Tracking deformable objects with WISARD networks

Massimo De Gregorio*, Maurizio Giordano†, Silvia Rossi‡, Mariacarla Staffa‡, Bruno Siciliano‡
*Istituto di Cibernetica "Eduardo Caianiello", CNR
Via Campi Flegrei 34, Pozzuoli, Italy
Email: massimo.degregorio@cnr.it
†Istituto di Calcolo e Reti ad Alte Prestazioni, CNR
Via Pietro Castellino 111, Naples, Italy
Email: maurizio.giordano@cnr.it
‡Department of Electrical Engineering and Information Technology, University of Naples Federico II
Via Claudio 21, 80125, Naples, Italy
Email: {mariacarla.staffa, silvia.rossi, bruno.siciliano}@unina.it

***Abstract*— In this paper, we investigate a new approach based on WISARD Neural Network for the tracking of non-rigid deformable object. The proposed approach allows deploying an on–line training on the texture and shape features of the object, to adapt in real–time to changes, and to partially cope with occlusions. Moreover, the use of parallel classificators trained on the same set of images allows tracking the movements of the objects. We evaluate our tracking abilities in the scenario of pizza making that represents a very challenging benchmark to test the approach since in this context the shape of the object to track completely changes during the manipulation.**

I. Introduction

The object tracking problem consists in reconstructing the trajectory of objects along the sequence of images. It is considered as a basic problem in many computer vision applications and it is inherently difficult, especially when applied to real world conditions, where unstructured forms are considered for tracking, real time responses are required for adapting the robot movements in time, computational capabilities are limited to on-board units and where problems of brightness and non-stationary background can affect the performance of the elaboration system. Moreover, in case of non-rigid objects, the task of dynamic tracking becomes even more challenging. The state of the art of tracking deformable objects is still rather far from the real applicability within robotic applications. Recent projects have been proposed, in the last few years, trying to cope with this kind of problem. The recent RODYMAN project proposes, for example, the development of a unified framework for dynamic dexterous manipulation control, considering mobile platforms able to manipulate non-prehensile non-rigid objects, trying to fill the gap in the current state of the art. In order to achieve dexterous manipulation abilities, a fundamental step is to provide robots with the ability to efficiently track the objects to be manipulated. Both dynamic object tracking and manipulation become, in fact, the most complex categories of robotic tasks, which, if solved, could increase the opportunities for a wide adoption of robots within human co-habited environments.

Here, our aim is to address the problem of making a robot able to track any deformable object without an *apriori* physical model of the object that can dynamically change its shape during the tracking. In this preliminary work, we propose a particular neural network approach, a WiSARD–based system [1], used as feature detector for tracking deformable objects during manipulation. This particular weightless neural system has the property of being noise tolerance and is capable of learning step–by–step the new appearance of the moving object on a dynamic background without needing a model of the object to track. The WiSARD can be adopted to deploy virtual sensors that, with a limited use of computational resources, can be used on-board for object tracking and dynamical selection of the desired targets to track. In this paper we will introduce our approach and will evaluate robustness of the proposed tracking method in the task of pizza making.

II. Related Works

A wide class of approaches in object tracking explicitly assumes a model of the tracked objects. In tracking deformable objects, some attempts have been proposed in order to have a flexible model to track the objects [2] and to represent the elasticity and deformation characteristics during the physical interaction. In some of these cases, authors consider a pre-defined initial shape to be manipulated into a deformable contour model. In [3] authors use physical, although very general, models and a set of contraints on the model to estimate the state of objects. A probabilistic approach is used to associate and track such points with respect to the points obtained from point clouds of a RGB-D camera. Our proposed solution is more in line with features or appearance–based approaches, as the method of [4], where non-rigid objects are tracked based on visual features such as color and/or texture, object contours, regions of interest. In [4], for example, a statistical distributions is used to characterize the object of interest. The approach is based on mean shift iterations to find the target candidate that is the most similar to a given target model. In [5] authors presented a feature method for tracking both rigid and deformable objects (like human beings) in video sequences. The proposed tracking algorithm segments object regions based on motion and extracts some feature points to track

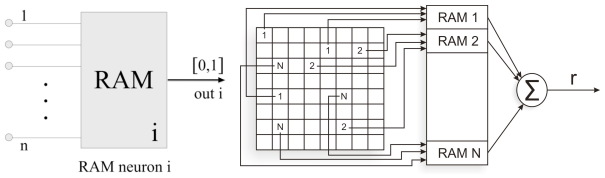

Fig. 1. RAM-neuron and a WiSARD discriminator.

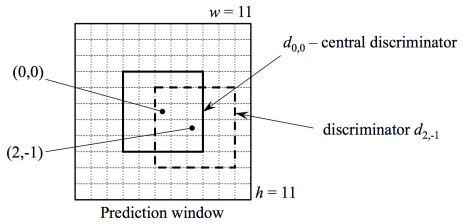

Fig. 2. Prediction window and discriminator retinas.

by using optical flow with online training. Conversely, in our work we track the complete appearance of the object. [6] uses a train of discriminative classifiers in an online manner to separate the object from the background with a model, which evolves during the tracking process as the appearance of the object changes. Finally, in [7] the estimation of non–rigid object is obtained by means of energy minimization and graph cuts.

III. WiSARD Networks

Weightless neural networks (WNNs) are based on networks of Random Access Memory (RAM) nodes. As illustrated by Figure 1, a RAM-based neuron is capable of recognizing n bits (n-tuple) inputs coming from a target pattern. WiSARD systems are a particular type of WNN. The WiSARDs can be developed directly on reprogrammable hardware. This characteristic finds a concrete applicability in embedded robotic systems.

On a WiSARD, RAM input lines (retina) are connected to the input pattern by means of a biunivocal pseudo-random mapping as a set of uncorrelated n-tuples (see right part of Figure 1). Each n-tuple is used as a specific address of a RAM–based neuron, in such a way that the input field is completely covered. A WiSARD discriminator is composed by a set of RAM–based neurons, and it is, in general, trained with representative data of a specific class/category. In our case the discrimintaor are trained on the current pattern. All RAM neurons are initialized with 0s in all of its contents; upon presentation of a (often binary) pattern, the contents of the specific RAM location addressed by the n-tuple are set to 1. The information stored by RAM nodes during the training phase is used to deal with previous unseen patterns. When one of these is given as input, RAM memory contents addressed by the input pattern are read and summed by Σ. The number r thus obtained, which is called the *discriminator response*, is equal to the number of RAMs that output '1'. The summing device enables this network of RAM nodes to exhibit generalization and noise tolerance [8].

A. WiSARD for Movements Tracking

The WiSARD system we propose, is formed by a certain number of RAM–discriminators each one looking at different parts of the image (see Figure 2). When a pattern is given as input, each discriminator gives a response to that input. The various responses are evaluated by an algorithm which compares them and computes the relative confidence c of the highest response. We can distinguish left, right, up and down discriminators, respectively to track left, right, up and down displacements of the tracked object. Doing so, each discriminator is identified by its relative coordinates. The displacement of all the retinas forms what it is called *prediction window*. In particular, since we consider prediction window precision of 10 pixels, we will use $21 \times 21 = 441$ discriminators (included the central one). The generic discriminator $d_{i,j}$ is going to be responsible for detecting the object in case its new position is identified by (i, j) in the prediction window. The discriminators accept as input binary patterns.

B. DRASiW for Shape Tracking

DRASiW is an extension to the WiSARD model provided with the ability of producing pattern examples, or prototypes, derived from learned categories. RAM–discriminators are modified in what their memory locations may hold and, correspondingly, in their training algorithm. These changes allow one to store q-bit words in memory locations; this information, in turn, can be exploited in the generation of "mental" images (MIs) of learned pattern categories. The training algorithm of RAM–discriminators is changed mainly in one aspect only: instead of storing '1's, it just increments $(+1)$ memory location contents that are addressed by input patterns. The various memory content values can now be associated to subpattern frequency in the training set. In order to avoid RAM memory location saturation, we introduce a forgetting mechanism (bleaching [9]) that allows DRASiW to store in its MI an updated shape of the tracked object. The system always trains itself with the image on the retina of the discriminator that outputs the best response. In particular, all the sub-patterns of the new image on the retina are combined with those of the MI (this means increasing their frequencies in the RAM contents). On the other hand, those subpatterns which were not addressed by the image on the retina are decremented (-1). So doing, DRASiW system will get always an updated MI of the object shape it is tracking. Furthermore, with the MI stored during time, we can produce a sort of object shape history. This history can represent a fundamental facility in the case we need to extract from the tracking a cinematic/dynamic model of the object to be manipulated.

IV. Our Framework

The tracking activity done by the overall system is performed as follows. In order to transform the input video

frame image in a suitable format for WiSARD discriminators, i.e. a black and white image, we developed a filter that binarizes the image of interest by identifying the more frequent colors in a given region (*focus area*) included in the target object. The main filter steps are depicted in Figure 3.

Before the tracking starts, the object to be tracked is selected by the user by drawing a bounding box (the blue rectangle in Figure 3). The filter identifies a focus area. More precisely, the focus ares is represented by a box (the red rectangle in Figure 3) centered in the bounding box and whose size is equal to the α% of the bounding box. The filter uses the focus area to compute the histogram representing the pixel color (HSV) frequencies in the focus. The histogram is then ordered and cut to leave the more frequent pixel colors representing the β percentage of the focus area. The selected colors are used to find and mark (i.e. black) the pixels in the bounding box as belonging to the object to be tracked, while all the other pixels are set as background (i.e. white).

At the beginning, the system is fed with such binary image representing the object (with its initial shape and position) to be followed. This image is used to train all discriminators. Note that the filtering process is repeated for each input frame in order to adapt to the dynamism of the environment conditions. When the object starts moving, the WiSARD system tries to localize the object through the discriminator responses. The higher is the response the more probable the object is in that part of the prediction window. Once the system localizes the object in the new $(\underline{i}, \underline{j})$ position (that is, discriminator $d_{\underline{i},\underline{j}}$ has the highest response), the mental model of the object is trained adding the image on the $d_{\underline{i},\underline{j}}$ discriminator to take into account the new possible shape. The position of the central retina will be set to $d_{\underline{i},\underline{j}}$. Finally, the system will evaluate the mass center of the MI that, in our case study, will be used to evaluate the tracking performance.

V. Experimental Results in Pizza Making

As a benchmark to test our approach for tracking deformable object, we adopt the *Pizza Making* case study. Despite its apparent simplicity, this task represents a very challenging test bed, both for object tracking and for robotic manipulation. Figure 4 shows a simulation of the RODYMAN robot in manipulating a pizza. Pizza is a non-rigid deformable object that can assume whatever shape we want. Hence, it is not possible to define a model for the tracking.

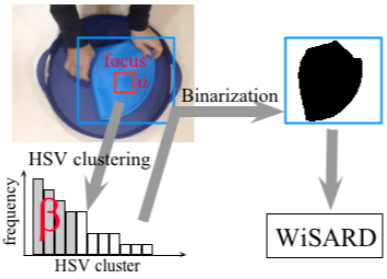

Fig. 3. The color histogram filter.

Fig. 4. Sketches of the RODYMAN robot manipulating a pizza.

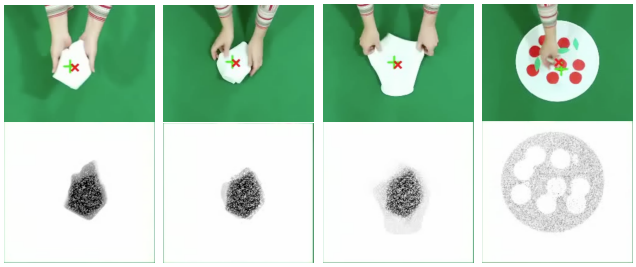

Fig. 5. Snapshots from the original video with the center of mass and relative WiSARD MIs during tracking

We individuated five different sub-tasks (see Figure 5) as follows: 1)Translation (T): the pizza is in the hands of the user who makes horizontal, vertical and circular movements; 2) Manipulation (M): the user manipulates the pizza; 3) Extension (E): the user modifies the appearance of the pizza dough in order to reach its final shape; 4) Occlusions (O): during some phases, such as manipulation or seasoning, the user may occlude the pizza on the table; 5) Seasoning and Baking (S): also seasoning ingredients may occlude the pizza dough that is now ready to be baked.

We evaluated the performance of the WiSARD considering both its ability to track the shape of the pizza and to follow the position of the object in time. As shown in Figure 5, the Mental Image (MI) of the network keeps track of the shape of the pizza during the interaction. Darkener points on the MI represents points (n-tuple of points) more frequently seen. In particular, in the third represented mental model one can see that, as soon as the user starts to enlarge the shape of the pizza dough, the MI consequently adapts its values. In order to evaluate the tracking abilities, the system evaluates the center of mass on the MI (i.e. the red cross in Figure 5). Finally, we keep track of the position of the central retina classifier (i.e the green cross in Figure 5) that is updated according to the classifiers with the best response at the previous frame.

In Table I we reported the data for 10 runs for each of the five subtasks. For each task we collected the average error of the tracking process computed as the difference of positioning of the MI center of mass, as generated by the WiSARD network, and the center of the real object from a Ground Truth (GT) evaluation. Specifically, the GT is computed by visually evaluating, frame by frame, the center of mass and selecting the correspondent points over the videos. For the experimentation, silicon pizza doughs of different colors were used. Finally, last column of Table I reports evaluation of performances of the overall task for the 10 executions. The average frame rate of the process

was about $5fps$.

	T	M	E	O	S	Overall
error (px)	5.09	2.83	3.92	3.83	4.59	4.41

TABLE I

ERRORS MEASURES ON SUB-TASKS AND THE OVERALL TASK.

We can note that, in general, the tracking process performs well its task. In particular, during translations (T), the WiSARD system is able to track the target object in all the directions with a very low positioning error. The average error in this case is $5.09px$, as showed in Table I. However, since this task is the only one involving big movements of the mass center, the positioning error is slightly bigger than in the other tasks. More similar to this task, in term of positioning error, it is the Seasoning (S) ($4.59px$ error), where the error is not due to the movements (in fact, during this phase the position of pizza is fixed) but to the occlusions that can occur during the task. Furthermore, while in the occlusion phase we evaluated quick occlusions of the pizza made by the user hand, in seasoning there are permanent occlusions (for example tomatoes and basil) that may cause a little modification in the mass center. Considering the manipulation (M), we note that when the pizza is fixed the system reaches a low tracking error of about $2.83px$, while in the case of extension (E), some occlusions occur (e.g. the human hands occlude the pizza during manipulation), and the tracking error increases a little bit ($3.92px$). Such result is comparable with the results we have with intended occlusions (O) ($3.83px$ error).

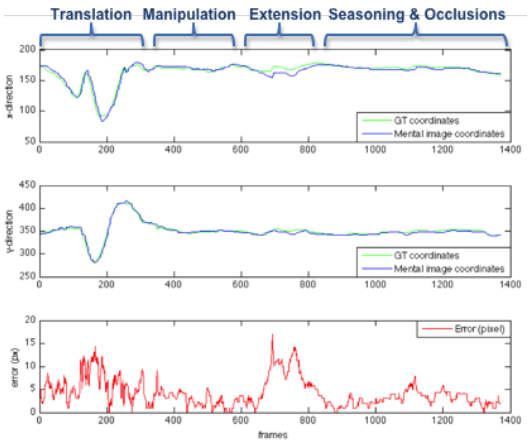

Fig. 6. Plot showing an example of the GT and MI centroid coordinates with respect to horizontal and vertical directions, and tracking error evaluated as the distance between these coordinates in the overall task.

In Figure 6 we show an example of execution of our visual servoing system in tracking the target object during the complete *pizza making* task. In the first two plots we present the trend of the GT and MI centroid coordinates respectively in the horizontal and vertical directions; while, the third plot shows the tracking error evaluated as the distance between these coordinates during the overall task.

VI. CONCLUSIONS AND FUTURE WORKS

The main contribution of this paper is to propose a methodology for object tracking in order to achieve both flexibility and robustness in tracking non-rigid deformable objects without prior-model of them. The on–line training characteristic of the proposed WiSARD network allows the robot to adapt in real–time to any new situations, such as, a new shape of the object and color and luminosity changes, and so on. The use of parallel classificators trained on the same image allows tracking the movements of the object in the space with an acceptable average error. Moreover, the reinforcing behavior of the DRASiW mechanism enables to partially cope with occlusions. The obtained results can, however, be drastically improved by parallelizing the code where possible and by optimizing the distribution of classificators in the space (e.g. a dense network near the central retina and a more sparse disposition on the neighborhood). Moreover, the adoption of more accurate filtering techniques can be investigated.

As many methods that deal with online learning of the object shape, also our approach cannot completely solve the problems of occlusion in the case the tracked object is completely occluded for a long period of time. Hence, as future work, we plan to investigate the adoption of a dead reckoning strategy to anticipate the object current/next position by using its previously determined positions. We also intend to extent the proposed method from this preliminary 2D approach to 3D and to introduce some inferences about some characteristics of the object in order to fill the gap in manipulation issues, which are not yet considered in this preliminary work.

ACKNOWLEDGMENT

This work was supported by the RODYMAN Project under the ERC Advanced Grant no. 320992.

REFERENCES

[1] I. Aleksander, M. De Gregorio, F. França, P. Lima, and H. Morton, "A brief introduction to weightless neural systems," in *ESANN09*, pp. 299–305.

[2] C. Nastar and N. Ayache, "Fast segmentation, tracking, and analysis of deformable objects," in *Proceedings of the Fourth International Conference on Computer Vision, 1993*, May 1993, pp. 275–279.

[3] J. Schulman, A. Lee, J. Ho, and P. Abbeel, "Tracking deformable objects with point clouds." in *ICRA*. IEEE, 2013, pp. 1130–1137.

[4] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, vol. 2, 2000, pp. 142–149.

[5] J. Shin, S. Kim, S. Kang, S.-W. Lee, J. Paik, B. Abidi, and M. Abidi, "Optical flow-based real-time object tracking using non-prior training active feature model," *Real-Time Imaging*, vol. 11, no. 3, pp. 204 – 218, 2005, special Issue on Video Object Processing.

[6] B. Babenko, Y. Ming-Hsuan, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Trans. on attern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1619–1632, Aug 2011.

[7] B. Willimon, I. D. Walker, and S. Birchfield, "3d non-rigid deformable surface estimation without feature correspondence." in *ICRA*. IEEE, 2013, pp. 646–651.

[8] I. Aleksander and H. Morton, *An introduction to neural computing*. London: Chapman & Hall, 1990.

[9] M. D. G. Bruno P.A. Grieco, Priscila M.V. Lima and F. M. Frana, "Producing pattern examples from mental images," in *Neurocomputing*, 2010, p. 73(79):1057 1064.