A Situated Model for Sensory-motor Coordination in Gaze Control

G. de Croon , E.O. Postma , H.J. van den Herik

IKAT, Universiteit Maastricht, P.O. Box 616, 6200 MD, Maastricht, The Netherlands. Voice: 0031-433883477 Fax: 0031-433884897

Abstract

This paper shows that sensory-motor coordination contributes to the performance of situated models on the high-level task of artificial gaze control for gender recognition in static natural images. To investigate the advantage of sensory-motor coordination, we compare a non-situated model of gaze control with a situated model. The non-situated model is incapable of sensory-motor coordination. It shifts the gaze according to a fixed set of locations, optimised by an evolutionary algorithm. The situated model determines gaze shifts on the basis of local inputs in a visual scene. An evolutionary algorithm optimises the model's gaze control policy. In the experiments performed, the situated model outperforms the non-situated model. By adopting a Bayesian framework, we show that the mechanism of sensory-motor coordination is the cause of this performance difference. The essence is that the mechanism maximises task-specific information in the observations over time, by establishing dependencies between multiple actions and observations.

Key words: Situated model, Sensory-motor coordination, Gaze control, Active perception, Bayesian framework, Evolutionary algorithm

1 Introduction

A situated model of intelligence is a model in which the motor actions co-determine the future sensory inputs (Pfeifer and Scheier, 1999). The combination of sensory inputs and motor actions forms a closed loop. A situated model can exploit the closed loop in such a way that the performance on a particular

Email addresses: g.decroon@cs.unimaas.nl (G. de Croon), postma@cs.unimaas.nl (E.O. Postma), herik@cs.unimaas.nl (H.J. van den Herik).

task is optimised, i.e., it can employ sensory-motor coordination (Pfeifer and Scheier, 1999). Hence, a situated model of intelligence may use sensory-motor coordination to solve specific tasks (Pfeifer and Scheier, 1999; O'Regan and Noë, 2001).

Several studies have investigated the mechanism of sensory-motor coordination, either with a physical robot (Nolfi, 2002; Nolfi and Marocco, 2002), or in simulation (Beer, 2003; van Dartel et al., in press). They show that sensory-motor coordination facilitates the execution of tasks, so that the performance is enhanced. In addition, they investigate the how and why of this enhancement. So far, the research has mainly focused on low-level tasks, e.g., classifying geometrical forms (Floreano et al., 2004). As a consequence, this paper investigates the following research question: *Can sensory-motor coordination contribute to the performance of situated models on high-level tasks?* In this paper we restrict ourselves to the analysis of two models both performing the same task, viz., gaze control for gender recognition in static natural images. The motivation for the choice of this task is three-fold: (1) it is a challenging task, to which no situated gaze-control models have been applied so far; (2) the use of a simulated model and static images (instead of a physical robot in a realistic environment) saves optimisation time (Floreano et al., 2004) and facilitates analysis, while preserving the possibility to study principles of sensory-motor coordination (Pfeifer and Scheier, 1999); (3) it enables the comparison of two models that differ only in their ability to coordinate sensory inputs and motor actions. We will compare a non-situated with a situated model of gaze control. If the situated model's performance is better, we focus on a second research question: *How does the mechanism of sensory-motor coordination enhance the performance of the situated model on the task?* We explicitly state that we are interested in the relative performance of the models and the cause of a possible difference in performance. It is not our intention to build a gender-recognition system with the best classification performance. Our only requirement is that the models perform above chance level (say 60% to 80%), so that a comparison is possible.

The remainder of the paper is organised as follows. In Section 2 we describe the non-situated and the situated model of gaze control. In Section 3 we outline the experiment used to compare the two models of gaze control. In Section 4 we show the experimental results and analyse the gaze control policies involved within a Bayesian framework. In Section 5 we discuss the relevance of the results. Finally, we draw our conclusions in Section 6.

2 Two Models of Gaze Control

Below, we describe the non-situated model of gaze control (Section 2.1) and the situated model of gaze control (Section 2.2). Then we discuss the adaptable parameters of both models (Section 2.3).

2.1 Non-situated Model of Gaze Control

The non-situated model consists of three modules, as illustrated in Fig. 1 by the dashed boxes, labelled ‘I’, ‘II’, and ‘III’. The first module receives as sensory input the raw input from the window with centre ‘x’, the current fixation location. In Fig. 1 the raw input is shown on the left in box I; it contains a part of the face. From that window, input features are extracted (to be described later). These input features serve as input to the second module, a neural network. The input layer of the neural network is depicted by the box ‘input layer’. Subsequently, the neural network calculates the activations of the hidden neurons in the ‘hidden layer’ and of the output neuron in the ‘output layer’. There is one output neuron that indicates the class of the image. The third module (left in Fig. 1) determines the next fixation location, where the process is repeated. Below we describe the three modules of the non-situated model of gaze control in more detail.

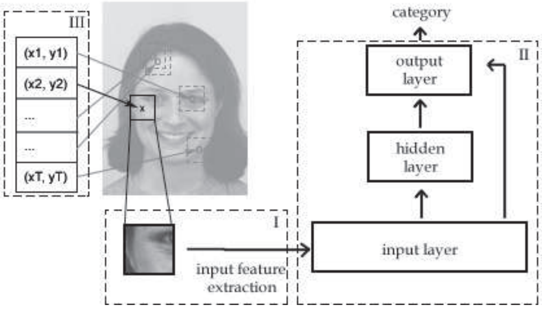

Fig. 1. Overview of the non-situated model of gaze control.

2.1.1 Module I: Sensory Input.

In this subsection, we focus on the extraction procedure of the input features. For our research, we adopt the set of input features as introduced in (Viola and Jones, 2001), but we apply them differently.

An input feature represents the difference in mean light intensity between two areas in the raw input window. These areas are determined by the feature’s type and location. Figure 2 shows eight different types of input features (top row) and nine differently sized locations in the raw input window from which the input features can be extracted (middle row, left). The sizes vary from the whole raw input window to a quarter of the raw input window. In total, there are $8 \times 9 = 72$ different input features. In the figure, two example input features are given (middle row, right). Example feature 'L' is a combination of the first type and the second location, example feature 'R' of the third type and the sixth location. The bottom row of the figure illustrates how an input feature is calculated, namely by subtracting the mean light intensity in the image covered by the grey surface from the mean light intensity in the image covered by the white surface. The result is a real number in the interval $[-1, 1]$. In the case of example feature L, only the left half of the raw input window is involved in the calculation. The mean light intensity in the raw input window of area 'A' is subtracted from the mean light intensity of area 'B'.

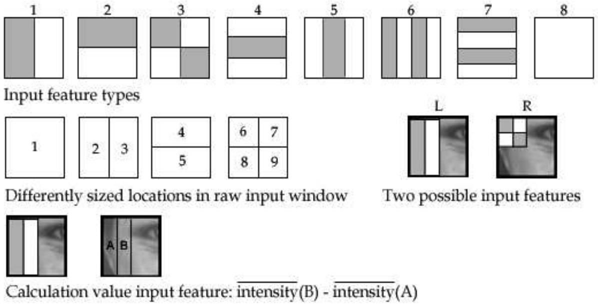
1
2
3
4
5
6
7
8
Input feature types
1
2
3
4
5
6
7
8
9
Differently sized locations in raw input window
L
R
Two possible input features
A
B
Calculation value input feature: $\overline{\text{intensity}}(B)$ - $\overline{\text{intensity}}(A)$

Fig. 2. An input feature consists of a type and a location.

2.1.2 Module II: Neural Network.

The second module is a neural network that takes the extracted input features as inputs. It is a fully-connected feedforward neural network with h hidden neurons and one output neuron. The hidden and output neurons all have sigmoid activation functions: $a(x) = \tanh(x)$, $a(x) \in \langle -1, 1 \rangle$. The activation of the output neuron, o_1, determines the classification (c) as follows.

$$c = \begin{cases} \text{Male} & \text{, if } o_1 > 0 \\ \text{Female} & \text{, if } o_1 \leq 0 \end{cases} \tag{1}$$

2.1.3 Module III: Fixation locations.

The third module controls the gaze in such a way, that for every image the same locations in the image are fixated. It contains coordinates that represent *all* locations fixated by the non-situated model. The model first shifts its gaze to location (x_1, y_1) and then classifies the image. Subsequently, it fixates the next location, (x_2, y_2), and again classifies the image. This process continues, until the model has fixated all locations from (x_1, y_1) to (x_T, y_T) in sequence, assigning a class to the image at every fixation. The performance is based on these classifications (see Section 3.2). Out of all locations in an image, an evolutionary algorithm selects the T fixation locations. Selecting the fixation locations also implies selecting the order in which they are fixated.

2.2 Situated Model of Gaze Control

The situated model of gaze control (inspired by the model in (Floreano et al., 2004)) is almost identical to the non-situated model of gaze control. The only difference is that the gaze shifts of the situated model are not determined by a third module, but by the neural network (Fig. 3). Therefore, the situated model has only two modules. Consequently, the current neural network has three output neurons. The first output neuron indicates the classification as in (1). The second and the third output neurons determine a gaze shift (Δx, Δy) as follows.

$$\Delta x = \lfloor m o_2 \rfloor \tag{2}$$

$$\Delta y = \lfloor m o_3 \rfloor, \tag{3}$$

where o_i, $i \in \{2, 3\}$, are the activations of the second and third output neurons. Moreover, m is the maximum number of pixels that the gaze can shift in the x- or y-direction. As a result, Δx and Δy are expressed in pixels. If a shift results in a fixation location outside of the image, the fixation location is repositioned to the nearest possible fixation location. In Fig. 3 'x' represents the current fixation location, and 'o' represents the new fixation location as determined by the neural network.

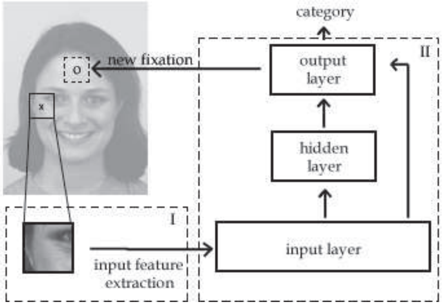

Fig. 3. Overview of the situated model of gaze control.

2.3 Adaptable Parameters

Above we described both the non-situated and the situated model of gaze control. In our experiments below, four parameter types will define specific instantiations of the non-situated model. We refer to these instantiations as agents. The four parameter types are: the input features, the spatial scale of the raw input window from which features are extracted, the neural network weights, and the coordinates of all fixation locations. Specific instantiations of the situated model are defined by the first three parameter types mentioned above. In both cases an evolutionary algorithm generates and optimises the agents (i.e., parameter values) by evaluating their performance on the gaze-control task. In our experiments, we do not evolve the neural networks' structures, in order to facilitate the comparison between both models.

3 Experimental Setup

In this section, we describe the gender-recognition task on which we compare the non-situated and the situated model of gaze control (Section 3.1). In addition, we discuss the evolutionary algorithm that optimises the models' adaptable parameters (Section 3.2). Finally, we mention the experimental settings (Section 3.3).

3.1 Gender-Recognition Task

Below, we motivate our choice for the task of gender recognition. Then we describe the data set used for the experiment. Finally, we outline the procedure of training and testing the two types of gaze-control models.

We choose the task of gender recognition in images containing photos of female or male faces, since it is a challenging and well-studied task (Bruce and Young, 2000). There are many differences between male and female faces that can be exploited by gender-recognition algorithms (Moghaddam and Yang, 2002; Calder et al., 2001). State-of-the-art algorithms use global features, extracted in a non-situated manner. So far, none of the current algorithms is based on gaze control with a local fixation window.

The data set for the experiment consists of images from J.E. Litton of the Karolinska Institutet in Sweden. It contains 278 images with angry-looking and happy-looking human subjects. These images are converted to gray-scale images and resized to 600×800 pixels.

One half of the image set serves as a training set for both the non-situated and the situated model of gaze control. Both models have to determine whether an image contains a photo of a male or female, based on the input features extracted from the gray-scale images. For the non-situated model, the sequence of T fixation locations is optimised by an evolutionary algorithm. For the situated model, the initial fixation location is defined to be the centre of the image and the subsequent $T-1$ fixation locations are determined by the gaze-shift output values of the neural network (outputs o_2 and o_3). At every fixation, the models have to assign a class to the image. After optimising classification on the training set, the remaining half of the image set is used as a test set to determine the performance of the optimised gaze-control models. Both training set and test set consist of 50% males and 50% females.

To assess the generalisation to other tasks, we also apply both the non-situated and situated model of gaze control to a task of facial expression recognition, using the same image set. Instead of classifying the images according to gender, the models have to classify the same images in terms of the expression (happy and angry).

3.2 Evolutionary Algorithm

As stated in Section 2.3, an evolutionary algorithm optimises the parameter values that define the non-situated and the situated agents, i.e., instantiations of the non-situated and situated model, respectively. There are three reasons for selecting an evolutionary algorithm as our training paradigm. First, the recognition task involves two subtasks, the selection of the next location and the classification of the contents. The model has to be optimised for both subtasks simultaneously. Evolutionary algorithms are capable of, and very suitable for, such a simultaneous optimisation (Zitzler, 2002; Khare et al., 2003). Second, the neural network of the situated model cannot be trained by a gradient-based method such as backpropagation, because for gaze control the desired outputs are unknown. Third, evolutionary algorithms are effective in avoiding local maxima when applied to the optimisation of neural networks (Yao, 1999).

In our experiment, we perform 15 independent ‘evolutionary runs’ to obtain a reliable estimate of the average performance. Each evolutionary run starts by creating an initial population of M randomly initialised agents. Each agent operates on every image in the training set, and its performance is determined by the following fitness function:

$$f(a) = \frac{t_{c,I}}{IT} \ , \qquad (4)$$

in which a represents the agent, $t_{c,I}$ is the number of time steps at which the agent correctly classified images from the training set, I is the number of images in the training set, and T is the total number of time steps (fixations) per image. We note that the product IT is a constant that normalises the performance measure. The $\frac{M}{2}$ agents with the highest performance are selected to form the population of the next generation. Their adaptable parameter sets are mutated with probability P_f for the input feature parameters and P_g for the other parameters, e.g., representing coordinates or network weights. If mutation occurs, a feature parameter is perturbed by adding a random number drawn from the interval $[-p_f, p_f]$. For other types of parameters, this interval is $[-p_g, p_g]$. In our evolutionary algorithm we do not apply crossover, since it might produce more harm than benefit in evolving multilayer perceptrons (Yao, 1999). For every evolutionary run, the selection and reproduction operations are performed for G generations.

3.3 Experimental Settings

In our experiment the models use ten input features. Furthermore, the neural networks of both models have 3 hidden neurons, $h = 3$. All weights of the neural networks are constrained to a fixed interval $[-r, r]$. Since preliminary experiments showed that evolved weights were often close to 0, we have chosen the weight range to be $[-1, 1]$, $r = 1$. The scale of the window from which the input features are extracted ranges from 50 to 150 pixels. Preliminary experiments showed that this range of scales is large enough to allow gender recognition, and small enough for local processing, which requires intelligent gaze control. The situated model's maximal gaze shift m is set to 500, so that the model can reach almost all locations in the image in one time step.

For the evolutionary algorithm we have chosen the following parameter settings: $M = 30$, $G = 300$, and $T = 5$. The choice of T turns out not to be critical to the results with respect to the difference in performance of the two models (see Section 4.2.3). The mutation parameters are: $P_f = 0.02$, $P_g = 0.10$, $p_f = 0.5$, and $p_g = 0.1$.

4 Results

In this section, we show the performances of both models (Section 4.1). Then we analyse the best situated agent to gain insight into the mechanism of sensory-motor coordination (Section 4.2).

4.1 Performance

Table 1 shows the mean performances on the test set (and standard deviation) for the gender recognition task of the best agents of the 15 evolutionary runs. Performance is expressed as the proportion of correct classifications. The table shows that for the gender recognition task, the mean performance of the best situated agents is 0.15 higher than that of the best non-situated agents. Figure 4 shows the histograms of the best performances obtained in the 15 runs for non-situated agents (white) and for situated agents (gray). Since both distributions of the performances are highly skewed, we applied a bootstrap method (Cohen, 1995) to test the statistical significance of the results. It revealed that the difference between the mean performances of the two types of agents is significant ($p < 0.05$).

Table 1. Mean performance ($\bar{\mathrm{f}}$) and standard deviation (σ) of the performance on the test set of the best agents of the evolutionary runs.

Gender Task	$\bar{\mathrm{f}}$ ($\pm\sigma$)
Non-situated	0.60($\pm$0.057)
Situated	0.75($\pm$0.055)
Expression Task	$\bar{\mathrm{f}}$ ($\pm\sigma$)
Non-situated	0.55($\pm$0.036)
Situated	0.77($\pm$0.053)

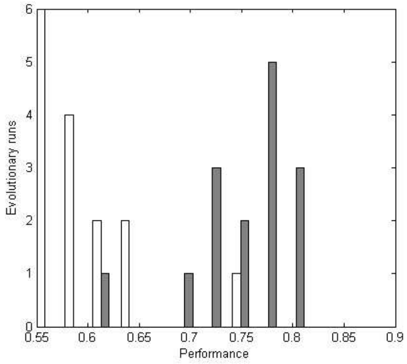

Fig. 4. Histograms of the best fitness of each evolutionary run. White bars are for non-situated agents, gray bars for situated agents.

Table 1 also shows the results for the facial expression recognition task. Also for this task, there is a statistically significant difference between the non-situated model and situated model of gaze control (bootstrap method, $p < 0.05$). This result suggests that the superiority of the situated model generalises to other recognition tasks. Below, we analyse the results of the gender recognition task.

4.2 Analysis

To understand how the situated agents exploit the closed loop of actions and inputs, we examine their behaviour within a Bayesian framework. After introducing this framework, we analyse the evolved gaze-control policy of the best situated agent of all evolutionary runs. The analysis clarifies how sensory-motor coordination optimises the performance on the gender-recognition task.

4.2.1 Bayesian framework

The behaviour of our situated agents is best explained within a Bayesian framework (see (Mitchell, 1997) for an introductory review). As explained in Section 2.2, the agents contain a single neural network that outputs both a class and a gaze shift. In this subsection we will refer to the function represented by the network that maps an input feature vector to a class as the agent's 'classifier' and the function that maps an input feature vector to a gaze shift as the agent's 'controller'. We first explain the role of the agent's classifier and then explain the role of the controller that determines the gaze shifts.

The classifier is evaluated (and trained) on its classification of the input feature vectors at all time steps t ($t \in \{1, 2, \ldots, T\}$) of all images i in the training set ($i \in \{1, 2, \ldots, I\}$), see Eq. (4). It has to learn a mapping from a vector containing all input feature vectors to a vector containing all corresponding classes, i.e., a mapping from a vector $\mathbf{o} = (o_{11}, o_{12}, \ldots, o_{1T}, o_{21}, o_{22}, \ldots, o_{2T}, \ldots, o_{I1}, o_{I2}, \ldots, o_{IT})$ to a vector $\mathbf{g} = (c_{11}, c_{12}, \ldots, c_{1T}, c_{21}, c_{22}, \ldots, c_{2T}, \ldots, c_{I1}, c_{I2}, \ldots, c_{IT})$, where each o_{it} is an input feature vector of ten real numbers in $[-1, 1]$ extracted from image i at time step t. We refer to such an input feature vector as an 'observation'. Each c_{it} is the class of the associated image i, $c_{it} = c_i \in C$, $C = \{M, F\}$.

Bayes' rule allows finding a suitable mapping from $\mathbf{o}$ to $\mathbf{g}$. It states that for all different classes c_j: $P(c_j \mid o_{it}) = P(o_{it} \mid c_j)\ P(c_j)/P(o_{it})$, where $P(c_j \mid o_{it})$ is the probability of class c_j, given a single observation o_{it}. In Bayesian learning this probability, referred to as the 'posterior probability', is determined for all classes c_j to find the class with maximal posterior probability [1]. The goal of learning is to find a classifier that always returns the class with maximal posterior probability, referred to as a maximal a posteriori classifier. In our experiments the classifier is trained by the evolutionary algorithm to maximise the classification performance on the training set. As a consequence, the classifier approximates a maximal a posteriori classifier.

To understand the role of the situated agent's controller, the notion of entropy is of importance. The performance of a maximal a posteriori classifier depends on the entropy of the posterior probabilities for the two classes. A maximal entropy of the posterior probabilities indicates that given the observation, both classes are equally probable. A minimal entropy indicates that given the observation, only one class is possible. A lower entropy of the posterior probabilities improves the performance of the classifier. Many active-vision approaches to classification aim to minimise entropy by performing multiple observations.

[1] The class with maximal posterior probability is usually found by maximising the maximal likelihood: $c_i = \text{argmax}_{c_j \in C} P(o_{it} \mid c_j)$, under the assumption that $P(c_j)$ is equal for every class.

For instance, in (Kröse and Bunschoten, 1999) class probabilities are estimated in a robot localisation task. The class is determined by performing multiple observation to optimise the probability $P(c_j \mid o_{i1}, o_{i2}, \ldots, o_{iT})$. The feedforward neural-network classifier employed in our experiments is not capable of optimising this probability, since it only has access to one observation at a time.

Both non-situated agents and situated agents *can* minimise the entropy of the posterior probabilities by means of choosing fixation locations. Since both types of agents have the same type of classifier, the cause of the better performance of the situated agent must be that it is better at finding observations $\mathbf{o}$ that are easier to map to the class vector $\mathbf{g}$. What is the difference between non-situated and situated agents that can explain this? The situated agent's controller can influence subsequent observations on the basis of the current observation. This influence allows the situated agent to exploit multiple observations, in contrast to the non-situated agent.

4.2.2 Gaze Behaviour per Class.

In the previous subsection, we suggested that the controller gathers observations that minimise the entropy of the posterior probabilities, i.e., observations that are easier to classify. In this subsection we present some empirical evidence for our suggestion. We provide the best situated agent with observations typical for male or female images and then analyse the resulting gaze path to assess class-dependent behaviour. The analysis shows that the agent fixates locations at which its classifier performs well, i.e., where the entropy of the posterior probabilities is low. In addition, it shows that the agent exploits multiple observations. Namely, one of the ways in which the agent obtains different observations for male and female images is by fixating different locations for both classes.

To obtain an impression of the fixation locations at which the classifier performs well for male or for female images, we measure the situated agent's classification performance on the training set at all positions of a 100 × 100 grid superimposed on the image. At every position we determine the classification performance for both classes. The 'local average performance' for male images is: $k_m(x, y) \ / \ I_m$, where $k_m(x, y)$ is the number of correctly classified male images at location (x, y), and I_m is the total number of male images in the training set. This local average performance is closely related to the entropy of the posterior probabilities: high performance corresponds with low entropy and vice versa. The left part of Fig. 5 shows a picture of the local average performances represented as intensities for all locations. The highest intensity represents perfect classification. The left part of Fig. 6 shows the local average performances for female images. The figures show that dark areas in Fig. 5 tend to have high intensity in Fig. 6 and vice versa. Hence, there is an obvious trade-off between good classification of males and good classification of females [2]. The presence of a trade-off implies that classification of males and females should ideally take place at different locations.

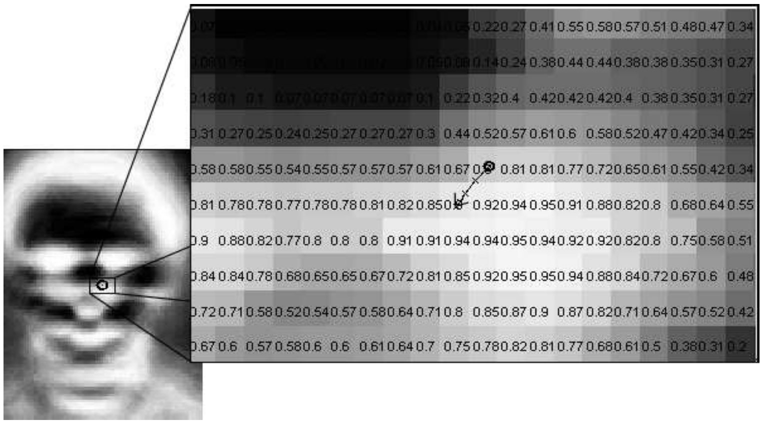

Fig. 5. Local average performances of male images in the training set. The arrow superimposed on the large inset represents the gaze path of the agent, when it receives inputs typical for male images.

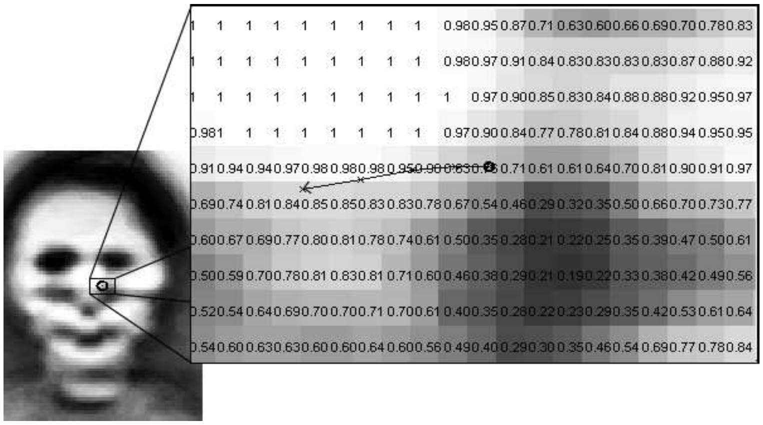

Fig. 6. Local average performances of female images in the training set. The arrow superimposed on the large inset represents the gaze path of the agent, when it receives inputs typical for female images.

We analyse the gaze path that originates when we provide the situated agent with either typically male observations at all fixation locations or typically female observations at all fixation locations. To determine a 'typically male observation' at a given location, we extract the input feature vector at that location for every male image in the training set and calculate the average input feature vector. We determine a 'typically female observation' in the

[2] Note that the images are not inverted copies: in locations where male and female inputs are quite different, good classification for both classes can be achieved.

same way, but for the female images in the training set. The right part of Fig. 5 zooms in on the picture and shows the gaze path that results when the agent receives typical male inputs at all fixation locations. The first fixation location is indicated by an 'o'-sign, the last fixation location by an arrow. Intermediate fixations are represented with the 'x'-sign. The black lines in Fig. 5 connect the fixation locations. The agent moves from a region with performance 0.80 to a region with performance 0.90. The right part of Fig. 6 shows the same information for images containing females, revealing a movement from a region with a performance of 0.76 through a region with a performance of 0.98. This shows that the agent fixates locations at which its classifier performs well on the class associated with the observations. Both figures also show that the situated agent takes misclassifications into account: it avoids areas in which the performance for the alternative class are too low. For example, if we look at the right part of Fig. 5, we see that the agent fixates locations to the bottom left of the starting fixation, while the local average performance is even higher to the bottom right. The reason for this behaviour is that in that area, the performance for female images is rather low (Fig. 6).

In summary, the agent fixates different locations for different observation histories (typically male or typically female). This implies that the agent exploits multiple observations to optimise its performance; the agent uses multiple gaze shifts, i.e. uses multiple observations of a certain type, to reach better classification areas. Non-situated agents cannot exploit multiple observations to fixate better classification areas, since the fixation locations are determined in advance for all images. As a consequence, non-situated agents cannot fixate different locations for the two classes.

4.2.3 Gaze Behaviour per Specific Image.

For specific images, the agent often deviates from the general gaze path, venturing even into regions of the image that have low local average performances. However, for specific images that differ considerably from the average, these areas might be well-suited for classification. In this subsection, we demonstrate that the situated agent exploits the structure of specific images. In addition, we show that the agent's gaze behaviour enhances classification performance over time.

We demonstrate that the situated agent exploits the structure of specific images by comparing its performance with a predicted performance that is based on the local average performances. The predicted performance of the agent at time step t in image i is defined as the local average performance at the agent's fixation location at time step t in image i.

Figure 7 shows both the actual performance (solid lines) and the predicted

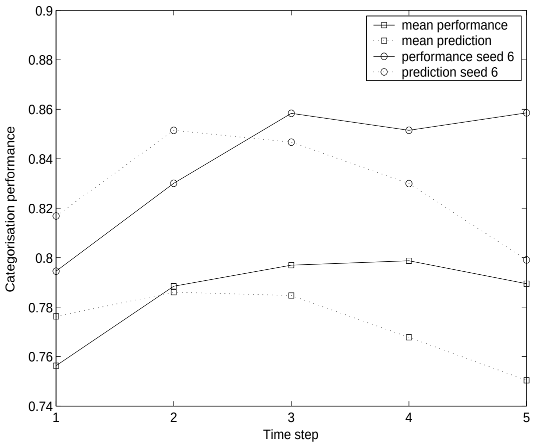

Fig. 7. Actual performance (solid lines) and the predicted performance (dotted lines) over time, for the best situated agent in particular (circles), and averaged over all situated agents (squares).

performance (dotted lines) over time, for the best situated agent in particular (circles), and averaged over all situated agents (squares). For the last three time steps the actual performances of the situated agents are consistently higher than the predicted performances. Apparently, the actual performance at a location is better than the average performance at that location. Hence, the situated agents use their observations to select subsequent image-specific fixation locations that lead to good classification.

We now illustrate how the best situated agent exploits the structure of specific images. The best situated agent bases its classification partly on the eyebrows of a person. If the eyebrows of a male are lifted higher than usual, the agent occasionally fixates a location right and above of the starting fixation. On average, this location is not good for male classification (its local average performance is 0.57, see Fig. 5), since in our training set eyebrows are usually not lifted. However, for some specific images it *is* a good area, because it contains a (part of a) lifted eyebrow. Since the situated agent only fixates this area if the eyebrows are lifted, the actual performance for such images is higher than the predicted performance.

The final result of the situated agents' gaze behaviours is the optimisation of the (actual) performance over time. Figure 7 shows that the actual performance increases after $t = 1$. The fact that performance generally increases over time suggests that sensory-motor coordination establishes dependencies between multiple actions and observations that are exploited to optimise classification performance.

As mentioned in Section 3.3, other settings of T ($T > 1$) lead to similar results. Fig. 8 shows the influence of the total number of time steps T, for $T \in \{5, 10, 15, 20\}$, on the mean performances of non-situated agents (dashed-dotted line) and situated agents (solid line). It also shows the standard errors for the mean performances. All mean performances are based on 15 evolutionary runs.

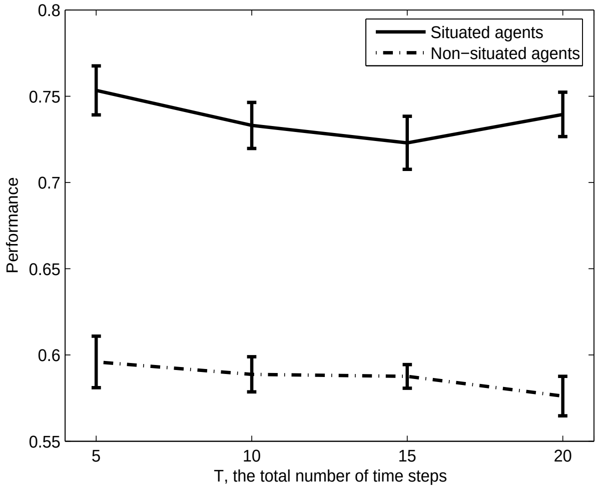

Fig. 8. Influence of the parameter T on the mean performance of non-situated agents (dashed-dotted line) and situated agents (solid line).

4.2.4 Entropy over Time.

In subsection 4.2.1, we argued that the situated model's controller minimises the entropy of the posterior distribution over time. In this subsection, we estimate the entropy of the posterior distribution at each time step for the best situated agent. By treating the agent's observations as elements of a set O, we calculate the entropy of the posterior distribution as follows. We define the entropy $E(t)$ at a time step t as the weighted sum of the Shannon entropies (Shannon, 1948) of the posterior distributions of all observations $o \in O$ at that time step:

$$E(t) = \sum_{o \,\in\, O} P(o,t)\ H(P(C \mid o,t)) \tag{5}$$

$$H(P(C \mid o,t)) = \sum_{c \,\in\, C} P(c \mid o,t) \log_2(\frac{1}{P(c \mid o,t)}) \tag{6}$$

, in which $P(o,t)$ is the probability of observation o at time step t and $H(P(C \mid o,t))$ is the Shannon entropy of the posterior probability distribution for observation o at time step t. Furthermore, C is the set of all mutually exclusive classes c.

In order to get reliable estimates of $P(o,t)$ and $P(c \mid o,t)$, we reduce the large number of input feature vectors by mapping them onto a limited set of prototypes. We perform this mapping with k–means clustering (see, e.g., (Jain et al., 1999)). The value of k indicates the number of clusters and is therefore proportional to the amount of overlap (Euclidian distance) between the prototypes. To reduce the overlap, we select small values of k. We have performed k-means clustering with a Euclidian distance measure for various choices of k. Figure 9 shows the performance of the best situated agent over time and the entropy over time for $k = 10$ and for $k = 15$, averaged over 30 runs of the k–means algorithm. The figure also includes the standard errors associated with the average values.

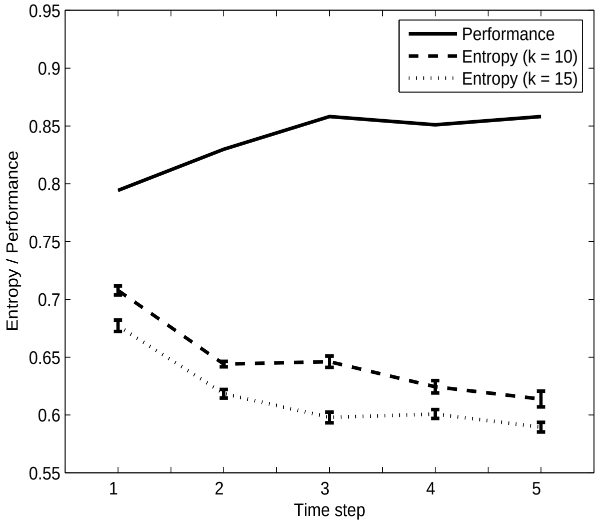

Fig. 9. Entropy over time of the best situated agent averaged over 30 runs of the k–means algorithm for $k = 10$ (dashed line) and $k = 15$ (dotted line). The performance over time is also shown (solid line).

The figure illustrates that the entropy of the posterior distributions decreases over time. This is not only the case for the best situated agent. Figure 10 shows the entropy over time ($k = 10$) for both the non-situated model and the situated model, averaged over all 15 best agents of the evolutionary runs.

Figure 10 illustrates that, on average, the entropy decreases over time for situated agents. The average entropy seems to decrease (non-monotonously) for non-situated agents as well, but the magnitudes of the error bars indicate that the entropies vary significantly. In fact, the entropies of non-situated agents do not decrease or increase reliably over time.

Since the entropy of a distribution is inversely related to its information, our results indicate that the situated model's controller maximises task-specific information in the observations over time.

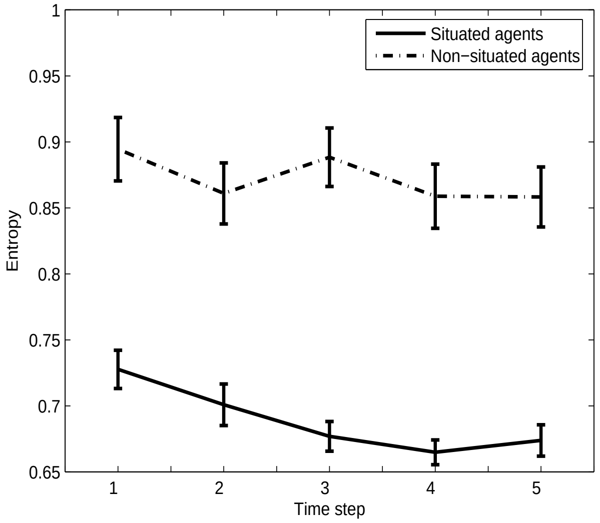

Fig. 10. Mean entropy over time and corresponding standard errors for all best evolved non-situated agents (dashed-dotted line) and situated agents (solid line).

5 Discussion

Our analysis and the results on the gender recognition and facial expression recognition tasks, lead us to expect that our results generalise to other image-classification tasks. However, further experiments and analysis are necessary to confirm this expectation.

Our results may be relevant to two research areas. First, the results may be relevant to the research area of computer vision. Most research on computer vision focuses on improving pre-processing (i.e., finding appropriate features) and on classification (i.e., mapping the features to an appropriate class) (Forsyth and Ponce, 2003). However, a few studies focus on a situated (or 'closed-loop') model (Köppen and Nickolay, 1996; Peng and Bhanu, 1998). Our study extends the application of a situated model using a local input window to the high-level task of gender recognition.

Second, our results are closely connected to research on human gaze control. Of course there is an enormous difference between a situated model and a real human subject. Nonetheless, there might be parallels between the gaze-control policies of the situated model and that of human subjects. We are aware of a few other studies that focus explicitly on the use of situated computational models in gaze control. In (Schlesinger and Parisi, 2001) an infant's gaze behaviour is studied with the help of a situated model. In (Sprague and Ballard, 2004), a situated model is trained for gaze control during visually guided sidewalk navigation. Both studies rely on simplified visual environments. Our model employs more realistic input, albeit static images instead of environmental dynamics.

6 Conclusion

Our results lead us to draw two conclusions as answers to the research questions posed in the introduction. First, we conclude that sensory-motor coordination contributes to the performance of situated models on the high-level task of artificial gaze control for gender recognition in natural images. Second, we conclude that the mechanism of sensory-motor coordination optimises classification performance by establishing useful dependencies between multiple actions and observations; the controller of a situated agent facilitates the task for its classifier by maximising task-specific information in the observations over time. In other words, the situated agent searches adequate classification areas in the image by determining fixation locations that depend on the presumed class and on specific image properties.

We envisage three directions of future research. First, as mentioned in Section 1, simulating gaze control in static images allows an investigation of principles of sensory-motor coordination in gaze control. However, simulated gaze control might abstract from important factors influencing an embodied system of gaze control, such as limitations to the possible motor actions. Therefore, we intend to verify the current results, using a real servo-motor system instead of a simulation. Secondly, we will focus on enhancing a situated model of gaze control with a state-of-the-art classifier and more features, and will compare the model with other classifiers on benchmark-tasks. Thirdly, we will compare the situated gaze control model with an active vision model that employs a belief state. An interesting element of such a comparison is that the situated model optimises its performance by establishing *dependencies* between multiple observations (Section 4.2), while active vision models with belief states base the updating of their belief state on the assumption that observations are independent.

Acknowledgements

This research is carried out within the ToKeN VindIT project (grant number 634.000.018) of the Netherlands Organisation for Scientific Research (NWO).

References

Beer, R. D., 2003. The dynamics of active categorical perception in an evolved model agent. Adaptive Behavior 11:4, 209–243.

Bruce, V., Young, A., 2000. In the eye of the beholder. Oxford University Press.

Calder, A. J., Burton, A. M., Miller, P., Young, A. W., Akamatsu, S., 2001. A principal component analysis of facial expressions. Vision research 41:9, 1179–1208.

Cohen, P., 1995. Empirical Methods for Artificial Intelligence. MIT Press, Cambridge, Massachusetts.

Floreano, D., Kato, T., Marocco, D., Sauser, E., 2004. Coevolution of active vision and feature selection. Biological Cybernetics 90:3, 218–228.

Forsyth, D. A., Ponce, J., 2003. Computer Vision: a Modern Approach. Prentice Hall, New Jersey.

Jain, A., Murthy, M., Flynn, P., 1999. Data clustering: A review. ACM Computing Surveys 31 (3).

Khare, V., Yao, X., Deb, K., April 2003. Performance scaling of multi-objective evolutionary algorithms. In: Fonseca, C. M., Fleming, P. J., Zitzler, E., Deb, K., Thiele, L. (Eds.), Evolutionary Multi-Criterion Optimization. Second International Conference, EMO 2003. Springer. Lecture Notes in Computer Science. Volume 2632, Faro, Portugal, pp. 376–390.

Köppen, M., Nickolay, B., 1996. Design of image exploring agent using genetic programming. In: Proc. IIZUKA'96. Iizuka, Japan, pp. 549–552.

Kröse, B. J. A., Bunschoten, R., 1999. Probabilistic localization by appearance models and active vision. In: IEEE Int. Conf. on Robotics and Automation. pp. 2255–2260.

Mitchell, T. M., 1997. Machine Learning. McGraw-Hill Companies, Inc.

Moghaddam, B., Yang, M. H., 2002. Learning gender with support faces. IEEE Trans. Pattern Analysis and Machine Intelligence 24:5, 707–711.

Nolfi, S., 2002. Power and the limits of reactive agents. Neurocomputing 42, 119–145.

Nolfi, S., Marocco, D., 2002. Evolving robots able to visually discriminate between objects with different size. International Journal of Robotics and Automation 17:4, 163–170.

O'Regan, J. K., Noë, A., 2001. A sensorimotor account of vision and visual consciousness. Behavioral and Brain Sciences 24:5, 883–917.

Peng, J., Bhanu, B., 1998. Closed-loop object recognition using reinforcement learning. IEEE Transactions on Pattern Analysis and Machine Intelligence 20:2, 139–154.

Pfeifer, R., Scheier, C., 1999. Understanding Intelligence. MIT Press, Cambridge, MA.

Schlesinger, M., Parisi, D., 2001. The agent-based approach: A new direction for computational models of development. Developmental review 21, 121–146.

Shannon, C., 1948. A mathematical theory of communication. The Bell System Technical Journal 27, 379–423, 623–656.

Sprague, N., Ballard, D., 2004. Eye movements for reward maximization. In: Thrun, S., Saul, L., Schölkopf, B. (Eds.), Advances in Neural Information Processing Systems 16. MIT Press, Cambridge, MA.

van Dartel, M. F., Sprinkhuizen-Kuyper, I. G., Postma, E. O., van den Herik, H. J., in press. Reactive agents and perceptual ambiguity. Adaptive Behavior.

Viola, P., Jones, M. J., 2001. Robust real-time object detection. Cambridge Research Laboratory, Technical Report Series.

Yao, X., 1999. Evolving artificial neural networks. Proceedings of the IEEE 87 (9), 1423 – 1447.

Zitzler, E., 2002. Evolutionary algorithms for mulitobjective optimisation. In: Giannakoglou, K., Tsahalis, D., Periaux, J., Papailiou, K., Fogarty, T. (Eds.), Evolutionary Methods for Design, Optimisation and Control. CIMNE, Barcelona, Spain.