OBJECT CLASSIFICATION ON A CONVEYING BELT

Miguel A. Fernandez[1], A. Fernandez Caballero[2], Juan Moreno, & Gabriel Sebastian

Departamento de Informatica
Escuela Politecnica Superior de Albacete
Universidad de Castilla-La Mancha, Albacete, Spain
email:[1]miki@info-ab.uclm.es, [2]caballer@info-ab.uclm.es

Abstract. This paper describes a system for object detection on a conveying belt. The detection algorithms are based on the permanence effect that basically consists on increasing a state variable associated to each pixel of the sensor when the moving element is located on that particular pixel point, and to decrease it otherwise. Using this permanence effect, a characteristic called LSR can be obtained (relationship between length and speed). This characteristic generally defines the moving element uniquely, making it possible to detect it on the conveying belt.

The permanence effect as well as the LSR characteristic biologically inspired mechanisms and very close to the accumulation mechanisms in the synapses of the biological neurons.

The described system has learning capacity, being able to modify the group of elements to be detected without necessity of modifying the system's algorithms. This system has been tested in the laboratory using a series of medicines, obtaining very satisfactory results. To show the versatility of the used mechanisms another application of the same system is included, in this case the recognition of faces.

1. Application description

This paper describes an object classification application on a conveying belt. The aim of the application is the recognition of a set of objects from the images taken by a camera located above a conveying belt. It must be stated from the very beginning that the proposed application has only been tested as a laboratory prototype. That's why this paper presents the results obtained in a series of simulations.

The final system will consists of (a) a camera, (b) a digitalization card, (c) a personal computer, and (d) a specific card for video processing. The application consists in the classification of objects passing in front of the camera objective on a conveying belt as indicated in figure 1.

The main distinguishing characteristic of this system with respect to others designed for the same aim, lies on its learning capacity. After a learning process consisting in showing a sequence of images accompanied by a signal which denominates the appearing elements, the system is capable of classifying the elements when passing under the camera at uniform speed.

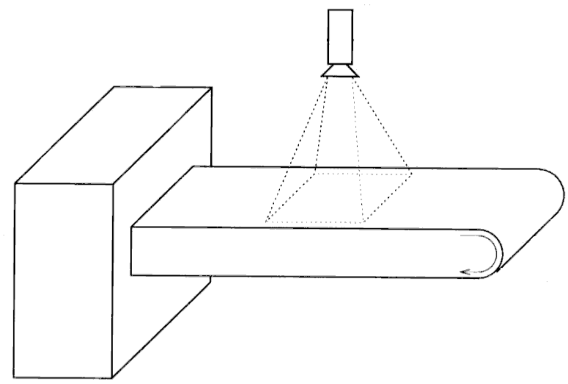

Figure 1. Camera above a conveying belt.

In a specific application, after the learning process, the system is able to classify and count the number of medicines (see figure 3) passing in front of the camera on the conveyor. Although this example shows the classification of a set of medicines passing on the conveyor, the system would be capable of classifying different sets of objets, such as labels, parts or others. It is essential that each element has uniform speed in front of the camera. At the end of the paper an example of face recognition is introduced.

The process can be carried out in real time, analyzing every pixel in each image with a simple but specific hardware. Before accomplishing the hardware, a simulation of it has been realized and all mechanisms and algorithms on which the system is based have been tested.

The simulation has been realized using the working environment shown in figure 2. As it can be seen it is made up of: (1) a black and white CCD TV camera, (2) a CCIR standard black and white television monitor, (3) a commercial image digitalization board, and (4) a personal computer.

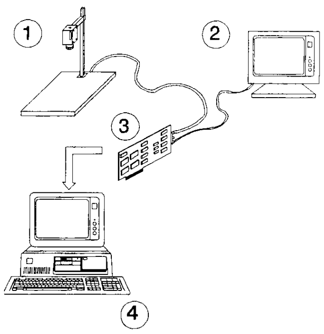

Figure 2. Simulation environment.

The simulation process was designed with the possibility to control and debug computational and learning processes. The application uses as input an undefined sequence of photograms (10,000 approximately). Although the future goal of the system is to be able to work in real time, the prototype described does not yet present this working capacity, although it does process the information frame by frame, allowing the input of a new frame when the previous one has been processed.

The working method of this simulation can be divided into the following steps:

1) *Design and assembly of the training, test and operation sequences.* This stage is carried out designing the scene background and the mobile elements. Afterwards a sequence file in which each photogram is described in a register is designed.

2) *Design of the assistance values*. The assistance values, used during the learning stage to indicate the application which object is being shown, are generated at the same time as the sequence files.

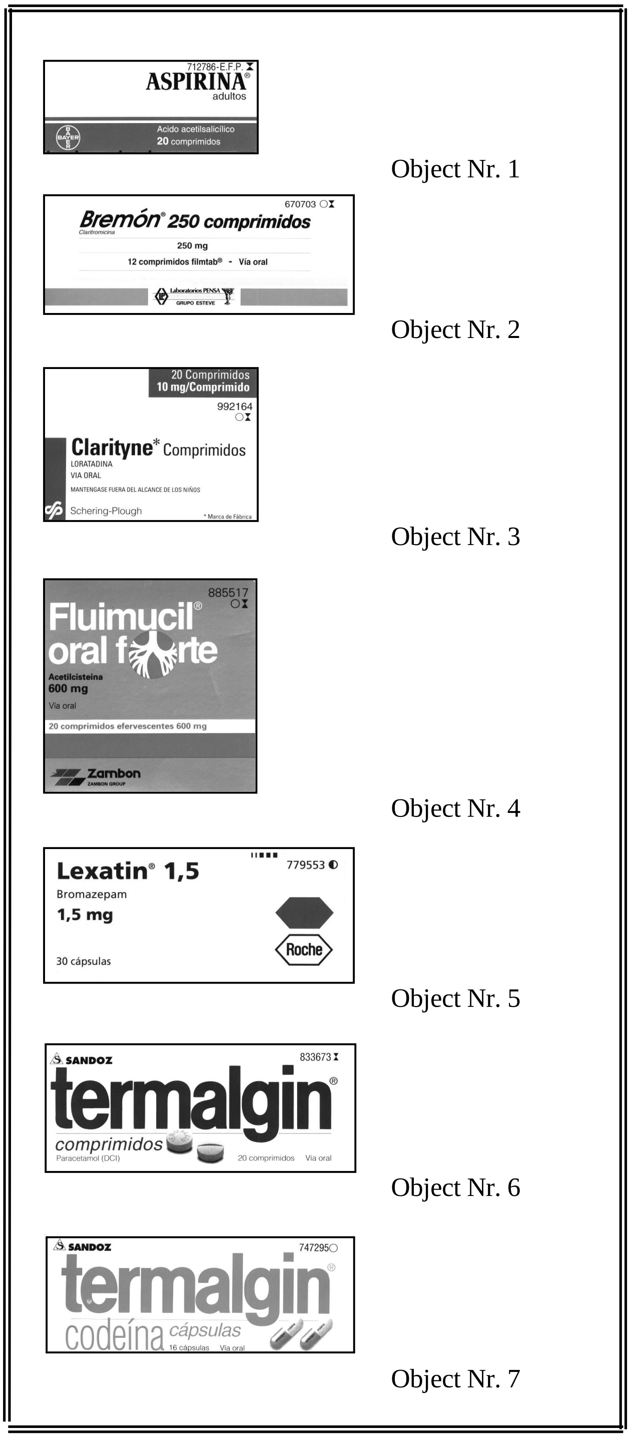

Figure 3. Used objects/medicines.

3) *State definition and preparation*. The application defines the values of every parameter of the system. These values, however, can be modified for specific applications. Such modifications can be made before starting the learning sessions or at any other point of the process.

4) Behavior and learning trace and analysis. Once the sequences are ready, the assistance values and the initial state of the system have been prepared, the process can be started.

Note that the philosophy of the application in itself implies the possibility to change the objects and adapt the system without requiring the intervention of a computer programmer. Simply placing the elements on the conveying belt and showing them accompanied by an assistance signal.

We would like to illustrate the whole described process with a specific example. The objects used in the application are those shown in figure 3. The learning process consisted in passing six times 1184 photograms. The different medicines appear consecutively, one by one, accompanied by a code (object number) which identifies each of them. The system can differentiate up to a maximum of 8 different elements. To elaborate this training sequence the image of each medicine has been digitized and transformed into an icon travelling on a static image background. All medicines are moving in the scene at the same speed (1 pixel/frame), separated in time by short intervals from one to the next, thus simulating the motion on a conveying belt.

After the learning stage (6 x 1184 photograms), the system is once more shown the same sequence, but this time without the signal identifying each medicine. The system answers correctly emitting an identification code corresponding to the medicine moving in the scene at each moment. Therefore it is capable to count the different types of medicines.

The simplicity of the system and in particular that of the learning process must be emphasized. Learning to classify is as easy as just passing the elements accompanied by their identification code in front of the camera. To obtain the required results, this operation has to be performed three times in most cases. It should be noted as well that the system is really versatile respect to the type of element to be classified. This is first due to the simplicity of the characteristic on which the classification is based. Consider also the fact that the learning capacity is present at the characteristic extraction stage as well as at the classification level. This work presents a particular application directed to identification of medicines passing on a conveying belt. But, the same system with a similar learning process can be used to classify a very wide variety of elements.

We have carried out tests with geometric figures, spare parts and even with pictures from human faces (see figure 4). In this last case four out of the six faces were perfectly distinguished (1, 2, 3, and 4) while the system wasn't able to distinguish between objects 5 and 6. The result may seem surprising at first sight. Why can the system not distinguish between 5 and 6, while it doesn't find any problem to differentiate the rest The answer lies on the nature of the characteristic used for the classification. It will be explained in the following sections.

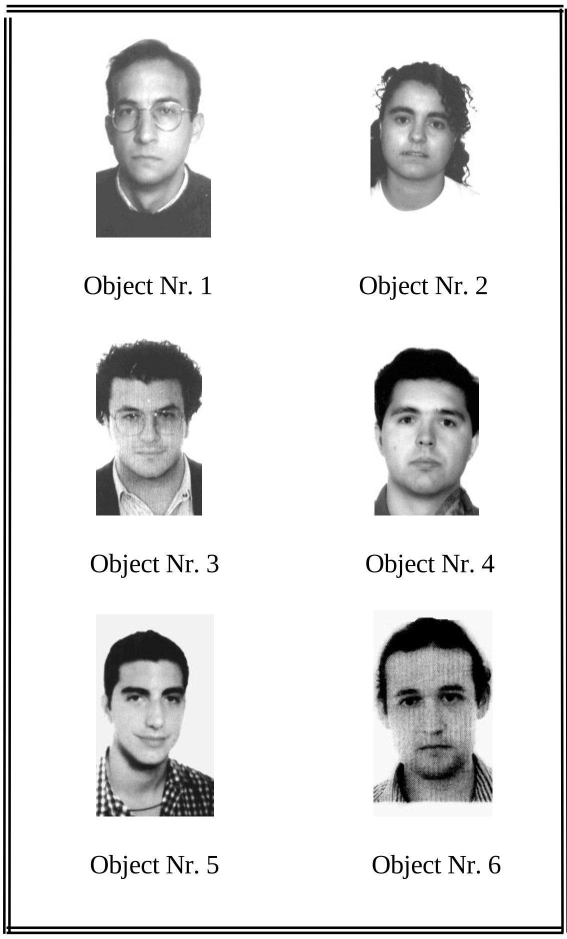

Figure 4. Used objects/faces.

2. General description of the model

2.1. Biological inspiration

Our model is based in the way motion is analyzed by the Pipiens frog and in the charge-discharge characteristics of neural synapses in the visual path.

We specially focused on the model being developed by J. Mira on the computation carried out by a synapses [Mir93] [Fer95]. There is enough evidence of the available knowledge in Calvin and Granband [Cal79], Koch [Koc90], Rall and Segev [Ral90], J. Mira et al. to be able to talk about analog microcomputation on a single neuron.

2.2. Permanence effect

Our work proposes as perceptive equivalent of this neurophysiological process at synaptic level the *permanence effect* created by images in vision sensors. This effect can be easily observed when an intense light focus -the torch used by a circus fakir, for example- blinds the vidicon of a video camera for seconds. As the fakir moves the torch, the luminous trail created by the torch motion can be observed on the television monitor.

Thanks to the saturation produced in those pixels of the vidicon receiving the blinding light of the torch, we can see the followed trajectory. The permanence effect as such is that there is a permanence value associated to each data. A specific value of the signal charges the permanence value, while a different one discharges it, normally with different intensities [Fer95] Fer97].

2.3. Length/speed ratio

Classification of objects on a conveyor in real time is just one particular case of detection of moving targets. Thinking about this problem, we realized that the period of time a specific gray level is present in the scene in the same coordinate is really very relevant. A rapid object stays for a shorter time than a slower one, and, at a same speed –which actually is our case- the greater the size of the target, the longer it stays on each pixel in the sensor. For this reason it seems reasonable to study the descriptor *length/speed ratio LSR* [Fer92].

The system presented in this paper uses the permanence effect described above . A same gray level remaining on a pixel in the sensor through a sequence of images, produces that the charge (or permanence) value associated to that pixel gradually increases from image to image. The variation of that gray level generates the LSR value (the charge value at that moment) associated to that pixel. This value is proportional to the time the same gray level has been invariant in such pixel (we use thick gray levels, namely 16, instead of the usual 256). After each variation of the gray level in a pixel of the sensor, its associated permanence value is reset (zero value). If the gray level in a same image pixel does not vary for a long period of time, the permanence value associated to that pixel will reach a saturation level which is not significant. The elements with no motion do not generate LSR data.

A moving element travelling in the scene will be composed of different uniform gray level areas and therefore it will generate a set of LSR data related to the shape of the different gray level areas and to their speed. The generated LSR data are independent from the scene background, as long as it is a static background. This set of LSR data is called LSR footprint and our subsequent classification is based on this concept.

Artificially this characteristic can be locally computed, it is not especially complex (real time) and presents good discriminant characteristics , and, therefore it is capable of learning. This means it is *neural*. This characteristic which weights the relationship between the length and speed of a uniform gray level stain travelling in the scene, and can be obtained by measuring the time the same gray level stain occupies a pixel in the sensor. A long stain travelling in the scene will occupy image pixels for a long period as it passes, and therefore high permanence values will be generated; logically, a short stain travelling in the scene will occupy less time in the sensor pixels, thus the permanence values generated will be lower the higher their speed is.

2.4. Multilayer structure

Once the characteristic used as foundation for further classification has been described, let us present the global structure of the system. A multilayer structure with three stages has been used as illustrated in figure 5.

In the first stage, called segmentation stage, gray level changes in each sensor pixel are detected. This layer presents hysteresis mechanisms to eliminate noise problems which could cause false variations in gray levels, mainly in the border of two consecutive gray levels.

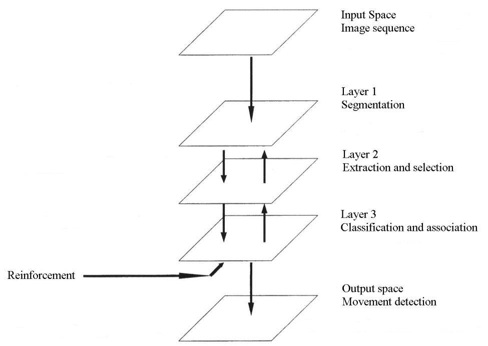

Figure 5. Multilayer structure.

The second stage is named characteristic extraction and selection stage and is dedicated to obtain LSR footprints. In the described system the LSR footprint is transformed into a binary vector of 8 bits, where each bit indicates the existence or not of LSR data with a specific value. The learning process at this level has been designed to define the LSR_{min} and LSR_{max} values of each of the 8 intervals. This in order to achieve that the layer output (8 binary values, each of them indicating the presence of LSR discharges within a specific interval) may correctly differentiate all the elements which make up the training family.

Thanks to the third stage, denominated classification and association stage, we differentiate the classes holding the elements of the learning family that make use of the set of characteristics extracted in the previous stage. Two learning processes can be distinguished at this level. On the one hand we distinguish the process that aims to separate the input space into classes, in such a way that each class is associated to an input element. On the other hand there is second phase of learning that associates each class to the code used to identify it.

The mechanisms used in all stages are neurally inspired and satisfy the requirements for this type of computation. The system, therefore, is robust as it is capable of identifying the input elements even though they may be accompanied by noise or they may be defective.

2.5. State of development

The previously described simulator -on which the basic algorithms of our system have been developed- can not work in real time as the computational load exceeds image frame latency (20 milliseconds). For this reason we have worked with synthetic images made up of icons and backgrounds from digitalization of real images taken by a TV camera. The simulator processes just a subset of pixels of each image. It works by composing the image and processing it before generating the next image.

After verifying the system's operation by means of the simulator, a specific hardware has been designed. This hardware is based on FPGA and memory which, hosted in a personal computer, will be capable of working with the whole image and will therefore allow to work in real time with camera sequences. The hardware has already been constructed and our research group is developing the configuration of the FPGA in order to carry out the required processes.

3. Limitations and possibilities

Of course some of this method's limitations have be admitted right from the beginning. These are mainly based on the fact that the use of specific descriptors does not guarantee a complete description, especially when the information is reduced from a sequence of images which contain moving elements into an eight bits vector which indicates the presence or absence of specific LSR values in each photogram. Therefore the elements generating the same LSR data can not be differentiated by the system.

Let us now consider invariants in the LSR characteristic. This characteristic remains constant facing variations in the distance to the sensor (dilation) or variations in motion direction (translation). These invariances are unimportant in the application of identification of elements on a conveyor we are dealing with now, but are very desirable in multiple applications in which the moving elements are not located at a fixed distance from the camera or do not have to appear with fixed direction. The LSR characteristic is not invariant however in turns with respect to motion direction (rotation). This variance is not very restrictive since moving elements generally present `first the head and the tail at the end´. In the present application this lack of invariance does not present problems if the medicines are always placed in the same way on the conveyor.

4. Summary and conclusions

To conclude we can affirm that the proposed system, as observed by the results obtained in the simulation, is valid for its application as an object classifier on a conveyor. More specifically, it can be used to classify and subsequently count or separate the different medicines or presented objects.

In our opinion, the most remarkable characteristics of the described application can be appreciated from two different points of view:

(1) From the final users viewpoint, the most remarkable feature is the simplicity of the process, not requiring the intervention of a computer programmer, nor the modification of algorithms, since the learning mechanisms are self-contained in the system's one. The system's versatility is also very important. As we have demonstrated, it can be used to classify training sets of very different nature.

(2) Although from the users point of view the system presents attractive characteristics, perhaps the most important features must be analyzed from the point of view of its internal nature. That is to say of the LSR footprint's use as the base for the classification and the introduction of learning processes, both at characteristic extraction level selection and regarding classification mechanisms. The most important feature of the proposed system is its capacity for selective information destruction. In fact this is the way biological systems work. A large amount of information is destroyed, and only the necessary information required to obtain the desired goals is processed. Our system works as an information destroyer capable of reducing all the information that exists in an undefined sequence of images to a vector of 8 bits. In this vector each bit indicates the existence of a LSR value between predetermined values in each frame. We insist on the fact that this important reduction of information is carried out by introducing learning processes which determine which are the most significant LSR values for each training set. Our present working lines include the design of a prototype with working capacity in real time, the search for new descriptors and the creation of dialogue and cooperation procedures between both.

Bibliography

[Cal79] W.H. Calvin & K. Graubard. (1979). Styles of neural computation. In: Schmitt & Warden, eds. The Neurosciences Fourth Study Program, 513-524. The MIT Press.

[Fer92] M.A. Fernandez & J. Mira. (1992). Permanence memory: A system for real time motion analysis in image sequences. IAPR Workshop on Machine Vision Applications, 249-252. MVA'92, Tokyo.

[Fer95] M.A. Fernández, J. Mira, M.T. López , J.R. Alvarez, A. Manjarrés & S. Barro. (1995). Local acumulation of persistent activity at synaptic level: Application to motion analyis. From Natural to Artificial Neural Computation, IWANN'95, 137-143. Springer-Verlag, Germany.

[Fer97] M.A. Fernández. (1997). Una arquitectura modular de inspiración biológica con capacidad de aprendizaje para el análisis de movimiento en secuencias de imagen en tiempo real. Tesis Doctorales, 48. Universidad de Castilla-La Mancha. Ph. D. thesis.

[Koc90] C. Koch. (1990). Biophysics of computation: Towards the mechanisms underlying information processing in single neurons. In: E.L. Schwartz, ed. Computational Neurosciences, 97-113. The MIT press.

[Mir93] J. Mira, A.E. Delgado, J.R. Alvarez & M. Santos. (1993). Towards more realistic self-contained models of neurons: High-order recurrence and local learning. New Trends in Neural Computation., IWANN'93, 57-62. Springer-Verlag, Germany.

[Ral90] W. Rall & I. Segev. (1990). Dendritic branches, spines, synapsis and excitable spine clusters. In: E.L. Schwartz, ed. Computational Neurosciences, 69-81. The MIT Press.