GENERIC OBJECT CLASSIFICATION FOR AUTONOMOUS ROBOTS

Memòria del Projecte Fi de Carrera
d'Enginyeria en Informàtica
realitzat per
Raúl Pérez Trapilla..........................
i dirigit per
Sergio Escalera i Petia Radeva.........
Bellaterra,.29..de..gener....de 2007.

Abstract

One of the main problems of autonomous robots interaction is the scene knowledge. Recognition is concerned to deal with this problem and to allow robots to interact in uncontrolled environments. In this paper, we present a practical application for object fitting, normalization and classification of triangular and circular signs.
The system is introduced in the Aibo robot of Sony to increase the robot interaction behaviour. The presented methodology has been tested in real simulations and categorization problems, as the traffic signs classification, with very promising results.

Resumen

Uno de los principales problemas de la interacción de los robots autónomos es el conocimiento de la escena. El reconocimiento es fundamental para solventar este problema y permitir a los robots interactuar en un escenario no controlado. En este documento, presentamos una aplicación práctica de captura del objeto, normalización y clasificación de señales triangulares y circulares. El sistema es introducido en el robot Aibo de Sony para mejorar el comportamiento de la interacción del robot. La metodología presentada ha sido testeada en simulaciones y problemas de categorización reales, como es la clasificación de señales de tráfico, con resultados muy prometedores.

Resum

Un dels principals problemes de la interacció dels robots autònoms és el coneixement de l'escena. El reconeixement és fonamental per a solucionar aquest problema i permetre als robots interactuar en un escenari no controlat. En aquest document presentem una aplicació practica de la captura d’objectes, normalització i classificació de senyals triangulars i circulars. El sistema és introduït en el robot Aibo de Sony per a millorar la interacció del robot. La metodologia presentada ha estat testejada en simulacions i problemes de categorització reals, com és la classificació de senyals de transit, amb resultats molt prometedors.

Generic Object Classification for Autonomous Robots

Generic Object Classification for Autonomous Robots

Raúl Pérez Trapilla

Informatics, ETSE, UAB, Campus UAB, 08193 Bellaterra, Spain

Abstract

One of the main problems of autonomous robots interaction is the scene knowledge. Recognition is concerned to deal with this problem and to allow robots to interact in uncontrolled environments. In this paper, we present a practical application for object fitting, normalization, and classification of triangular and circular signs. The system is tested in the Aibo robot of Sony to increase the robot interaction behavior. The presented methodology has been tested in real simulations and categorization problems, as the traffic signs classification, with very promising results.

Key words: Object Recognition, Autonomous Robots, Model fitting, spatial normalization, multi-class Classification, Adaboost.

1 Introduction

Autonomous robots[1] are desired to perform tasks in unstructured environments without continuous human guidance. Any meaningful interaction with the environment will involve multiple related sentences that describe some complex ongoing events. To allow the robot to act in an intelligent and flexible way, it requires to recognize objects and learn the identity of unknown objects. In fact, there are an infinite number of scenes that contain the same object, which makes direct computation of scene geometry from a single image impossible. The use of external knowledge about the world and the current visual task reduces the number of plausible scene interpretations and may make the problem solvable. This approach is referred to as knowledge-based vision. Work in the area of knowledge-based vision incorporates methods from the field of AI in order to focus on the influence of context on scene understanding, the role of high level knowledge, and appropriate knowledge representations for visual tasks.

Computer vision is the technology concerned with computational understanding and use of the information present in visual images. In part, computer vision is analogous to the transformation of visual sensation into visual perception in biological vision. For this reasons the motivation, objectives, formulation, and methodology of computer vision frequently intersect with knowledge about their counterparts in biological vision. However, the goal of computer vision is primarily to enable engineering systems to model and manipulate the environment by using visual sensing. The machine's ability to monitor its environment, allowing it to adjust its actions based on what it has sensed, is a prerequisite for intelligence, such as Mars micro-rover shown in fig.1. The term intelligent machine is an anthropomorphism in that intelligence is defined by the criterion that the actions would appear intelligent if a person were to do it. A precise, unambiguous, and commonly held definition of intelligence does not exist.

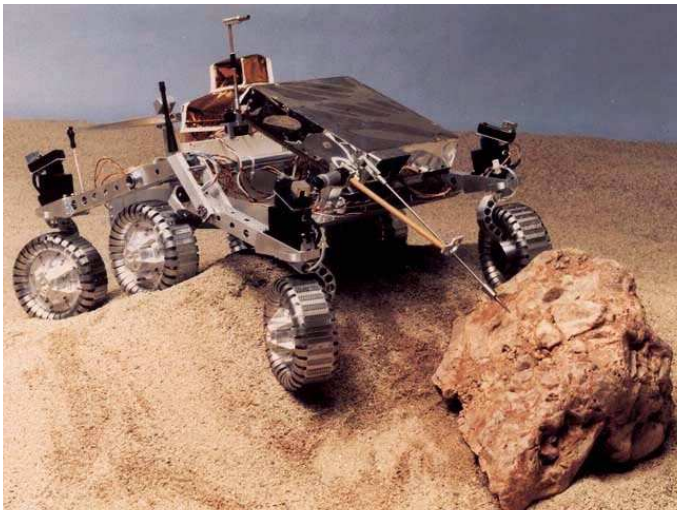


Fig. 1. Mars micro-rover mobile robot.

Since the physical embodiment of the machine or the particular task performed by the machine does not mark it as intelligent, the appearance of intelligence must come from the nature of the control or decision-making process that the machine performs. Given the centrality of control to any form of intelligent machine, intelligent control is the essence of an intelligent machine. A central objective of image interpretation is to recognize the scene contents. Recognition involves identifying an object based on a variety of criteria. It may involve identifying a certain object in the image as one seen before once an object is detect, categorization is required. Schemes for visual classification usually proceed in two stages. First, features are extracted from the image, and the object to be classified is represented using these features. Second, a classifier is applied to the measured features to reach a decision regarding the represented class. Powerful methods have been developed for performing visual classification. Some of the most used are K-Nearest Neighbors [7], Tangent Distance [9], Fisher Linear Discriminant Analysis [6] or Principal Component Analysis

[8]. Nowadays, Support Vector Machine [10] and Adaboost [11] are the most frequently used.

In this paper, we deal with the multi-class classification task applied to autonomous robots. Robotics deals with the practical application of many artificial intelligence techniques to solve real world problems. This combines problems of sensing and modeling the world, planning and performing tasks, and interacting with the world. The Aibo robot from Sony is a perfect tool to implement and test artificial intelligent techniques in robotics. It establishes communication with people by displaying emotions, and assumes various behaviors based on information which it gathers from its environment. In this way, we used this tool to test the present system. In our application, we use the results of the Adaboost procedure as a detection algorithm. The use of this algorithm let us to detect regions of interest with high probability of containing signs. Once the Adaboost returns a ROI [15], we fit the model using the hough transform (triangular sign) or fast radial symmetry (circular sign). Once we have fit the object, we compare a wide set of the state-of-the-art classification strategies in order to obtain the label of the present object in the scene. The system is emulated in the Sony Aibo robot, showing high performance on classifying different objects in real-time.We also tested our system in a real categorization problem: real traffic sign classification. Besides, we presented this work in different scientific exhibitions.

This paper is organized as follows: Section 2 explain the Aibo robot tool. Section 3 overview the classification technique applied in this paper. Section 4 presents our system. Section 5 shows the experiments and results, and section 6 concludes the paper.

2 Aibo Robot

Robots are growing in complexity and their use in industry is becoming more widespread. The main use of robots has so far been in the automation of mass production industries, where the same, definable tasks must be performed repeatedly in exactly the same fashion. Industrial robots can be manufactured in a wide range of sizes and so can handle more tasks requiring heavy lifting than a human could. Car production is the primary example of the employment of large and complex robots for producing goods.

Robots are also useful in environments which are unpleasant or dangerous for humans to work in, for example bomb disposal, work in space (eg. Canadarm2) or underwater, in mining, and for the cleaning of toxic waste. Robots are also used for patrolling these toxic areas, robots equipped for this job are e.g. the Robowatch OFRO[2], and Robowatch MOSRO.Automated Guided

Vehicles (AGVs) are movable robots that are used in large facilities such as warehouses hospitals and container ports, for the movement of goods, or even for safety and security patrols. Such vehicles follow wires, markers or laser-guidance to navigate around the location and can be programmed to move between places to deliver goods or patrol a certain area. One robot being used in the United States is the Tug robot by Aethon Inc[3], an automated delivery system for hospitals. This robot travels around hospitals to deliver medical supplies, medication, food trays, or just about anything to nursing stations. Once it is finished it goes back to its charging station and waits for its next task. Domestic robots are now available that perform simple tasks such as vacuum cleaning and grass cutting. Nowadays domestic robots have the aim of providing companionship (social robots) or play partners (ludobots) to people. In this scope we find the Aibo robot.

AIBO (Artificial Intelligence roBOt[4], also means "love" or "attachment" in Japanese) is one of several types of robotic pets designed and manufactured by Sony; there have been several different models since their introduction in 1999. Able to walk, "see" its environment via camera, and recognize spoken commands, they are considered to be autonomous robots, since they are able to learn and mature based on external stimuli from their owner or environment, or from other AIBOs. The AIBO has seen used as an inexpensive platform for artificial intelligence research, because it integrates a computer, vision system, and articulators in a package vastly cheaper than conventional research robots.

Fig. 2. Aibo relationship.

The AIBO robot establishes communication with people by displaying emotions, and assumes various behaviors (autonomous actions) based on information which it gathers from its environment. The AIBO robot is not only a robot, but an autonomous robot with the ability to complement your life. While living with you, the AIBO robot's behavioral patterns will develop as it learns and grows.Emotions and instincts form the basis for the AIBO robot's autonomous behavior. Based on all sorts of factors which it picks up from its surroundings, the AIBO robot undergoes changes in spirit that display themselves in the form of emotional expression.

3 Multi-class Classifiers

To deal with the multi-class categorization problem, we perform a wide comparative among the state-of-the-art multi-class classifiers: K-Nearest Neighbors [7], Fisher Linear Discriminant Analysis [6], Support Vector Machines [10], and Adaboost [11].

Table 1 shows the formulation for the different multi-class classifiers and the rest of the paper techniques:

Var	Meaning
N	number of samples
x	vector features test
n	dimension
i, j, m, t	index
M	number of runs
q	distance
y	Training samples
C	number of class
c_i	class with index i
z	new projected samples
H	number of ranges
g	gradient
ve	affected pixel
X, Y	components
θ	angle
λ	set of intersection points

Var	Meaning
W	projection matrix
S_b	Scatter matrix between-class
S_w	Scatter matrix intra-class
s	index
T	Transpose
l	label
d, r, γ	Kernel parameters
ξ, b	SVM optimization parameters
w	eigen vectors
ϕ	Kernel function
h	range of distance
p	point
O	orientation matrix
a	pendent of the line
φ	constant

Table 1
Paper formulation

K-Nearest Neighbors

Among the various methods of supervised statistical pattern recognition, the Nearest Neighbor is the most traditional one, it does not consider a priori assumptions about the distributions from which the training examples are drawn. It involves a training set of both positive and negative cases. A new sample is classified by calculating the distance to the nearest training case; the sign of that point then determines the classification of the sample. The $k-NN$ classifier extends this idea by taking the k nearest points and assigning the sign of the majority. It is common to select k small and odd to break ties (typically 1, 3 or 5). Larger k values help reduce the effects of noisy points within the training data set, and the choice of k is often performed through cross-validation. In this way, given a input test sample vector of features x of dimension n, we estimate its Euclidean distance d (eq.1) with all the training samples (y) and classify to the class of the minimal distance.

$$q(x, y) = \sqrt{\sum_{j=1}^{n}(x_j - y_j)^2} \tag{1}$$

Fisher Linear Discriminant Analysis

Given the binary classification problem, Fisher projects at one dimension each pair of classes (reducing to $C - 1$ where C is the number of classes), multiplying each sample by its projection matrix, which minimize the distance between samples of the same class, and maximizes the distance between the two classes.The result is shown in fig. 3, where the blue and red points belong to the samples of the two projected classes, and the green line indicates the threshold that best separates them .

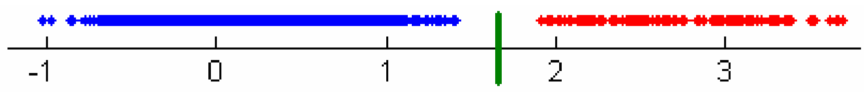


Fig. 3. Fisher projection for two classes and threshold value.

The algorithm is:

Given the set of N column vectors $\{\overrightarrow{y_i}\}$ of dimension n, we calculate the mean of the data. For C classes $\{c_1, c_2, , c_C\}$, the mean of the class c_i that contains N_i elements is:

$$\overrightarrow{\mu}_{yi} = \frac{1}{N_i} \sum_{\overrightarrow{y}_j \varepsilon c_i} \overrightarrow{y}_i \tag{2}$$

The separability maximization between classes will be defined as the quotient between the scatter matrix between-class:

$$S_b = \sum_{i=1}^{I} N_i(\overrightarrow{\mu}_{yi} - \overrightarrow{\mu}_y)(\overrightarrow{\mu}_{yi} - \overrightarrow{\mu}_y)^T \tag{3}$$

and the scatter matrix intra-class:

$$S_w = \sum_{i=1}^{I} \sum_{\overrightarrow{y}_j \varepsilon c_i} (\overrightarrow{y}_j - \overrightarrow{\mu}_{yi})(\overrightarrow{y}_j - \overrightarrow{\mu}_{yi})^T \tag{4}$$

obtaining a projection that define an optimal discriminant features.

The projection matrix W maximizes:

$$\frac{det(W^T S_B W)}{det(W^T S_W W)} \tag{5}$$

Let$\{\overrightarrow{w}_1, \overrightarrow{w}_s, ..., \overrightarrow{w}_n\}$ be the generalized eigenvectors of SB and SW. Then, selecting the $d < n$ that corresponds to the highest eigenvalue, we have the projection matrix $W = [\overrightarrow{W}_1, \overrightarrow{W}_s, ..., \overrightarrow{W}_n]$, project the samples to the new space by using:

$\overrightarrow{z} = W_d^T \overrightarrow{y}$

The generalized eigenvectors of eq.(5) are the eigenvectors of $S_B S_W^{-1}$

Support Vector Machines

The goal of SVM is to produce a model which predicts target value of data instances in the testing set which are given the attributes . Given a training set of instance-label pairs (y_i, l), where $y_i \varepsilon R^n$ and $l \varepsilon \{1, -1\}$, the support vector machines require the solution of the following optimization problem:

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{i=1}^{I} \xi_i \tag{6}$$

Subject to

$$l(w^T \phi(y_i) + b) \geq 1 - \xi_i, \xi_i \geq 0 \tag{7}$$

Here training vectors y_i are mapped into a higher (maybe infinite) dimensional space by the function ϕ . Then SVM finds a linear separating hyperplane with the maximal margin in this higher dimensional space. $c > 0$ is the penalty parameter of the error term. We can define, $K(y_i, y_j \equiv \phi(y_i)^T\phi(y_j))$ called the kernel function. Though new kernels are being proposed by researchers, the most common four basic kernels are:

Linear: $K(y_i, y_j) = y_i^T y_j$

Polynomial: $K(y_i, y_j) = (\gamma y_i^T y_j + r)^d, \gamma > 0$

Radial basis function (RBF): $K(y_i, y_j) = \exp(-\gamma \parallel y_i - y_j \parallel^2), \gamma > 0$

Sigmoid: $K(y_i, y_j) = \tanh(\gamma y_i^T y_j + r)$

Here γ, r, and d are kernel parameters.

Adaboost

The AdaBoost boosting algorithm has become over the last few years a very popular algorithm to use in practice. The main idea of AdaBoost is to assign each example of the given training set a weight. At the beginning all weights are equal, but in every round the weak learner returns a hypothesis, and the weights of all examples classified wrong by that hypothesis are increased. That way the weak learner is forced to focus on the difficult examples of the training set. The final hypothesis is a combination of the hypotheses of all rounds, namely a weighted majority vote, where hypotheses with lower classification error have higher weight. Summarizing, the approach consists of a) choosing a (weak) classifier, b) modifying example weights in order to give priority to examples where the previous classifiers fail, and c) combining classifiers in a multiple classifier. The combined classifier allows a good generalization performance with the only requirement that each weak learner obtains an accuracy better than random. The Adaboost procedure has been used for feature selection, detection, and classification problems. In our problem, the Gentle Adaboost has been previously applied to detect the regions of interest (ROI) with high probability of containing signs from the Aibo video data[15]. In fig. 4, the Gentle Adaboost algorithm used in the first step of this system[15] is shown.

In fig.4, the weights of each of the samples of the training set are initialized. Normally, the same weight is assigned to each sample satisfying $\sum_{i=1}^{N} f_i = 1$. At iteration m of the algorithm, a weak classifier evaluates the feature space and selects the best feature based on the weights of the samples. The samples

Gentle AdaBoost

1. Given N examples $(x_1, y_1), \ldots, (x_N, y_N)$ with $x \in \Re^k, y_i \in \{-1, 1\}$
2. Start with weights $w_i = 1/N$, $i = 1, \ldots, N$.
3. Repeat for $m = 1, \ldots, M$

   (a) Fit the regression function $f_m(x)$ by weighted least-squares of y_i to x_i with weights w_i

   (c) Set $w_i \leftarrow w_i \cdot \exp(-y_i \cdot f_m(x_i))$, $i = 1, \ldots, N$, and renormalize weights so that $\sum_i w_i = 1$.
4. Output the classifier $sign\left[\sum_{m=1}^{M} f_m(x)\right]$

Fig. 4. Gentle Adaboost algorithm

are re-weighted with a exponential loss function, and the process is repeated M times or when the training classification error is zero. The final strong classifier of the Gentle Adaboost algorithm[12] is an additive model that use a threshold as a final classifier. To classify a new input, the results of applying the m weak classifiers with the test sample are added or subtracted depending on the accuracy of each weak classifier. In the common case of using decision stumps as a weak classifier, the additive model assigns the same weight to each of the hypothesis, so all the features are considered to have the same importance. The last fact is the main difference between the Gentle Adaboost and the traditional Adaboost versions.

4 System

This section explains the details of our system scheme shown in fig. 5, focusing on the relationship between each of the methods explained above and their integration in a real time recognition system. The system is composed by three main stages: object detection, model fitting and normalization, and classification.

4.1 Object Detection

Detected region are provided by the detection process based on an attentional cascade of boosting classifiers applying the Haar-like featuring estimated over the integral image. Given an Adaboost positive sample, it determines a region of interest (ROI) that contains an object [15] (training detector and object detection steps of fig.5). However, besides the ROI we miss information about scale and position, so before applying recognition we need to apply a spatial normalization. Concerned with the correlation of sign distortion, we look for affine transformations that can perform the spatial normalization to improve final recognition.

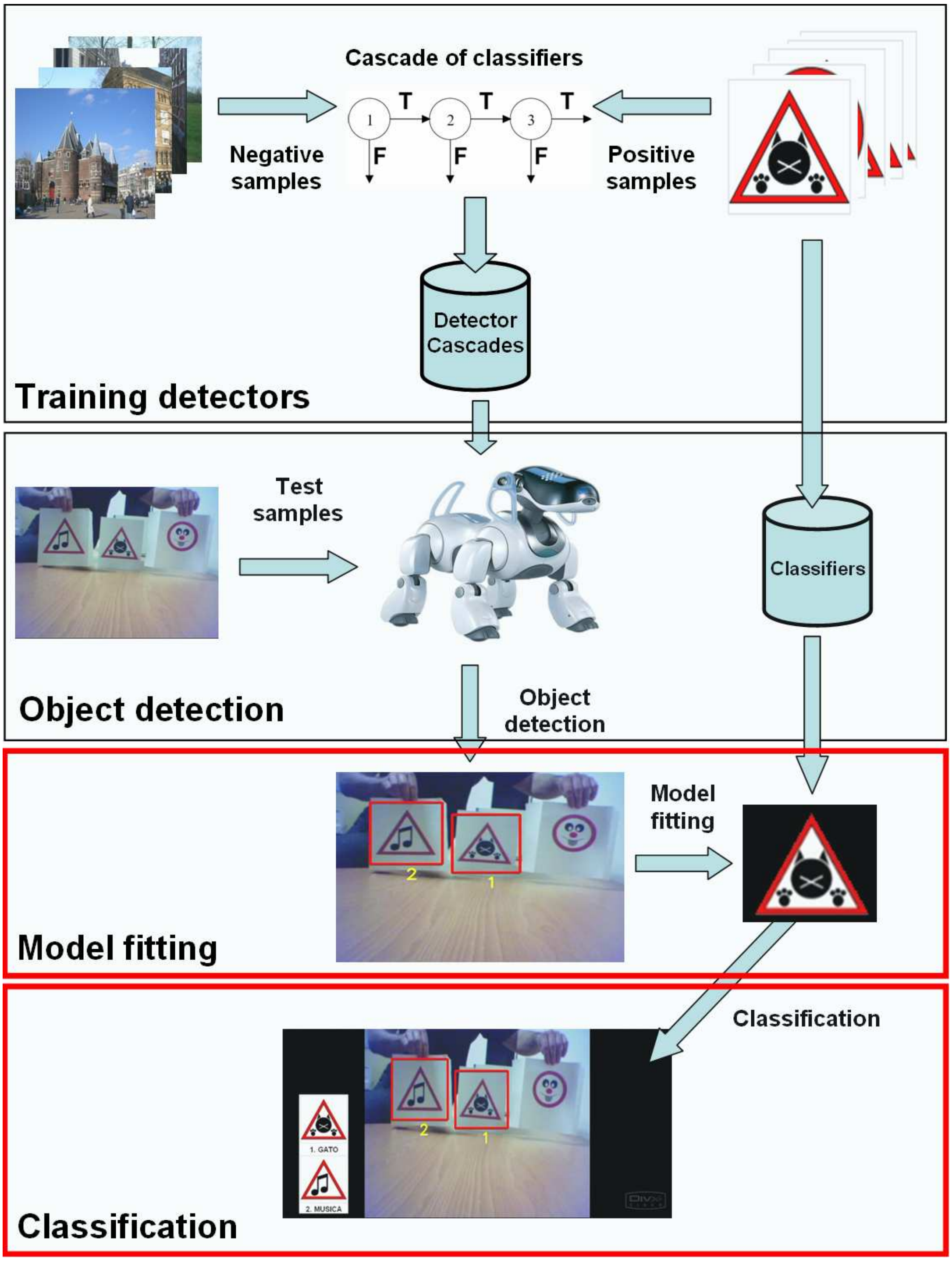


Fig. 5. Object detection and classification system of the Aibo robot

To deal with the Model fitting step of fig.5,we apply the fast radial symmetry to fit circular signs, that offers great robustness against the noise [5]. On the other hand, for the case of triangular signs, the method that allows a successful model fitting is based on the Hough transform [13].

Fast radial symmetry

In order to capture the model contained in the detected ROI, we consider the radial properties of the circular signs to fit a possible instance and to estimate its center and radius.The fast radial symmetry is calculated over a set of one or more ranges H depending on the scale of the features one is trying to detect. The value of the transform at range indicates the contribution to radial symmetry of the gradients at a distance h away from each point. At each range h, we examine the gradient g at each point p, from which a corresponding positively-affected pixel $p_{+ve}(p)$ and negatively-affected pixel $p_{-ve}(p)$ are determined and accumulated in the orientation projection image O_n:

$$a) \quad P_{+ve}(p) = p + round\frac{g(p)}{||g(p)||}h, \qquad P_{-ve}(p) = p - round\frac{g(p)}{||g(p)||}h \quad (8)$$

$$b) \quad O_h(P_{+ve}(p)) = O_h(P_{+ve}(p)) + 1, \qquad O_h(P_{-ve}(p)) = O_h(P_{-ve}(p)) + 1 (9)$$

Now, to locate the radial symmetry position, we search for the position (X, Y) of maximal value at accumulated orientations matrix $O^T = \sum_{i=1}^{h} O_h$. Locating that maximum we determine the radius length. This procedure allows to obtain robust results for circular traffic signs fitting. An example is shown in fig. 6, where the X and Y gradient components, gradient module, orientations matrix, and estimated center and radius are shown.

(a) (b) (c) (d) (e) (f)

Fig. 6. (a) Input image, (b) X-derived, (c) Y-derived, (d) image gradient g, (e) total orientations accumulator matrix O^T, (f) Captured center and radius.

Hough transform

The Hough transform is a technique which can be used to isolate features of a particular shape within an image. Because it requires that the desired features be specified in some parametric form, the generalized Hough transform can be employed in applications where a simple analytic description of a features is not possible.The underlying principle of the Hough transform is that there are an infinite number of potential lines that pass through any point, each at a different orientation. The purpose of the transform is to determine which of these theoretical lines pass through most features in an image - that is, which lines fit most closely to the data in the image. In order to determine that two points lie on the same potential line, it is necessary to create a representation of a line that allows meaningful comparison in this context. In the standard Hough transform, each line is represented by two parameters, commonly called r and θ (theta), which represent the length and angle from the origin of a normal to the line in question fig. 7. Using this parametrization, an equation of the line can be written as:

$$q = X\cos\theta + Y\sin\theta \tag{10}$$

$$Y_1 = round(\frac{q - X_1\cos\theta}{\sin\theta}) \tag{11}$$

$$Y_2 = round(\frac{q - Y_2\cos\theta}{\sin\theta}) \tag{12}$$

$$a = \frac{Y_2 - Y_1}{X_2 - X_1} \tag{13}$$

$$b = Y_1 - aX_1 \tag{14}$$

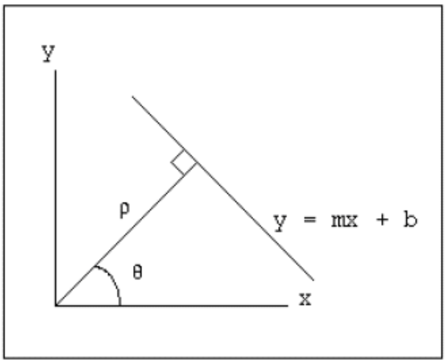
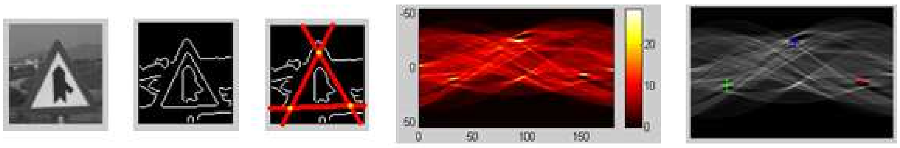


Fig. 7. Correspondence to Hough

Given an input image (fig. 8)(a) we used Canny detector to obtain the contours map (fig. 8(b)) before of applying the Hough transform..

Given a region of interest that contains a sign, we know a priori that the three possible angles of each side of the triangle and an error margin. With this information, we searched at three possible angles range for the three three representative lines at Hough space fig. 8(e). Once we had the three lines, we only needed to calculate its intersection to find the three corners of the triangle fig. 8(c). With the three corners of the triangle we can transform the image to correct the affine transformations to proceed with the classification procedure.



(a) (b) (c) (d) (e)

Fig. 8. (a) Input image, (b) Canny contours, (c) detected triangle lines and intersections, (d) Hough space, (e) three detected lines for three known angles and margin error,

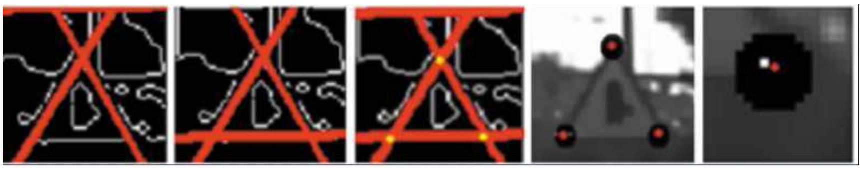
Nevertheless,before final transformation, we need to consider additional restrictions to obtain the three representative border lines of a triangular traffic sign. Each line has associated a position in relation to the others. In fig. 9(a) a false horizontal detected sign due to the background confusion is shown. As this line does not have the expected spatial restrictions of the object, we iterate Hough to detect the following representative line considering its range of degrees. The corrected image is shown in fig. 9(b). Once we have the three detected lines we calculate their intersection, as shown in fig. 9(c). Given the parameters a and φ that define the equation $y = a \times x + \varphi$ for each of the three lines, the intersection point (X, Y) for each pair of lines is defined as follows:

$$X_t = (\varphi_2^i - \varphi_1^i)/(a_1^i - a_2^i), \quad Y_t = a_1^i X_t + \varphi_1^i \mid t, i \in [1, ..., 3] \tag{15}$$

To assure that the lines are the expected ones, we complement the procedure searching for a corner at a circular region at each intersection surroundings (as shown in fig. 9(d) and (e)):

$$\lambda = \{(X_i, Y_i) \mid \exists p < ((X - X_i)^2 + (Y - Y_i)^2 - d^2)\} \mid i \in [1, ..., 3] \tag{16}$$

where λ is the set of valid intersection points, and p corresponds to a corner point to be located in a neighborhood of the intersection point.



(a) (b) (c) (d) (e)

Fig. 9. (a) Three detected lines, (b) Corrected horizontal line, (c) Lines intersections, (d) Corner checking, (e) Corner found.

Normalization:

The analysis of sign images is a difficult problem given the low quality of the images, To cope with this problem, we need elaborate and study different spatial normalization. Once the sign model is fitted using the commented methods, the next procedure is the spatial normalization of the shape before classification. The steps are: transform the image to make the recognition invariant to small affine deformations, rescale to the signs database size, filter with Weickert [14] anisotropic filter, and mask the image to exclude background at the classification step. To prevent the effects of illumination changes, the histogram equalization improves image contrast and yields a uniform histogram.

4.3 Classification

Once we have the image normalized, a set of classification strategies are applied to deal with the multi-class sign categorization in order to obtain the label of the object.

5 Results

To validate the methods of our system, we apply a set of experiments: we simulate the system in the Aibo robot, we test the system in a real traffic signs and classification problem, and finally, we show different scientific exhibitions of the present work. In the following section, we show the parameters that optimize the performance of our system.

5.1 *Work Parameters*

5.1.1 *Model fitting*

The Hough transform applied is the probabilistic hough transform because it is more efficient in case of few long linear segments. The Distance resolution in our system is 1 pixel-related units, and the angle resolution measured is $\frac{\pi}{180}$ radians. Others parameters are : 10 accumulated pixel orientations for threshold, 2 pixels for the maximum gap between line segments lieing on the same line and, finally, 3 pixels for minimum line length. For the fast radial symmetry, we apply a range of possible radius between 12 and 32 pixels, and a threshold of 10% of the maxim gradient magnitude to compute the orientations.

5.1.2 *Classification*

The parameter used for the classification strategies are: 3 neighbors for $K-NN$, 99% or a previous PCA and 3 neighbors for the FLDA strategy, Radial Basis function SVM with the gamma parameter set to 1, and 40 runs of Gentle Adaboost with decision stumps. $K-NN$ algorithm is applied directly in the multi-class case, the rest of classification strategies are binary (they distinguish against just two classes). To extend the categorization strategies to the multi-class case, we used a voting scheme. The voting scheme consist on training each pair of possible classes ($C(C-1)/2$ pair of classes for C classes), and finally classify by the class with highest number of votes. The estimation of the accuracy is obtained using stratified ten-fold cross-validation at 95% of the confidence interval.

5.2 *Aibo Experiments*

For this experiment, we used a set of 500 triangular and 500 circular signs obtained from the method of [15]. The triangular and circular signs are divided in the classes of fig.13. Some real detected regions from the 416×320 pixels Aibo resolution are shown in fig.12

We can observe the good results of the vector classification in fig.10 and 11 for circular and triangular signs, respectively. The results in both cases are quite similar. In the same figures, Gentle Adaboost also obtain good results, and the last positions are for FLDA and KNN, respectively.

An example of the detection and classification of triangular signs is shown in fig.14, with an experimental interface designed for our application in [15].

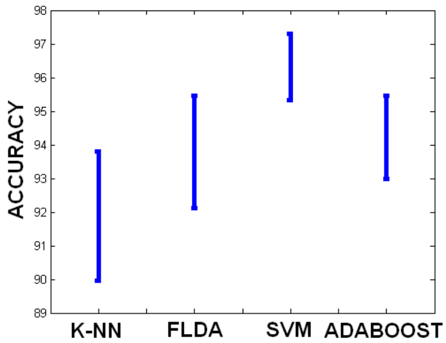


Fig. 10. Multi-class classifiers Accuracy for circular signs

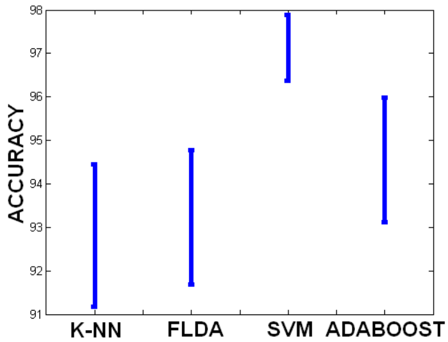


Fig. 11. Multi-class classifiers Accuracy for tringular signs

Fig. 12. detected regions from Aibo resolution

Fig. 13. Aibo experiment classes

5.3 Traffic signs classification

For the problem of real traffic sign classification, we use regions from [16] that contain a wide set of traffic signs. In particular, we classify the 12 triangular

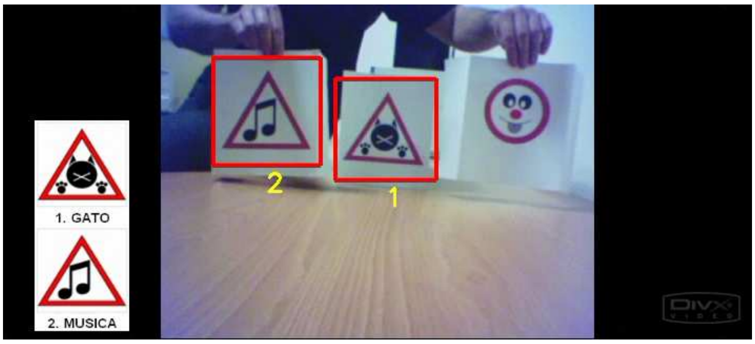


Fig. 14. Experimental interface with detection and classification of triangular signs

classes of fig.19, and the 18 circular classes of fig.17 and 18. The classification results are shown in figures fig.15 and 16 for circular and triangular classes, respectively. In this case, the ranking of the classifiers is similar compared to the Aibo results, and we also obtain high results for the triangular group applying our fitting and normalization strategies. Nevertheless, the circular results are some inferior in this case. It is a normal behavior since the speed group has very similar classes, and the resolution of the considered regions (at least 24×24 pixels) makes the classification ambiguous in some cases.

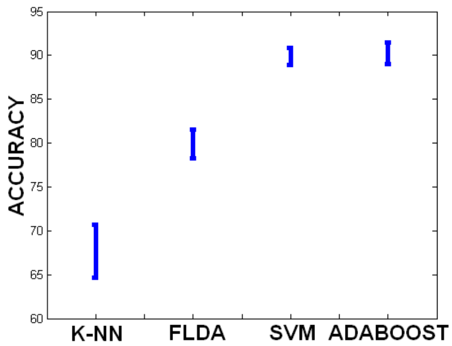


Fig. 15. Multi-class classifiers Accuracy for circular traffic signs

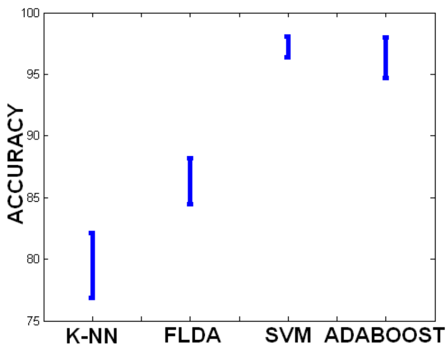


Fig. 16. Multi-class classifiers Accuracy for triangular traffic signs

In the figures 17,18,19 we can observe as all classes except classes of speed signs are quite different.

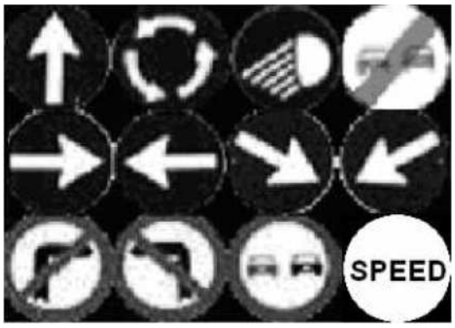


Fig. 17. Circular traffic classes

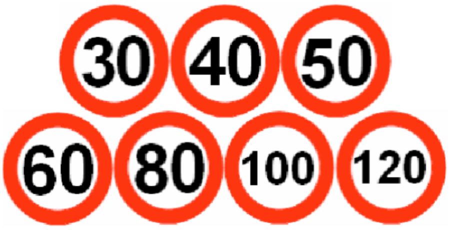


Fig. 18. Speed traffic classes

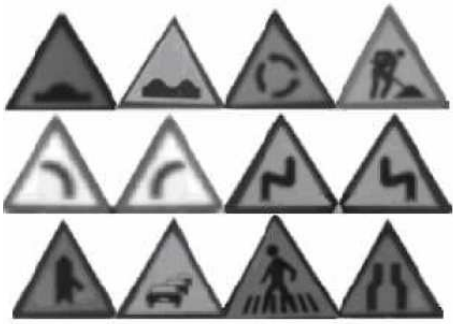


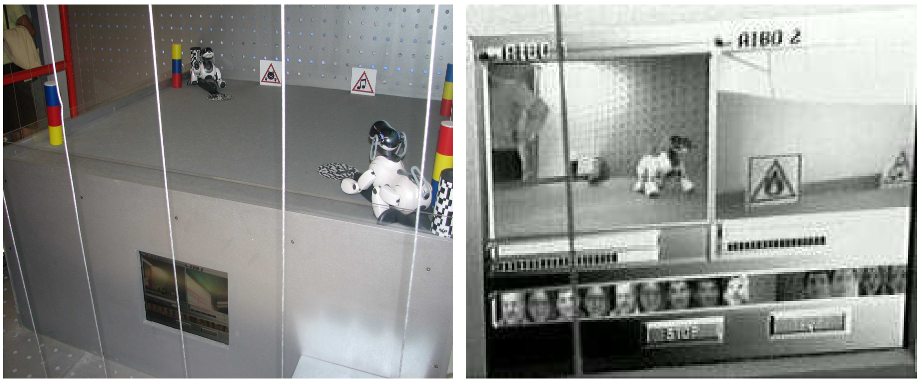
Fig. 19. Triangular traffic classes

5.4 Discussion

In this chapter, we comment different exhibitions of the present system on different social events.

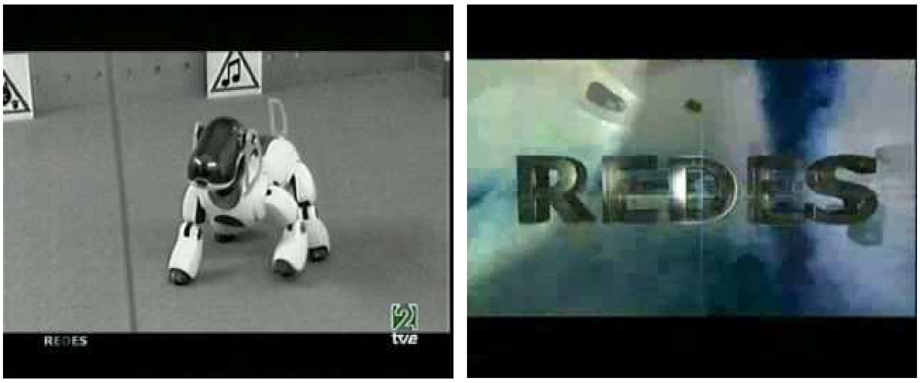
5.4.1 "Apropa't a la cincia"

Robotics is a focus of attention for a high number of scientists. Now, 50 years after the birth of the Artificial Intelligence, we presented, during the year 2006/2007 in the "Apropa't a la Cincia" event organized by the Generalitat de Catalunya, a simulation of the Aibo robot using our system. Different illustrations from the event are shown in figures fig. 20. For more details of the event see Appendix X or enter in http://www10.gencat.net/probert/catala/exposicio/ex14_ciencia.htm



(a) Aibo environment (b) Aibo interface

Fig. 20. Pictures of the exhibition "Apropa't a la cincia(2006/2007")



(a) Program frame (b) Documental logo

Fig. 21. Redes TV program "Programming emotions"(7.1.2007)

5.4.2 "Redes"

Besides, our work was emitted as a part of the documental "REDES" from TV2 with the little "programing emotions", in date of 7.1.2007. Different images from the show are shown in figure fig.21. For more details enter in http://www.rtve.es/tve/b/redes/semanal/prg418/index.html

6 Conclusions

We presented a multi-class classification system for triangular and circular signs that allow to autonomous robots to interact with its environment. The strategy fits the model and normalize the image region contain. Besides, a set of state-of-the-art classification strategic has been tested to obtain robust results. The presented methodology has been introduced in the Aibo robot of Sony and tested in a real traffic signs problem,with great success.

7 Acknowledges

I want to acknowledge Sergio Escalera and Petia Radeva by their direction in this project. To my companions Carlos and Pepelu by its contributions, and the inestimable aid of Xavi. Also I want to acknowledge my family and Miriam, by the support, helps, and sometimes by their patience.Thank you!!!

References

[1] George A. Bekey, "Autonomous Robots: From Biological Inspiration to Implementation and Control (Intelligent Robotics and Autonomous Agents)" 2005

[2] "http://www.adx2.com/content/category/8/22/59/", "OFRO +detect" Digital 2007

[3] "http://www.aethon.com/inthenews/archive.html", "The Tug in the News" Digital 2007

[4] "support.sony-europe.com/aibo/downloads/", "Sony Aibo Robot" Digital 2005

[5] G. Loy, and A. Zelinsky, "A Fast Radial Symmetry Transform for Detecting Points of Interest", Transactions on PAMI, 2003.

[6] MaxWelling , "Fisher Linear Discriminant Analysis", Department of Computer Science, University of Toronto, 2002.

[7] Wen-Jyi Hwang, Kuo-Wei Wen, "Fast kNN classification algorithm based on partial distance search", Chung Yuan Christian Univ 1998.

[8] Ian T. Jolliffe, "Principal Component Analysis", 2002.

[9] Patrice Y. Simard, Yann A. Le Cun, John S. Denker, Bernard Victorri, "Transformation Invariance in Pattern Recognition Tangent Distance and Tangent Propagation" ,Image Processing Services Research Lab, ATT Labs USA

[10] Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin, "A Practical Guide to Support Vector Classification", Department of Computer Science and Information Engineering, National Taiwan University, 2002.

[11] J. Friedman, T. Hastie, and R. Tibshirani, Additive logistic regression: a statistical view of boosting," Technical Report, 1998.

[12] S. Escalera, O. Pujol, and P. Radeva, "Boosted Landmarks of contextual descriptors and Forest-ECOC: A novel framework to detect and classify objects in cluttered scenes", International Conference on Pattern Recognition, Hong Kong, 2006.

[13] B. S. Morse, Segmentation (edge based, hough transform)," technical report, 2000.

[14] Weickert J., "Anisotropic diffusion in image processing", European Consortium for Mathematics in Industry (B.G. Teubner,Stuttgart), 1998.

[15] C. Gallardo, "Generic Object Detection for Autonomous Robots ," technical report UAB, 2007.

[16] url:"http://www.icc.es/" "Institut Cartogrfic de Catalunya"

[17] url:"http://www.seudigital.cat/ciencia/" "Barcelona ciutat de la Ciència"

[18] url:"http://www10.gencat.net/probert/catala/exposicio/ex14_ciencia.htm" "Apropa't a la ciència. De la Recerca a la Innovació"

8 Appendices

Appendices To complement the information of the project that can not be included in the article, some appendices have been included. In order to explain in more detail the domain of the work.

8.1 The AIBO Entertainment Robot ERS-7M3 parts

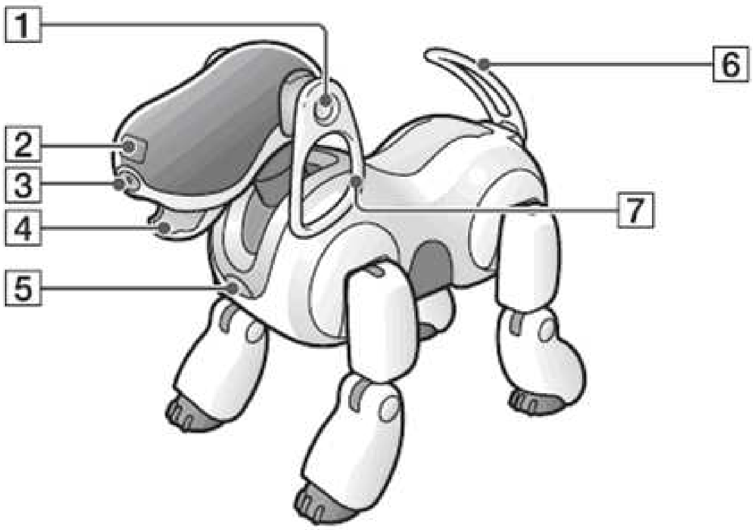


Fig. 22. Front view of Aibo Robot

[1] Stereo microphones Allow the AIBO Entertainment Robot to listen to the surrounding environment.

[2] Head distance sensor Measures the distance between the AIBO robot and other objects.

[3] Color camera Detects the color, shape, and movement of nearby objects.

[4] Mouth Picks up the AIBOne toy and expresses emotions.

[5] Chest distance sensor Measures the distance between the AIBO robot and other objects.

[6]Tail Moves up, down, left, and right to express the AIBO robot's emotions.

[7] Ears Indicates the AIBO robot's emotions and condition.

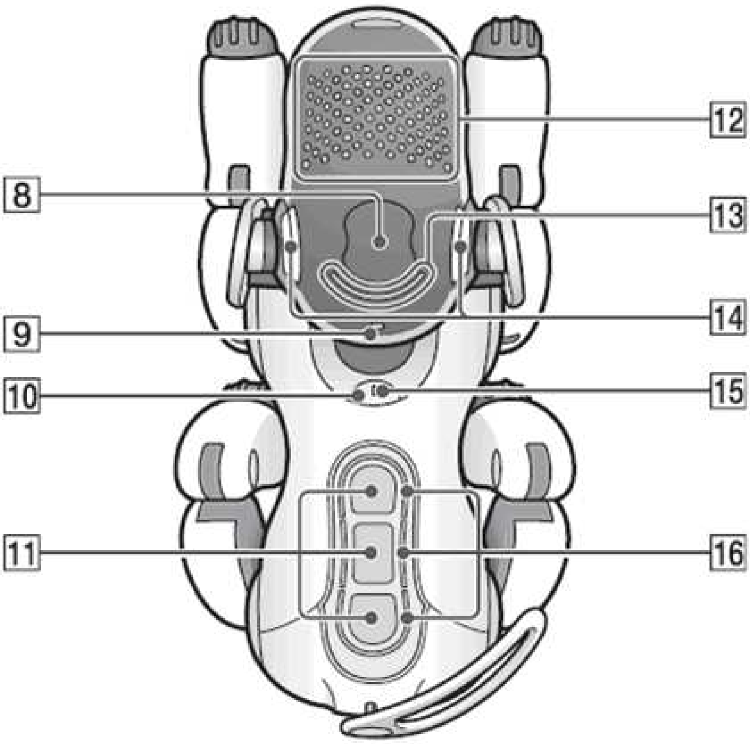


Fig. 23. Top view of Aibo Robot.

[8] Head sensor Detects and turns white when you gently stroke the AIBO robot's head.

[9] Wireless light (on the back of the AIBO robot's head) Indicator used with the wireless LAN function. This light turns blue when the AIBO robot is connected to the e-mail server.

[10] Pause button When pressed, the AIBO robot's activity will pause or resume.

[11] Back sensors (front, middle, and rear) Detect and turn white when you gently stroke the AIBO robot's back.

[12] Face lights (illuminated face) These lights turn various colors to show the AIBO robot's emotions and conditions.

[13] Head light Detects and turns white when you touch the head sensor. Lights / flashes orange when one of the AIBO robot's joints is jammed .

[14] Mode indicators (inner side of ears) These indicate the present mode and condition of the AIBO robot .

[15] Operation light During operation: turns green. During preparation for shutdown: flashes green. During charging: turns orange. When a charging error occurs: flashes orange. When operation stops: turns OFF. Outside hours of activity (Sleeping on the Energy Station): slowly flashing green.

[16] Back lights (front, middle, and rear) Detect and turn white when you gently touch the AIBO robot's back sensors. These lights also turn blue (front), orange (middle), and red (rear) to indicate a variety of actions.

This shows the AIBO robot with its stomach compartment cover off.24

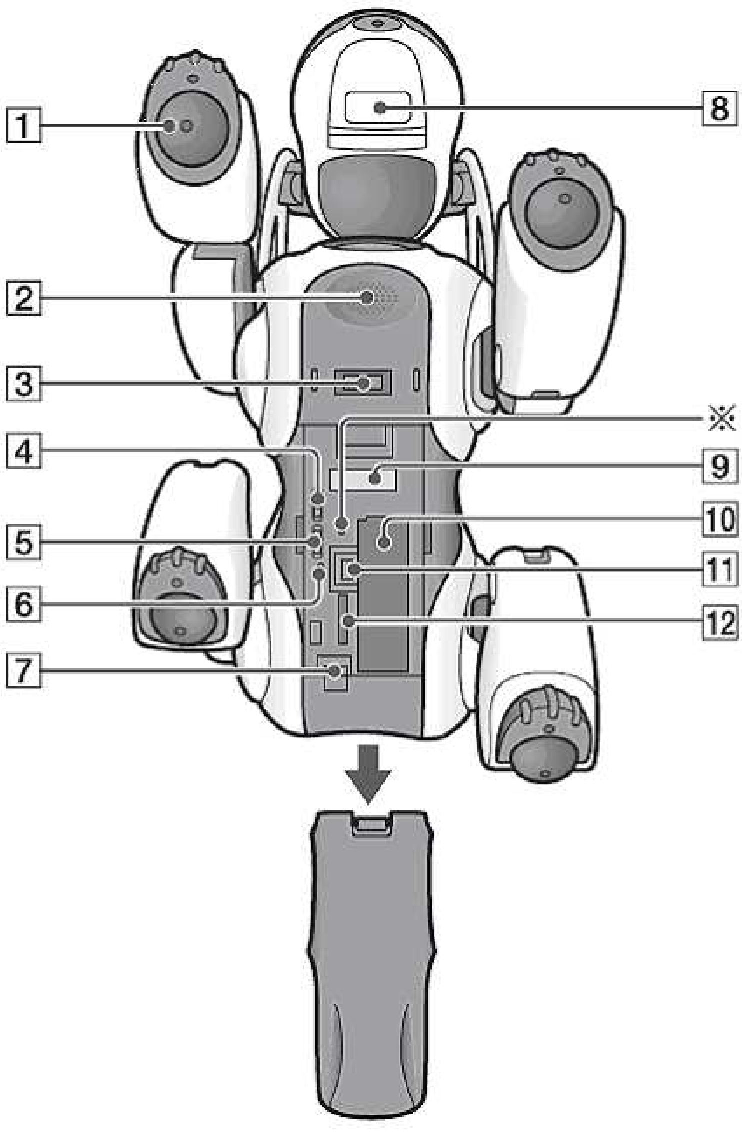


Fig. 24. Bottom view of Aibo Robot.

[1] Paw sensors These are located on the bottom of the AIBO Entertainment Robot's paws, and detect contact with any surface it touches. When the AIBO

robot extends one of its paws, it will react with happiness if you touch it.

[2] Speaker Emits music, sound effects, and voice guide.

[3] Charging terminal When you place the AIBO robot on the Energy Station, this part makes contact with the station to allow charging of the AIBO robot's battery.

[4] Volume control switch (VOLUME) Adjusts the volume of the speaker to one of four levels (including no sound).

[5] Wireless LAN switch (WIRELESS) This turns the AIBO robot's wireless LAN function ON or OFF.

[6] "Memory Stick" media access indicator This indicator turns red while the AIBO robot is reading or writing to a "Memory Stick" media. While the indicator is ON, you cannot remove the "Memory Stick" media or battery by means of the "Memory Stick" media eject button (Z) or the battery latch (Z). Under this circumstance, never attempt to forcibly remove the "Memory Stick" media.

[7] Battery pack latch (BATT Z) Flip this latch to the rear when you want to remove the battery.

[8] Chin sensor Senses when you touch the AIBO robot's chin.

[9] FCC ID/MAC address label Indicates the FCC ID and MAC address of the AIBO robot's wireless unit.

[10] Battery slot Holds the AIBO robot's lithium-ion battery.

[11] "Memory Stick" media eject button (Z) Press to eject the "Memory Stick" media. L "Memory Stick" media slot This is where you insert the provided AIBO-ware "Memory Stick" media.

If you experience difficulties ejecting the "Memory Stick" media or battery because of a malfunction or operation problems, place the AIBO robot in Pause mode, and then insert an object such as a paper clip into the emergency eject hole. (Do not use fragile objects, such as toothpicks, into the emergency eject hole as they may break.) Under normal circumstances, you do not need to use the emergency eject hole.

8.1.1 AIBO COLOR CAMERA

About the pictures o Pictures are stored on the "Memory Stick" media in JPEG format.The picture resolution is 416x320 pixels. Depending on lighting

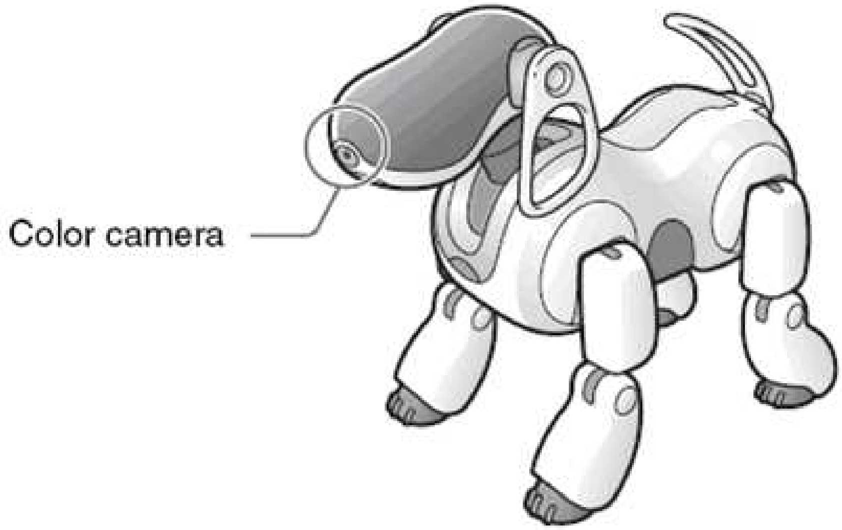


Fig. 25. Aibo color camera.

conditions at the time the picture is taken, flicker (horizontal stripes) may appear in pictures, or pictures may have red or blue hues.Fast movement may result in distortion of pictures.

The field of computer vision can be characterized as immature and diverse. Even though earlier work exists, it was not until the late 1970s that a more focused study of the field started when computers could manage the processing of large data sets such as images. However, these studies usually originated from various other fields, and consequently there is no standard formulation of the computer vision problem. Also, and to an even larger extent, there is no standard formulation of how computer vision problems should be solved. Instead, there exists an abundance of methods for solving various well-defined computer vision tasks, where the methods often are very task specific and seldom can be generalized over a wide range of applications. Many of the methods and applications are still in the state of basic research, but more and more methods have found their way into commercial products, where they often constitute a part of a larger system which can solve complex tasks (e.g., in the area of medical images, or quality control and measurements in industrial processes).

Computer vision is by some seen as a subfield of artificial intelligence where image data is being fed into a system as an alternative to text based input for controlling the behavior of a system. Some of the learning methods which are used in computer vision are based on learning techniques developed within artificial intelligence.

Since a camera can be seen as a light sensor, there are various methods in computer vision based on correspondences between a physical phenomenon related to light and images of that phenomenon. For example, it is possible to extract information about motion in fluids and about waves by analyzing images of these phenomena. Also, a subfield within computer vision deals with the physical process which given a scene of objects, light sources, and camera lenses forms the image in a camera. Consequently, computer vision can also be seen as an extension of physics.

A third field which plays an important role is neurobiology, specifically the study of the biological vision system. Over the last century, there has been an extensive study of eyes, neurons, and the brain structures devoted to processing of visual stimuli in both humans and various animals. This has led to a coarse, yet complicated, description of how real vision systems operate in order to solve certain vision related tasks. These results have led to a subfield within computer vision where artificial systems are designed to mimic the processing and behavior of biological systems, at different levels of complexity. Also, some of the learning-based methods developed within computer vision have their background in biology.

Yet another field related to computer vision is signal processing. Many existing methods for processing of one-variable signals, typically temporal signals, can be extended in a natural way to processing of two-variable signals or multi-variable signals in computer vision. However, because of the specific nature of images there are many methods developed within computer vision which have no counterpart in the processing of one-variable signals. A distinct character of these methods is the fact that they are non-linear which, together with the multi-dimensionality of the 19 22 signal, defines a subfield in signal processing as a part of computer vision.

Beside the above mentioned views on computer vision, many of the related research topics can also be studied from a purely mathematical point of view. For example, many methods in computer vision are based on statistics, optimization or geometry. Finally, a significant part of the field is devoted to the implementation aspect of computer vision; how existing methods can be realized in various combinations of software and hardware, or how these methods can be modified in order to gain processing speed without losing too much performance.

8.2.1 Related fields

Computer vision, Image processing, Image analysis, Robot vision and Machine vision are closely related fields. If you look inside text books which have either of these names in the title there is a significant overlap in terms of what techniques and applications they cover. This implies that the basic techniques that are used and developed in these fields are more or less identical, something which can be interpreted as there is only one field with different names.

On the other hand, it appears to be necessary for research groups, scientific journals, conferences and companies to present or market themselves as belonging specifically to one of these fields and, hence, various characterizations which distinguish each of the fields from the others have been presented. The following characterizations appear relevant but should not be taken as universally accepted.

Image processing and Image analysis tend to focus on 2D images, how to transform one image to another, e.g., by pixel-wise operations such as contrast enhancement, local operations such as edge extraction or noise removal, or geometrical transformations such as rotating the image. This characterization implies that image processing/ analysis does not produce nor require assumptions about what a specific image is an image of.

Computer vision tends to focus on the 3D scene projected onto one or several images, e.g., how to reconstruct structure or other information about the 3D scene from one or several images. Computer vision often relies on more or less complex assumptions about the scene depicted in an image.

Machine vision tends to focus on applications, mainly in industry, e.g., vision based autonomous robots and systems for vision based inspection or measurement. This implies that image sensor technologies and control theory often are integrated with the processing of image data to control a robot and that real-time processing is emphasized by means of efficient implementations in hardware and software. There is also a field called Imaging which primarily focus on the process of producing images, but sometimes also deals with processing and analysis of images. For example, Medical imaging contains lots of work on the analysis of image data in medical applications.

Finally, pattern recognition is a field which uses various methods to extract information from signals in general, mainly based on statistical approaches. A significant part of this field is devoted to applying these methods to image data. A consequence of this state of affairs is that you can be working in a lab related to one of these fields, apply methods from a second field to solve a problem in a third field and present the result at a conference related to a fourth field!

8.2.2 Examples of applications for computer vision

Another way to describe computer vision is in terms of applications areas. One of the most prominent application fields is medical computer vision or medical image processing. This area is characterized by the extraction of information from image data for the purpose of making a medical diagnosis of a patient. Typically image data is in the form of microscopy images, X-ray images, angiography images, ultrasonic images, and tomography images. An example of information which can be extracted from such image data is detection of tumours, arteriosclerosis or other malign changes. It can also be measurements of organ dimensions, blood flow, etc. This application area also supports medical research by providing new information, e.g., about the structure of the brain, or about the quality of medical treatments.

A second application area in computer vision is in industry. Here, information is extracted for the purpose of supporting a manufacturing process. One example is quality control where details or final products are being automatically inspected in order to find defects. Another example is measurement of position and orientation of details to be picked up by a robot arm. See the article on machine vision for more details on this area.

Military applications are probably one of the largest areas for computer vision, even though only a small part of this work is open to the public. The obvious examples are detection of enemy soldiers or vehicles and guidance of missiles to a designated target. More advanced systems for missile guidance send the missile to an area rather than a specific target, and target selection is made when the missile reaches the area based on locally acquired image data. Modern military concepts, such as battlefield awareness, imply that various sensors, including image sensors, provide a rich set of information about a combat scene which can be used to support strategic decisions. In this case, automatic processing of the data is used to reduce complexity and to fuse information from multiple sensors to increase reliability.

One of the newer application areas is autonomous vehicles, which include submersibles, land-based vehicles (small robots with wheels, cars or trucks), and aerial vehicles. An unmanned aerial vehicle is often denoted UAV. The level of autonomy ranges from fully autonomous (unmanned) vehicles to vehicles where computer vision based systems support a driver or a pilot in various situations. Fully autonomous vehicles typically use computer vision for navigation, i.e. for knowing where it is, or for producing a map of its environment (SLAM) and for detecting obstacles. It can also be used for detecting certain task specific events, e. g., a UAV looking for forest fires. Examples of supporting system are obstacle warning systems 21 24 in cars and systems for autonomous landing of aircraft. Several car manufacturers have demonstrated systems for autonomous driving of cars, but this technology has still not reached a level where it can be put on the market. There are ample examples of military autonomous vehicles ranging from advanced missiles to UAVs for recon missions or missile guidance. Space exploration is already being made with autonomous vehicles using computer vision, e. g., NASAs Mars Exploration Rover.

Other application areas include the creation of visual effects for cinema and broadcast, e.g., camera tracking or matchmoving, and surveillance.

8.2.3 Typical tasks of computer vision

Object recognition

Detecting the presence of known objects or living beings in an image, possibly together with estimating the pose of these objects.

Examples: Searching in digital images for specific content (content-based image retrieval) Recognizing human faces and their location in images. Estimation of the three-dimensional pose of humans and their limbs Detection of objects which are passing through a manufacturing process, e.g., on a conveyor belt, and estimation of their pose so that a robot arm can pick up the objects from the belt. Optical character recognition OCR (optical character recognition) takes pictures of printed or handwritten text and converts it into computer readable text such as ASCII or Unicode. In the past images were acquired with a computer scanner, however more recently some software can also read text from pictures taken with a digital camera.

Tracking

Tracking known objects through an image sequence.

Examples: Tracking a single person walking through a shopping center. Tracking of vehicles moving along a road.

Scene interpretation

Creating a model from an image/video.

Examples: Creating a model of the surrounding terrain from images, which are being taken by a robot-mounted camera. Anticipating the pattern of the image to determine size and density to estimate the volume using tomography like device. The cloud recognition is one the government project using this method.

Egomotion

The goal of egomotion computation is to describe the motion of an object with respect to an external reference system, by analyzing data acquired by sensors onboard on the object. i.e. the camera itself.

Examples: Given two images of a scene, determine the 3d rigid motion of the camera between the two views.

8.2.4 Computer vision systems

A typical computer vision system can be divided in the following subsystems:

Image acquisition

The image or image sequence is acquired with an imaging system (camera, radar, lidar, tomography system). Often the imaging system has to be calibrated before being used.

Preprocessing

In the preprocessing step, the image is being treated with low-level-operations. The aim of this step is to do noise reduction on the image (i.e. to dissociate the signal from the noise) and to reduce the overall amount of data. This is typically being done by employing different (digital)image processing methods such as: Downsampling the image. Applying digital filters convolutions, computing a scale space representation Correlations or linear shift invariant filters Sobel operator Computing the x- and y-gradient (possibly also the time-gradient). Segmenting the image. Pixelwise thresholding. Performing an eigen-transform on the image Fourier transform Doing motion estimation for local regions of the image (also known as optical flow estimation). Estimating disparity in stereo images.

Feature extraction

The aim of feature extraction is to further reduce the data to a set of features, which ought to be invariant to disturbances such as lighting conditions, camera position, noise and distortion. Examples of feature extraction are: Performing edge detection or estimation of local orientation. Extracting corner features. Detecting blob features. Extracting spin images from depth maps. Extracting geons or other three-dimensional primitives, such as superquadrics. Acquiring contour lines and maybe curvature zero crossings. Generating features with the Scale-invariant feature transform.

Registration

The aim of the registration step is to establish correspondence between the features in the acquired set and the features of known objects in a model-database and/or the features of the preceding image. The registration step has to bring up a final hypothesis. To name a few methods: Least squares estimation Hough transform in many variations Geometric hashing Particle filtering RANdom SAmple Consensus.

The Traffic Sign Recognition (TSR) is a field of applied computer vision research concerned with the automatical detection and classification of traffic signs in traffic scene images acquired from a moving car. Most part of the work done in this field is enclosed in the problem of the Intelligent Transportation Systems (ITS), which aim is to provide Driver Support Systems (DSS) with the ability to understand its neighborhood environment and so permit advanced driver support such as collision prediction and avoidance. Driving is a task based fully on visual information processing. The road signs and traffic signals define a visual language interpreted by drivers. Road signs carry many information necessary for successful driving - they describe current traffic situation, define right-of-way, prohibit or permit certain directions, warn about risky factors, etc. Road signs also help drivers with navigation. Two basic applications of TSR are under consideration in the research community - drivers aid (DSS) and automated surveillance of road traffic devices. It is desirable to design smart car control systems in such a way to allow evolution of fully autonomous vehicles in the future. The TSR system is also being considered as the valuable complement of the GPS-based navigation system. The dynamical environmental map may be enriched by road sign types and positions (acquired by TSR) and so help with the precision of current vehicle position.

Mobile mapping: the Geomobil project on Mobile mapping is a useful technique used to compile cartographic information from a mobile vehicle. The mobile vehicle is usually equipped with a set of sensors synchronized with an orientation system in order to link the obtained information with its position over the map. We are working with the mobile mapping system named Geomobil. The Geomobil is a Land Based Mobile Mapping System (LBMMS) developed by the Institut Cartogr‘afic de Catalunya (ICC) (fig.26). It is a modular system that allows the direct orientation of any sensor mounted on a roof platform. The Geomobil system is composed of the following subsystems: orientation subsystem, image subsystem, laser ranging subsystem, synchronization subsystem, power and environmental control subsystem. In our case we only use information from the image and orientation subsystems, which will be briefly explained in the rest of this point.

Geomobil system: the orientation subsystem is responsible for georeferencing the images acquired by the Geomobil. Thus it provides the coordinates (position) and the angles (attitude) of their projection centers. It is a system that combines inertial and GPS observations at a high level of integration, where the GPS derived trajectories are used to correct and calibrate the drifts of the Inertial Measurement Unit (IMU) gyros and accelerometers so that the position and velocity errors derived from inertial sensors are minimized. This combination of GPS and IMU systems allows the system to calculate the position even when the GPS satellites signals are blocked by terrain conditions (buildings, bridges, tunnels,...). The image subsystem design has been driven by two main requirements: to acquire images of at least 1Mpix and to get 10m stereoscopic overlap at a 10m distance from the van.

Fig. 26. Geomobil system.

The stereo overlap is conditioned by two factors: getting the maximum stereoscopic overlap free of obstacles and preserving a B/D ratio (stereoscopic Base - object Distance) as good as possible. The system links the captured images with their position and orientation data, and saves the information to the discs. The acquisition frequency is limited by the storage system capacity, and nowadays is programmed to take a stereo-pair of images each 10 meters or a turn higher than 60 degrees, which corresponds to the camera field of view

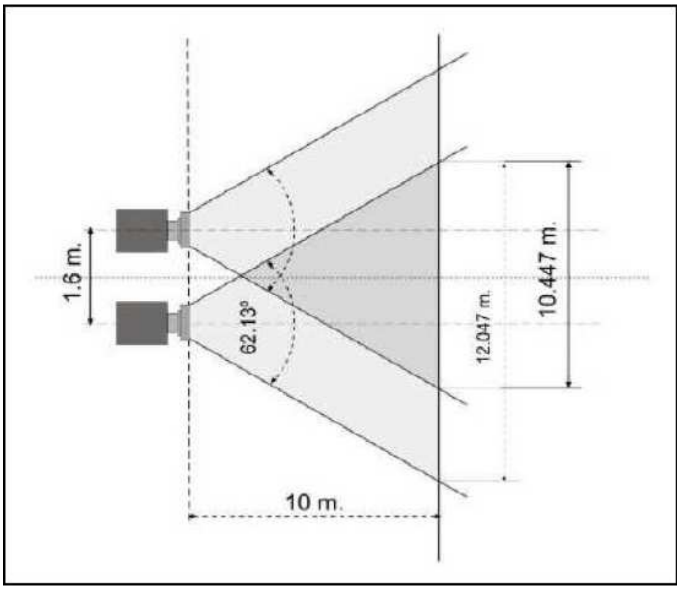


Fig. 27. Stereoscopic system diagram. We can see the relation between overlap zone and distance.

Feature	Value
Number of pixels	1024×1020
Pixel size	$12\mu m$
Focal length	$10.2mm$
FOV	62.13^o
IFOV	3 min. 38 sec.
Precision at $10m$ (across-track)	$0.8cm$
Precision at $10m$ (along-track)	$5.6cm$

Fig. 28. Geovan camera characteristics.

8.4 Artificial Intelligence History

8.4.1 Prehistory of AI

Humans have always speculated about the nature of mind, thought, and language, and searched for discrete representations of their knowledge. Aristotle tried to formalize this speculation by means of syllogistic logic, which remains one of the key strategies of AI. The first is-a hierarchy was created in 260 by Porphyry of Tyros. Classical and medieval grammarians explored more subtle features of language that Aristotle shortchanged, and mathematician Bernard Bolzano made the first modern attempt to formalize semantics in 1837.

Early computer design was driven mainly by the complex mathematics needed to target weapons accurately, with analog feedback devices inspiring an ideal of cybernetics. The expression "artificial intelligence" was introduced as a 'digital' replacement for the analog 'cybernetics'.

8.4.2 Development of AI theory

Much of the (original) focus of artificial intelligence research draws from an experimental approach to psychology, and emphasizes what may be called linguistic intelligence (best exemplified in the Turing test).

Approaches to Artificial Intelligence that do not focus on linguistic intelligence include robotics and collective intelligence approaches, which focus on active manipulation of an environment, or consensus decision making, and draw from biology and political science when seeking models of how "intelligent" behavior is organized. AI also draws from animal studies, in particular with insects, which are easier to emulate as robots (see artificial life), as well as animals with more complex cognition, including apes, who resemble humans in many ways but have less developed capacities for planning and cognition. Some researchers argue that animals, which are apparently simpler than humans, ought to be considerably easier to mimic. But satisfactory computational models for animal intelligence are not available.

Seminal papers advancing AI include "A Logical Calculus of the Ideas Immanent in Nervous Activity" (1943), by Warren McCulloch and Walter Pitts, and "On Computing Machinery and Intelligence" (1950), by Alan Turing, and "Man-Computer Symbiosis" by J.C.R. Licklider. See Cybernetics and Turing test for further discussion. There were also early papers which denied the possibility of machine intelligence on logical or philosophical grounds such as "Minds, Machines and Godel" (1961) by John Lucas. With the development of practical techniques based on AI research, advocates of 10 AI have argued that opponents of AI have repeatedly changed their position on tasks such as computer chess or speech recognition that were previously regarded as "intelligent" in order to deny the accomplishments of AI. Douglas Hofstadter, in Godel, Escher, Bach, pointed out that this moving of the goalposts effectively defines "intelligence" as "whatever humans can do that machines cannot". John von Neumann (quoted by E.T. Jaynes) anticipated this in 1948 by saying, in response to a comment at a lecture that it was impossible for a machine to think: "You insist that there is something a machine cannot do. If you will tell me precisely what it is that a machine cannot do, then I can always make a machine which will do just that!". Von Neumann was presumably alluding to the Church-Turing thesis which states that any effective procedure can be simulated by a (generalized) computer.

In 1969 McCarthy and Hayes started the discussion about the frame problem with their essay, "Some Philosophical Problems from the Standpoint of Artificial Intelligence".

8.4.3 Experimental AI research

Artificial intelligence began as an experimental field in the 1950s with such pioneers as Allen Newell and Herbert Simon, who founded the first artificial intelligence laboratory at Carnegie Mellon University, and John McCarthy and Marvin Minsky, who founded the MIT AI Lab in 1959. They all attended the Dartmouth College summer AI conference in 1956, which was organized by McCarthy, Minsky, Nathan Rochester of IBM and Claude Shannon.

Historically, there are two broad styles of AI research - the "neats" and "scruffies". "Neat", classical or symbolic AI research, in general, involves symbolic manipulation of abstract concepts, and is the methodology used in most expert systems. Parallel to this are the "scruffy", or "connectionist", approaches, of which artificial neural networks are the best-known example, which try to "evolve" intelligence through building systems and then improving them through some automatic process rather than systematically designing something to complete the task. Both approaches appeared very early in AI history.

Throughout the 1960s and 1970s scruffy approaches were pushed to the background, but interest was regained in the 1980s when the limitations of the "neat" approaches of the time became clearer. However, it has become clear that contemporary methods using both broad approaches have severe limitations.

Artificial intelligence research was very heavily funded in the 1980s by the Defense Advanced Research Projects Agency in the United States and by the fifth generation computer systems project in Japan. The failure of the work funded at the time to produce immediate results, despite the grandiose promises of some AI practitioners, led to correspondingly large cutbacks in funding by government agencies in the late 1980s, leading to a general downturn in activity in the field known as AI winter. Over the following decade, many AI researchers moved into related areas with more modest goals such as machine learning, robotics, and computer vision, though research in pure AI continued at reduced levels.

8.4.4 Micro-World AI

The real world is full of distracting and obscuring detail: generally science progresses by focusing on artificially simple models of reality (in physics, frictionless planes and perfectly rigid bodies, for example). In 1970 Marvin Minsky and Seymour Papert, of the MIT AI Laboratory, proposed that AI research should likewise focus on developing programs capable of intelligent behaviour in artificially simple situations known as micro-worlds. Much research has focused on the so-called blocks world, which consists of coloured blocks of various shapes and sizes arrayed on a flat surface.

8.4.5 Spinoffs

Whilst progress towards the ultimate goal of human-like intelligence has been slow, many spinoffs have come in the process. Notable examples include the languages LISP and Prolog, which were invented for AI research but are now used for non-AI tasks. Hacker culture first sprang from AI laboratories, in particular the MIT AI Lab, home at various times to such luminaries as John McCarthy, Marvin Minsky, Seymour Papert (who developed Logo there) and Terry Winograd (who abandoned AI after developing SHRDLU).

8.4.6 AI languages and programming styles

AI research has led to many advances in programming languages including the first list processing language by Allen Newell et. al., Lisp dialects, Planner, Actors, the Scientific Community Metaphor, production systems, and rule-based languages. GOFAI TEST research is often done in programming languages such as Prolog or Lisp. Bayesian work often uses Matlab or Lush (a numerical dialect of Lisp). These languages include many specialist probabilistic libraries. Real-life and especially real-time systems are likely to use C++. AI programmers are often academics and emphasise rapid development and prototyping rather than bulletproof software engineering practices, hence the use of interpreted languages to empower rapid command-line testing and experimentation.

The most basic AI program is a single If-Then statement, such as "If A, then B.” If you type an ’A’ letter, the computer will show you a ’B’ letter. Basically, you are teaching a computer to do a task. You input one thing, and the computer responds with something you told it to do or say. All programs have If-Then logic. A more complex example is if you type in ”Hello.”, and the computer responds ”How are you today?” This response is not the computer’s own thought, but rather a line you wrote into the program before. Whenever you type in ”Hello.”, the computer always responds ”How are you today?”. It seems as if the computer is alive and thinking to the casual observer, but actually it is an automated response. AI is often a long series of If-Then (or Cause and Effect) statements.

A randomizer can be added to this. The randomizer creates two or more response paths. For example, if you type ”Hello”, the computer may respond with ”How are you today?” or ”Nice weather” or ”Would you like to play a game?” Three responses (or ’thens’) are now possible instead of one. There is an equal chance that any one of the three responses will show. This is similar to a pull-cord talking doll that can respond with a number of sayings. A computer AI program can have thousands of responses to the same input. This makes it less predictable and closer to how a real person would respond, arguably because living people respond somewhat unpredictably. When thousands of input (”if”) are written in (not just ”Hello.”) and thousands of responses (”then”) are written into the AI program, then the computer can talk (or type) with most people, if those people know the If statement input lines to type.

Many games, like chess and strategy games, use action responses instead of typed responses, so that players can play against the computer. Robots with AI brains would use If-Then statements and randomizers to make decisions and speak. However, the input may be a sensed object in front of the robot instead of a ”Hello.” line, and the response may be to pick up the object instead of a response line.

8.4.7 Chronological History

Historical Antecedents

Greek myths of Hephaestus and Pygmalion incorporate the idea of intelligent robots. In the 5th century BC, Aristotle invented syllogistic logic, the first formal deductive reasoning system.

Ramon Llull, Spanish theologian, invented paper ”machines” for discovering nonmathematical truths through combinattions of words from lists in the 13th century. By the 15th century and 16th century, clocks, the first modern measuring machines, were first produced using lathes. Clockmakers extended their craft to creating mechanical animals and other novelties. Rabbi Judah Loew ben Bezalel of Prague is said to have invented the Golem, a clay man brought to life (1580).

Early in the 17th century, Rene Descartes proposed that bodies of animals are nothing more than complex machines. Many other 17th century thinkers offered variations and elaborations of Cartesian mechanism. Thomas Hobbes published Leviathan, containing a material and combinatorial theory of thinking. Blaise Pascal created the second mechanical and first digital calculating machine (1642). Gottfried Leibniz improved Pascal's machine, making the Stepped Reckoner to do multiplication and division (1673) and evisioned a universal calculus of reasoning (Alphabet of human thought) by which arguments could be decided mechanically.

The 18th century saw a profusion of mechanical toys, including the celebrated mechanical duck of Jacques de Vaucanson and Wolfgang von Kempelen's phony chessplaying automaton, The Turk (1769).

Mary Shelley published the story of Frankenstein; or the Modern Prometheus (1818).

19th and Early 20th Century

George Boole developed a binary algebra (Boolean algebra) representing (some) "laws of thought." Charles Babbage and Ada Lovelace worked on programmable mechanical calculating machines.

In the first years of the 20th century Bertrand Russell and Alfred North Whitehead published Principia Mathematica, which revolutionized formal logic. Russell, Ludwig Wittgenstein, and Rudolf Carnap lead philosophy into logical analysis of knowledge. Karel Capek's play R.U.R. (Rossum's Universal Robots)) opens in London (1923). This is the first use of the word "robot" in English.

Mid 20th century and Early AI

Warren Sturgis McCulloch and Walter Pitts publish "A Logical Calculus of the Ideas Immanent in Nervous Activity" (1943), laying foundations for artificial neural networks. Arturo Rosenblueth, Norbert Wiener and Julian Bigelow coin the term "cybernetics" in a 1943 paper. Wiener's popular book by that name published in 1948. Vannevar Bush published As We May Think (The Atlantic Monthly, July 1945) a prescient vision of the future in which computers assist humans in many activities.

The man widely acknowledged as the father of computer science, Alan Turing, published "Computing Machinery and Intelligence" (1950) which introduced the Turing test as a way of operationalizing a test of intelligent behavior.

Claude Shannon published a detailed analysis of chess playing as search (1950). Isaac Asimov published his Three Laws of Robotics (1950).

1956: John McCarthy coined the term "artificial intelligence" as the topic of the Dartmouth Conference, the first conference devoted to the subject. Demonstration of the first running AI program, the Logic Theorist (LT) written by Allen Newell, J.C. Shaw and Herbert Simon (Carnegie Institute of Technology, now Carnegie Mellon University).

1957: The General Problem Solver (GPS) demonstrated by Newell, Shaw and Simon.

1952-1962: Arthur Samuel (IBM) wrote the first game-playing program, for checkers (draughts), to achieve sufficient skill to challenge a world champion. Samuel's machine learning programs were responsible for the high performance of the checkers player.

1958: John McCarthy (Massachusetts Institute of Technology or MIT) invented the Lisp programming language. Herb Gelernter and Nathan Rochester (IBM) described a theorem prover in geometry that exploits a semantic model of the domain in the form of diagrams of "typical" cases. Teddington Conference on the Mechanization of Thought Processes was held in the UK and among the papers presented were John McCarthy's Programs with Common Sense, Oliver Selfridge's Pandemonium, and Marvin Minsky's Some Methods of Heuristic Programming and Artificial Intelligence.

Late 1950s and early 1960s: Margaret Masterman and colleagues at University of Cambridge design semantic nets for machine translation.

1961: James Slagle (PhD dissertation, MIT) wrote (in Lisp) the first symbolic integration program, SAINT, which solved calculus problems at the college freshman level.

1962: First industrial robot company, Unimation, founded.

1963: Thomas Evans' program, ANALOGY, written as part of his PhD work at MIT, demonstrated that computers can solve the same analogy problems as are given on IQ tests. Edward Feigenbaum and Julian Feldman published Computers and Thought, the first collection of articles about artificial intelligence.

1964: Danny Bobrow's dissertation at MIT (technical report from MIT's AI group, Project MAC), shows that computers can understand natural language well enough to solve algebra word problems correctly. Bert Raphael's MIT dissertation on the SIR program demonstrates the power of a logical representation of knowledge for question-answering systems.

1965: J. Alan Robinson invented a mechanical proof procedure, the Resolution Method, which allowed programs to work efficiently with formal logic as a representation language. Joseph Weizenbaum (MIT) built ELIZA (program), an interactive program that carries on a dialogue in English language on any topic. It was a popular toy at AI centers on the ARPANET when a version that "simulated" the dialogue of a psychotherapist was programmed.

1966: Ross Quillian (PhD dissertation, Carnegie Inst. of Technology, now CMU) demonstrated semantic nets. First Machine Intelligence workshop at Edinburgh: the first of an influential annual series organized by Donald Michie and others. Negative report on machine translation kills much work in Natural language processing (NLP) for many years.

1967: Dendral program (Edward Feigenbaum, Joshua Lederberg, Bruce Buchanan, Georgia Sutherland at Stanford University) demonstrated to interpret mass spectra on organic chemical compounds. First successful knowledge-based program for scientific reasoning. Joel Moses (PhD work at MIT) demonstrated the power of symbolic reasoning for integration problems in the Macsyma program. First successful knowledge-based program in mathematics. Richard Greenblatt (programmer) at MIT built a knowledge-based chess-playing program, MacHack, that was good enough to achieve a class-C rating in tournament play.

1968: Marvin Minsky and Seymour Papert publish Perceptrons, demonstrating limits of simple neural nets.

1969: Stanford Research Institute (SRI): Shakey the Robot, demonstrated combining animal locomotion, perception and problem solving. Roger Schank (Stanford) defined conceptual dependency model for natural language understanding. Later developed (in PhD dissertations at Yale University) for use in story understanding by Robert Wilensky and Wendy Lehnert, and for use in understanding memory by Janet Kolodner. Yorick Wilks (Stanford) developed the semantic coherence view of language called Preference Semantics, embodied in the first semantics-driven machine translation program, and the basis of many PhD dissertations since such as Bran Boguraev and David Carter at Cambridge. First International Joint Conference on Artificial Intelligence (IJCAI) held at Stanford.

1970: Jaime Carbonell (Sr.) developed SCHOLAR, an interactive program for computer assisted instruction based on semantic nets as the representation of knowledge. Bill Woods described Augmented Transition Networks (ATN's) as a representation for natural language understanding. Patrick Winston's PhD program, ARCH, at MIT learned concepts from examples in the world of children's blocks. Early 70's: Jane Robinson and Don Walker established an influential Natural Language Processing group at SRI.

1971: Terry Winograd's PhD thesis (MIT) demonstrated the ability of computers to understand English sentences in a restricted world of children's blocks, in a coupling of his language understanding program, SHRDLU, with a robot arm that carried out instructions typed in English.

1972: Prolog programming language developed by Alain Colmerauer.

1973: The Assembly Robotics Group at University of Edinburgh builds Freddy Robot, capable of using visual perception to locate and assemble models. The Lighthill report gives a largely negative verdict on AI research in Great Britain and forms the basis for the decision by the British government to discontine support for AI research in all but two universities.

1974: Ted Shortliffe's PhD dissertation on the MYCIN program (Stanford) demonstrated the power of rule-based systems for knowledge representation and inference in the domain of medical diagnosis and therapy. Sometimes called the first expert system. Earl Sacerdoti developed one of the first planning programs, ABSTRIPS, and developed techniques of hierarchical planning.

1975: Marvin Minsky published his widely-read and influential article on Frames as a representation of knowledge, in which many ideas about schemas and semantic links are brought together. The Meta-Dendral learning program produced new results in chemistry (some rules of mass spectrometry) the first scientific discoveries by a computer to be published in a referreed journal.

Mid 70's: Barbara Grosz (SRI) established limits to traditional AI approaches to discourse modeling. Subsequent work by Grosz, BonnieWebber and Candace Sidner developed the notion of "centering", used in establishing focus of discourse and anaphoric references in NLP. David Marr and MIT colleagues describe the "primal sketch" and its role in visual perception.

1976: Douglas Lenat's AM program (Stanford PhD dissertation) demonstrated the discovery model (loosely-guided search for interesting conjectures). Randall Davis demonstrated the power of meta-level reasoning in his PhD dissertation at Stanford.

Late 70's: Stanford's SUMEX-AIM resource, headed by Ed Feigenbaum and Joshua Lederberg, demonstrates the power of the ARPAnet for scientific collaboration.

1978: Tom Mitchell, at Stanford, invented the concept of Version Spaces for describing the search space of a concept formation program. Herbert Simon wins the Nobel Prize in Economics for his theory of bounded rationality, one of the cornerstones of AI known as "satisficing". The MOLGEN program, written at Stanford by Mark Stefik and Peter Friedland, demonstrated that an object-oriented programming representation of knowledge can be used to plan gene-cloning experiments.

1979: Bill VanMelle's PhD dissertation at Stanford demonstrated the generality of MYCIN's representation of knowledge and style of reasoning in his EMYCIN program, the model for many commercial expert system "shells". Jack Myers and Harry Pople at University of Pittsburgh developed INTERNIST, a knowledge-based medical diagnosis program based on Dr. Myers' clinical knowledge. Cordell Green, David Barstow, Elaine Kant and others at Stanford demonstrated the CHI system for automatic programming. The Stanford Cart, built by Hans Moravec, becomes the first computer-controlled, autonomous vehicle when it successfully traverses a chair-filled room and circumnavigates the Stanford AI Lab. Drew McDermott and Jon Doyle at MIT, and John McCarthy at Stanford begin publishing work on nonmonotonic logics and formal aspects of truth maintenance.

1980s: Lisp machines developed and marketed. First expert system shells and commercial applications.

1980: Lee Erman, Rick Hayes-Roth, Victor Lesser and Raj Reddy published the first description of the blackboard model, as the framework for the HEARSAY-II speech understanding system. First National Conference of the American Association for Artificial Intelligence (AAAI) held at Stanford.

1981: Danny Hillis designs the connection machine, a massively parallel architecture that brings new power to AI, and to computation in general. (Later founds Thinking Machines, Inc.)

1982: The Fifth Generation Computer Systems project (FGCS), an initiative by Japan's Ministry of International Trade and Industry, begun in 1982, to create a "fifth generation computer" (see history of computing hardware) which was supposed to perform much calculation utilizing massive parallelism.

1983: John Laird and Paul Rosenbloom, working with Allen Newell, complete CMU dissertations on Soar (program). James F. Allen invents the Interval Calculus, the first widely used formalization of temporal events.

Mid 80's: Neural Networks become widely used with the Backpropagation algorithm (first described by Paul Werbos in 1974).

1985: The autonomous drawing program, AARON, created by Harold Cohen, is demonstrated at the AAAI National Conference (based on more than a decade of work, and with subsequent work showing major developments).

1987: Marvin Minsky publishes The Society of Mind, a theoretical description of the mind as a collection of cooperating agents.

1989: Dean Pomerleau at CMU creates ALVINN (An Autonomous Land Vehicle in a Neural Network), which grew into the system that drove a car coast-to-coast under computer control for all but about 50 of the 2850 miles.

1990s: Major advances in all areas of AI, with significant demonstrations in machine learning, intelligent tutoring, case-based reasoning, multi-agent planning, scheduling, uncertain reasoning, data mining, natural language understanding and translation, vision, virtual reality, games, and other topics. Rodney Brooks' MIT Cog project, with numerous collaborators, makes significant progress in building a humanoid robot.

Early 90's: TD-Gammon, a backgammon program written by Gerry Tesauro, demonstrates that reinforcement (learning) is powerful enough to create a championshiplevel game-playing program by competing favorably with world-class players.

1997: The Deep Blue chess program (IBM) beats the world chess champion, Garry Kasparov, in a widely followed match. First official RoboCup football (soccer) match featuring table-top matches with 40 teams of interacting robots and over 5000 spectators.

1998: Tim Berners-Lee published his Semantic Web Road map paper [2]. Late 90's: Web crawlers and other AI-based information extraction programs become essential in widespread use of the World Wide Web. Demonstration of an Intelligent room and Emotional Agents at MIT's AI Lab. Initiation of work on the Oxygen architecture, which connects mobile and stationary computers in an adaptive network.

2000: Interactive robopets ("smart toys") become commercially available, realizing the vision of the 18th century novelty toy makers. Cynthia Breazeal at MIT publishes her dissertation on Sociable machines, describing Kismet (robot), with a face that expresses emotions. The Nomad robot explores remote regions of Antarctica looking for meteorite samples.

2004: OWL Web Ontology Language W3C Recommendation (10 February 2004).

8.5 *Apropa't a la Ciència*

8.5.1 *Apropa't a la ciència.De la Recerca a la Innovació*

Fig. 29. Cartell de l'exposició

This information has been extracted directly from: [18]

Apropa't a la ciència.De la Recerca a la Innovació (fig.29)

Inauguració: 11 d'octubre de 2006 a les 19 h Oberta al públic del 12 d'octubre al 31 de juliol de 2007 L'exposició Com un exponent destacat del Pla de Recerca i Innovació 2005 - 2008 de la Generalitat de Catalunya, i coincidint amb Barcelona Ciència 2007, "Apropa't a la ciència" pretén acostar d'una manera didàctica i atractiva la ciència als ciutadans; entesa aquesta en un sentit ampli. És a dir, com una eina útil per a la gent, malgrat el desconeixement de molts dels seus aspectes, però de la qual se'n deriven evidents repercussions socials i de millora per a la qualitat de vida, és a dir el RETORN SOCIAL DE LA CIÈNCIA. També està adreada a fomentar l'interés dels més joves cap a aquesta disciplina.

Any de la ciència Sota el títol de Barcelona Ciència 2007 l'ajuntament de Barcelona commemora el centenari del premi Nobel atorgat a Santiago Ramon y Cajal. Engega un ampli programa cultural que posarà un accent especial en el vincle entre les ciències, la cultura i la societat. A més a més, el 2007 es commemora també el centenari de la creació de l'Institut d'Estudis Catalans i el de la Junta de Ampliación de Estudios, institució precursora del Consejo Superior de Investigaciones Científicas. Per tal de donar un impuls decisiu a la política en relació amb la promoció de la cultura científica i crear una definitiva major sensibilitat social i cultural cap a la ciència, aquest programa cultural anuncia entre les seves línies concretes d'acció la proposta que l'any 2007 sigui declarat "Any de la Ciència" a la ciutat de Barcelona. [17]

L'exposició Apropat a la Ciència. De la recerca a la innovació ha estat dissenyada per donar a conèixer els plans de recerca i desenvolupament impulsats per la Generalitat de Catalunya per tal de fer veure la relació entre els coneixements científics i la innovació tecnològica i promoure vocacions científiques. Se'ns ofereix en l'exposició amb exemples ben concrets i molt suggerents. En ella no se'ns parla de res que recordi el contingut dels llibres de text (potser perquè es reconeix de manera implícita que, si ho fes, seria difícil l'apropament del públic que es vol aconseguir), la seva finalitat no és fer comprendre conceptes teòrics, fórmules ni equacions. Se'ns presenta, en canvi, una àmplia panoràmica de l'activitat científica real, la que es produeix en diversos contextos i impregna la vida de totes les persones. Per això, per la novetat que representa i per les noves possibilitats educatives que ofereix, és molt important donar aquest nou significat, d'empresa collectiva, a la paraula 'ciència'.

Anem del passat més remot, de quan encara no hi havia humans sobre la Terra fins a l'avenir incert dels viatges espacials i dels robots. En Pau d'Hostalets de Pierola ens proporciona una ocasió per a pensar en el lent procés que ha donat lloc a l'emergència de l'espècie humana (una evolució afortunada de la clavícula que proporciona noves possibilitats de manipulació) i la cadira Mares ens fa veure les dificultats d'adaptar un cos que ha de moure's a la inactivitat forosa en l'interior d'una nau espacial. Els robots ens fan pensar en quines tasques faran en lloc nostre i com fer-les o no fer-les podrà afectar les nostres pròpies capacitats. S'han ampliat les comunicacions, que connecten els satèllits artificials, tan llunyans, amb els mòbils, tan propers. Les intervencions humanes en el món són ara d'abast planetari i transformen la Natura, perquè en són part. Les tecnologies per a l'aprofitament de l'energia (els molins de vent), per a la conservació dels aliments, per a prevenir malalties o superar-ne d'incurables (les vacunes i els trasplantaments), han de poder arribar a tot arreu.. La mostra ocupa físicament la sala 3 del Palau Robert dividida en vuit àmbits, a més dels escenaris un d'entrada i un de sortida, en el primer dels quals es fa una ràpida pinzellada a les aportacions realitzades pels grans científics de la història com ara Newton o Curie, entre molts d'altres. Aquesta introducció inicial remet al darrer dels apartats de tancament de l'exposició, tot invitant les generacions futures a prendre el relleu científic.

8.5.2 La robòtica i les seves aplicacions socials

Un futur de persones i robots. Tot i que per a la majoria de persones els robots són una realitat confinada a les fàbriques i a la producció, el futur de la robòtica no passa per les naus industrials. Si l'any 2000 els robots industrials representaven el 95% dels 6.000 milions de dòlars del mercat global de la robòtica, es calcula que durant el 2025 no passaran de ser el 20% d'aquest mercat (situat als 65.000 milions de dòlars), perdent el seu lideratge en favor dels robots personals i de serveis. Els seus usos es distribuiran en tres grans apartats: - En el sector professional, des de l'agricultura a la cirurgia passant pel transport i la construcció. L'aplicació de la robòtica en aquest sector, en el que Europa manté un cert lideratge, constitueix l'evolució natural de la robòtica tradicional i ha de permetre automatitzar processos fins ara exclusius dels humans durant la seva vida professional. - En el sector domèstic, des de la neteja de la casa fins a la cura de la gent gran o dels malalts. Aquestes aplicacions, liderades actualment pels EUA i Corea, faran canviar la imatge del robot amb forma de "braç mecànic" que munta el vidre d'un cotxe a la línia de producció per un robot, en alguns casos antropomòrfic, que conviu i actua a l'entorn domèstic de les persones fent tasques de suport a les persones. - El sector de l'oci i l'entreteniment, des de les joguines robotitzades fins als entrenadors personals d'algun esport. Aquest sector, liderat pel Japó, és ja una realitat amb una gran capacitat de creixement que pot fins i tot superar als jocs d'ordinador.

A l'exposició veurem el robot AIBO (fig.30) i les possibilitats que un programari especial té per a les persones amb discapacitats, persones grans, etc.

Fig. 30. Stand de la fira