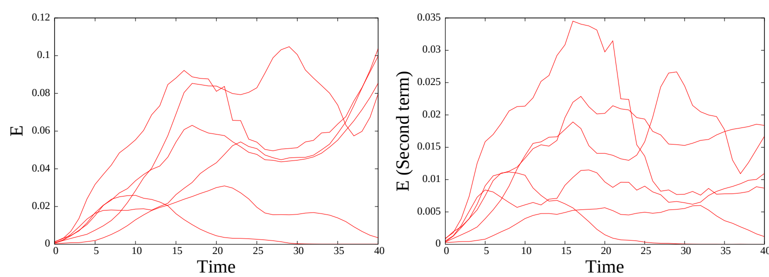

Fig. 8. *E_S and the second term of Eq. 2 during 40 steps of LTP*

The different shapes of the curves are due to the different starting points of the LTP as well as to the characteristics of the data. The two curves that tend to zero are predictions where no crash occurs, therefore the obstacles in the visual data dissappear.

2.5 Implementation

The trained forward model was implemented on the simulated agent used for collecting the data. The data coming from the camera was preprocessed and fed to the network as the agent was moving. The robot has the task of seeking a light source, avoiding collisions with nearby obstacles. The light sensing ability is completely independent of the vision or tactile senses of the robot. The robot, the light source and a number of objects are set on an initial random position.

Three behaviours are defined: a) light seeking, making the robot head directly into the light, b)prediction, if detecting high activation of the bumper units, the robot stops moving and performs an internal prediction and c) obstacle avoidance.

The same thresholds are used; when there is activation on the bumper units long term prediction (LTP) is triggered. In case LTP finds the possibility of a future crash the obstacle avoidance behaviour takes over the light seeking behaviour. Knowing that the activation of the bumper output units conveys information about the location of the obstacles this information is used to take avoiding action. Depending on the units that are registering the highest activation the robot decides where to turn.

Fig. 9 shows the robot seeking the light source hidden behind an obstacle, at the end of the trejectory the light source disappears underneath the robot. The robot performs a straight trajectory until the bumper units present high activation. The robot then performs an internal prediction which prompts it to change course.

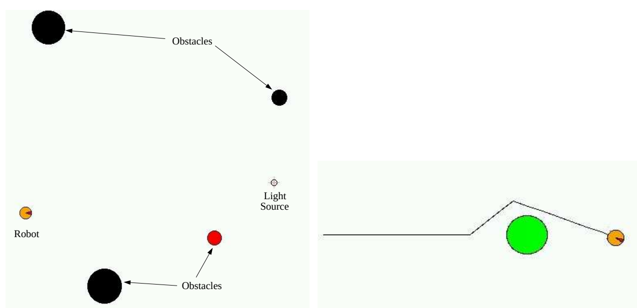

Fig. 9. *Robot simulator and a test run of the system*

Fig. 10 shows the robot making its way through a set of obstacles on its way towards the light source.

It is important to note that the robot can only predict sensory situations when the motor command has been constant for at least 3 steps. This means that once the robot turns, either by looking for the light or by avoiding an obstacle, it needs to move 3 steps with a constant motor command before being able to predict again.

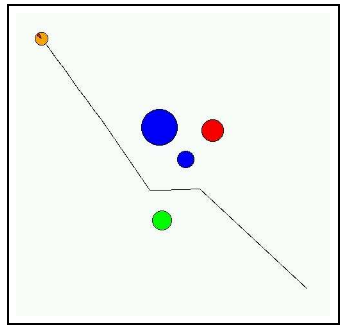

Fig. 10. *Robot seeking a light source and avoidning a set of obstacles.*

3 Conclusions

The presented experiments show that anticipation and forward models can provide agents with useful strategies. The agent presented here is capable of predicting the consequences of its own acts and at the same time take provisions for future actions. The system does this by learning an association between visual and tactile stimuli, which in the framework provided by TEC can be seen as an event.

It is interesting to observe the emergent behaviour shown by this system. Bumper activation values provide more information than expected. The results are easily extendable to problems requiring the agent to know where an obstacles is approaching from as it has been shown in the implementation of the model.

The system presented here differs from other works using forward models in important aspects. Hoffman et. al [4] presented a chain of forward models that provides an agent with the capability to select different actions to achieve a goal situation and perform mental transformations. The main difference with this system is that the forward model presented here performs the prediction of the same event (collision) using two sensor modalities (tactile and visual). The visual input to our forward model is completely grounded in that it does not have any sort of assumptions about distance. The system learns to estimate distance through interaction with its environment, associating visual input and tactile experience.

A form of anticipation is presented by Ziemke et. al [10], however their results for long term prediction are very constrained by the environment in which the robots evolved. In our case the model is capable of solving diferent scenarios, performing the necessary predictions and reacting timely to obstacles on its path.

Although the task presented here can be solved by different and simpler systems, this model presents the advantage of anticipation. The implementation of more complex tasks in the robot environment should prove that an anticipatory agent is capable of avoiding typical problems in which a reactive agent fails such as corners and dead-end situations.

Further use of the system here presented can include behaviour in which the understanding of ego motion and its consecuences is necessary. The system, as it is, should be able to differentiate between, on the one hand, the changes brought up in the environment because of its own movement and on the other, the changes in the environment brought up by other agents and their respective actions.

The implementation of forward models that make use of non-constant motor commands would get rid of the need of three consecutive straight movements in order to be able to predict. Such a system should present further planning abilities to solve tasks such as homing, coherent exploration of the environment and, generally speaking, behaviours showing that the agent has a good understanding of its environment.

The multi sensory representation of stimuli (visual and tactile) should also be present in the internal dynamics of the network. These dynamics should become more interesting with non-constant motor commands. As proposed by TEC [5], hidden units of the system would share resources when coding of actions and multi-sensory perception.

References

1. Butz, M. V., Sigaud, O., and Gerard, P. Internal models and anticipations in adaptive learning systems. In *Adaptive Behaviour in anticipatory learning systems (ABiALS'02)*, P. Butz M. V., Gerard and S. O., Eds. Edinburgh, Scotland, 2002.
2. Dearden, A., and Demiris., Y. Learning forward models for robots. In *Int. Joint Conferences on Artificial Intelligence* (Edimburgh, 2005), p. 1440.
3. Gibson, J. J. *The ecological approach to visual perception.* Houghton Mifflin, Boston, 1979.
4. Hoffmann, H., and R.Moeller. Action selection and mental transformation based on a chain of forward models. In *Proc. of the 8th Int. Conference on the Simulation of Adaptive Behavior* (Cambridge, MA, 2004), MIT Press, pp. 213–222.
5. Hommel, B., Müsseler, J., Aschersleben, G., and Prinz, W. The theory of event coding (TEC): A framework for perception and action planning. *Behavioral and Brain Sciences 24* (2001), 849–937.
6. Knoblich, G., and Prinz, W. Linking perception and action: An ideomotor approach. In *Higher-order motor disorders*, H.-J. Freund, M. Jeannerod, M. Hallett, and R. C. Leiguarda, Eds. Oxford University Press, Oxford, UK, 2005.
7. Möller, R. Perception through anticipation. A behavior-based approach to visual perception. In *Understanding Representation in the Cognitive Sciences*, A. Riegler, A. von Stein, and M. Peschl, Eds. Kluwer Academic Publishers, New York, 1999, pp. 169–176.
8. Witkowski, M. Anticipatory learning: the animat as a discovery engine. In *Adaptive Behaviour in anticipatory learning systems (ABiALS'02)*, B. M. V., P. Gerard, and S. O., Eds. Edinburgh, Scotland, 2002.
9. Ziemke, T. Rethinking grounding. In *Understanding Representation in the Cognitive Sciences*, A. Riegler, M. Peschl, and A. von Stein, Eds. Kluwer Academic Publishers, New York, 1999, pp. 177–190.
10. Ziemke, T., Jirenhed, D.-A., and Hesslow, G. Internal simulation of perception: a minimal neuro-robotic model. *Neurocomputing 68* (2005), 85–104.