WCCI 2012 IEEE World Congress on Computational Intelligence
June, 10-15, 2012 - Brisbane, Australia IJCNN

A proposal for human action classification based on motion analysis and artificial neural networks

Thiago da Rocha
Department of Electrical Engineering
University of Brasília
Brasília, Brazil
hiagoh@unb.br

Flávio de Barros Vidal
Department of Computer Science
University of Brasília
Brasília, Brazil
fbvidal@cic.unb.br

Alexandre Ricardo Soares Romariz
Department of Electrical Engineering
University of Brasília
Brasília, Brazil
romariz@ene.unb.br

***Abstract*—This paper describes the development and application of a method for human action recognition from motion analysis in a sequence of images using an artificial neural network. The proposed method is based on two stages: Computer Vision and Computational Intelligence. The Computer Vision stage is a combination of two motion analysis techniques: Histogram of Oriented Optical Flow and Object Contour Analysis. For the Computational Intelligence stage we use a Self-Organizing Map (SOM) optimized through Learning Vector Quantization (LVQ). The approach is then applied for classification of human actions in many real situations. Testing against a database with different kinds of human actions, we show the usefulness and robustness of this method, comparing it to other proposals in the literature.**

***Index Terms*—human action recognition, histogram of oriented optical flow, object contour analysis, self-organizing map (SOM).**

I. Introduction

The technology evolution that we experienced over the last decades increased the availability of computers with high processing and storage capacity, and video cameras with high quality image capture. It made it easier to create, store and upload videos. This capacity has been increasing as equipment becomes cheaper, simpler and more portable. The increase in the overall amount of available video has set a requirement for simpler video analysis, independent of human evaluation and exhaustive searches.

Considering this scenario, the areas such as surveillance, traffic control and entertainment deal with increasingly high amounts of video information, and require the development of new methodologies and techniques for video analysis. Natural applications of automatic video analysis include: motion-based recognition, vehicle navigation, surveillance automation, pedestrian and vehicle flow monitoring, quality control in factories, video indexing and man-machine interaction [1]. Among these areas and applications the visual analysis of human motion is of fundamental importance. Gavrila [2] shows a survey about this task.

In this work we develop and test a method for recognition of human actions in sequence of images using Computer Vision and Computational Intelligence. The work is organized in seven sections, including this Introduction, and References. In Section II we describe the Computer Vision techniques, which aim to extract human actions features in the images sequence and to create numerical representations that are used in the Computational Intelligence stage. In Section III we describe Artificial Neural Networks (ANN) algorithms used to build mathematical models able to generalize the domain of knowledge (human actions in the sequence of images) represented by numerical patterns obtained in the stage of vision. Then we present in Section IV the proposed method and the process used in the classification task. In Section V experimental results are analysed. Concluding remarks are stated in Section VI.

A. Related Works

Several works have been presented aiming to propose a method to the recognition of human actions. In general, a way of extraction and representation of action and/or object in the image sequence is defined, creating patterns that are used as input to pattern recognition algorithms that classify the current action on the video. The works cited below address those issues. They all used the set of images provided by Computer Vision Laboratory of Weizmann Institute of Science [3].

Wang, Huang and Tan [4] used a compact motion representation to the recognition of human actions. After calculating the optical flows of the sequence of images through Lucas and Kanade algorithm [5], [6], the authors partitioned the optical flow in blocks, to generate a local information optical flow, which is integrated to generate the global information. For each block a standard flow histogram is generated, with 8 bins. Next, they calculate statistical characteristics of the optical flow vectors, of the shape and of the trajectory, which are integrated to generate the final representation of the motion. This representation is used to generate multi-class AdaBoost classifiers. In the Weizmann database, the authors have achieved 93.3% of hit rate.

Wang and Leckie [7] used information derived from temporal space silhouettes to represent human actions. The calculation of the silhouettes between frames is carried out through techniques for extraction of background and temporal difference or by contour trackers. As the size and position of silhouettes vary with the motion, the silhouette is centralized, with the purpose of calculating an average silhouette. To create an average silhouette library, the authors quantized the average silhouettes using the *k-means* algorithm, generating groups of average silhouettes. For each action, the authors suggest the comparison between average silhouettes and the created clusters, forming a histogram, with k bins, one bin for each cluster. The bin count is incremented each time its cluster is chosen as the closest silhouette. Finally, all histograms are compared to determine the human action class to which they belong. With this proposal, they achieved 96.8% of hit rate.

Jhuang *et al.* [8] presented a biologically-motivated solution that carries out a statistical modeling of the scene. The proposed system is a space-time feature detector hierarchy. As input to the system, a sequence of images is analyzed by an array of units sensitive to direction, which, through system hierarchy stages, become space-time feature detectors invariant to position. They extract information from local motion with a set of flow filters. Responses are locally stored, converted to high-level responses and compared to more complex templates learned from examples. Different kinds of feature detectors were used, among them space-time gradients, optical flow, and space-time orientation. In the classification stage, the authors used SVM. With the Weizmann database, they achieved 98.8% of hit rate with the gradient-based detector.

Thurau and Hlavac [9] used a method for recognizing human actions based on primitive poses. For the authors, human action recognition based on primitive poses requires three steps. The first is the localization of a person in the image. The second is the pose recognition and the third the association of the pose to the most appropriate human action class. Their proposal is focused on solving the last two problems. They extended a descriptor based on Histograms of Oriented Gradient. The action classes are represented by primitive poses histograms. These histograms are combined with temporal information to represent the image sequence, and the classification is provided by comparing these histograms, carried out through the nearest-neighbor algorithm. The authors achieved a hit rate of 94.4 %.

Chaudhry *et al.* [10] represent each frame in the video by Histograms of Oriented Optical Flow (HOOF) and the recognition of human actions was executed by classification of temporal series of HOOF. Using the nearest neighbor technique the authors achieved 94.4% of hit rate. In order, another related related works can be found in [11], [12], [13], [14], [15] and achieved 87.7%, 88.8%, 91.6%, 97.4%, and 98% of hit rate, respectively.

II. Computer Vision stage

The Computer Vision stage uses as input an image sequence. In this stage, we are looking for a set of features that provides representations of human actions along the sequence. These representations can be obtained through many techniques, but in this case, the main feature of interest to be extracted is related to the shape of motion in the image domain. Considering that different actions can generate ambiguous motion profiles, some techniques are needed to increase the robustness of this stage. For this purpose we are using motion analysis techniques combined with object contour descriptors. In the Sections below we present more details about these techniques.

A. *Optical Flow Technique*

The Optical flow is a computer vision technique commonly used to estimate the motion between frames of the image sequence. This technique estimates the motion without any *a priori* knowledge about the content of the images and it is a good approach to the image motion field. The optical flow computation calculates a velocity field for each image pixel between the previous and the current frame. In the proposed approach we used the tracking correspondence algorithm *KLT*, proposed by Lucas and Kanade [6], improved by Tomasi and Kanade [16] and extended by Shi and Tomasi [17], focused on improving the accuracy of the algorithm through a selection step called *good features to track* algorithm. In the KLT algorithm, the method of pyramidal images described in [18] was used.

B. *Histogram of Oriented Optical Flow (HOOF)*

For improved robustness of the pattern recognition in a process that involves motion features in the image domain, we developed mechanisms that minimize artifacts caused by change of the object shape and velocity in the image domain. Inspired on features histograms used for object feature recognition by Optical Flow evaluation, Chaudhry *et al.* [10] presented a method for calculating the Histogram of Oriented Optical Flow (HOOF). The HOOF technique was introduced to solve issues related to quantity of tracked points, scale changes and motion direction. After the Optical Flow estimation for all video frames, each vector flow obtained is stored in accordance to the primary angle to the horizontal axis and is weighed according to its magnitude. Therefore, every optical flow vector $\vec{v} = [u \quad v]^T$, with direction $\theta = \tan^{-1}(\frac{v}{u})$, that satisfies Equation (1) will contribute with $\sqrt{u^2+v^2}$ for the sum of bin $b, 1 \leq b \leq B$.

$$-\frac{\pi}{2} + \pi\frac{b-1}{B} \leq \theta < -\frac{\pi}{2} + \pi\frac{b}{B} \quad (1)$$

Next, the histogram is normalized to make this method invariant to scale changes. In [10], to achieve good recognition results in the conditions cited above, a minimum of 30 bins was suggested.

C. *Object Contour Analysis*

In Zhang e Lu [19] contour analysis of an object in an image is one of the most important low-level characteristics for the processing of visual information. This feature is very important to help human beings in perception tasks. For recognizing actions in sequences of images, we must extract the object motion information to help in the classification process. We list below all steps used in object contour extraction:

1) color space conversion from *RGB* to *Grayscale*;
2) evaluation of absolute difference between the frames of the input sequence (Equation 2):

$$D(x,y) = |I(x,y) - J(x,y)| \quad (2)$$

where I and J are two consecutive frames in the image sequence, x and y are the spatial coordinates in the image and D is the absolute difference between I and J;

3) binarization of the images of absolute difference (Equation 3);

$$B(x,y) = \begin{cases} 1 & \text{if } I(x,y) > \text{threshold} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where $B(x,y)$ is the binary absolute difference of the frames I and J;

4) computation of the centroid of the object motion detected in binary image;
5) sum of all binaries pixels of absolute differences and dislocated images in the images sequence.

D. Edge Contour Tracking

Also known as Edge Detections, the Edge Contour Tracking process is a technique applied in digital images to extract the image edges and extract generic information about the shape of objects (or patterns) in the image. With an evaluation of the pattern contour we can map features and use them *a posteriori* to classify it. An algorithm commonly used for contour tracking is the *Moore-Neighbor Tracing* algorithm [20], used in this work to detect the image's edge from objects in the original images sequences.

III. Computational Intelligence stage

The Computational Intelligence module receives as input the numerical representation of human actions extracted on the Computer Vision step. The goal of this module is the final recognition of human action. The Self-Organizing Map (SOM) and Learning Vector Quantization (LVQ) are used in this stage and briefly described is this section.

A SOM, proposed by Kohonen [21], consists of an array of neurons organized in arbitrary single-layer topology. The input of the net are vectors in the p-dimensional space, usually $\Re^p$. Each neuron i of layer U has an associated vector, also in space $\Re^p$, $\vec{m_i} = [m_{1i}, m_{2i}, ..., m_{pi}]$, whose elements are also called synaptic weights. The learning algorithm is based on a similarity match between inputs and synaptic vectors, and on the pre-defined topological arrangement of the network.

After the Self-Organizing (unsupervised) learning algorithm, for each point in input space, a "winner" neuron will have the largest response. From the classification of the learning examples, each winning neuron can be associated with a class. The self-organization performed by the algorithm will likely result in similar inputs being mapped in nearby neurons, thus creating a map, which can be used directly for classification.

The Learning Vector Quantization algorithm (LVQ), also proposed by Kohonen [22], can be used to improve the SOM through supervised learning. It updates the synaptic weights of the winner neuron according to its classification. If input data and winner neuron are the same class (right classification), than the weights of the synapses connected to the winner neuron are updated such that it moved towards the data point. Otherwise, the synaptic vector of the winner is moved away from the data point.

IV. Proposed Method

To recognize human actions in sequences of images, we proposed a method (Figure 1), that combines Computer Vision and Computational Intelligence techniques in two stages.

In the Computer Vision (CV) stage, we evaluate the Optical Flow and HOOF of the image sequence as previously described in Sections II-A and II-B. After we analyze the entire object contour during the motion in image domain, we obtain a numerical representation for these human actions. These numerical data are used as input for the Computational Intelligence (CI) stage. The same process generates numeric signatures during both training procedures we used (artificial neural networks with complete and reduced training sets), and during the classification of test cases. In Figure 1 all blocks have many parameters to be adjusted and in the Section V more details will be described.

A. Classification stage overview

The classification stage receive as input the decisions indicated by the Computational Intelligence stage and outputs a final classification of the human action the image sequence. This overall classification is based on the output of two neural classifiers: A first classifier uses estimates from Optical Flow and Histogram of Oriented Optical Flow (HOOF). As some gestures generate similar HOOF-based representations, a second classifier is also used, based on object contour during motion. Decisions from both classifiers are considered in the decision process, in an architecture similar to a committee machine based on Mixture of Experts [23]. Algorithm 1 was developed to determine if both neural classifiers are needed. If, from the training data, the average histograms from the HOOF representation are similar, they are grouped as a single class, and the final class disambiguation is done with the aid of the contour-based classifier. It receives as input all HOOF representations from image sequences of the training set. In the first step we calculate the average of the histograms for each class of natural action. Then we calculate the χ^2 distances between all average histograms. Those classes with distance lower than the threshold, defined at the beginning of the process, are considered similar and grouped.

The contour-based classifier, then, is used as an expert for disambiguation within groups of actions with similar HOOF-based representations. To this purpose, this classifier is trained specifically with a smaller training set, containing only the examples drawn from groups of similar actions. If the HOOF-based classification indicates an action that does not belong to the group of actions with ambiguous HOOF representation, the action will be classified just based on the optical flow profile.

V. Experimental Results

To check the performance of the proposed method, we used the database provided by the Computer Vision Laboratory of the Weizmann Institute of Science [24], that contains 83 image sequences showing nine different people, each one performing nine natural actions (1 - bend; 2 - jumping jack; 3 - jump in place; 4 - jump; 5 - run; 6 - gallop sideways; 7 - walk; 8 -

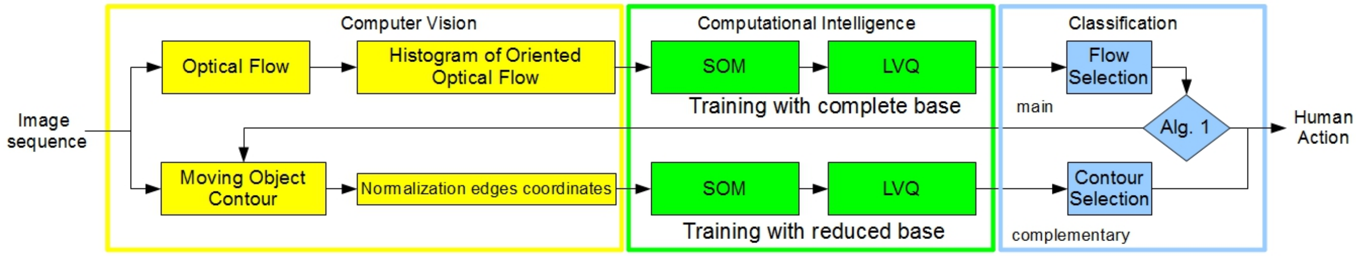


Fig. 1. Proposed Method

Algorithm 1: Algorithm to group human action classes from input training set.

Input: training set with n HOOF referring to c human action classes. The set must have been labeled with label cl for each image sequence example.
Output: Similar classes, that can be grouped.
Define a threshold l that will characterize the classes as similar, setting a real value, in the interval $[0, 1]$, to l. ;
forall the *class i (from $i = 1$ to c)* **do**
  Calculate the average histogram $\bar{h_i}$ of the class i, from all HOOF with $cl = i$, as: ;
  $\bar{h_i} = \frac{1}{N}\sum_{k=0}^{N-1} h_k$;
  where N denote the quantity of HOOF belonging to class i.
forall the *class i (from $i = 1$ to c)* **do**
  forall the *class j (from $j = 1$ to c)* **do**
    Calculate $d_{\chi^2}(\bar{h_i}, \bar{h_j})$, the χ^2 distance between the average histogram, as: ;
    $d_{\chi^2}(\bar{h_i}, \bar{h_j}) = \frac{1}{2}\sum_{b=1}^{B} \frac{|\bar{h_{1;b}} - \bar{h_{2;b}}|^2}{\bar{h_{1;b}} + \bar{h_{2;b}}}$;
    if $d_{\chi^2}(\bar{h_i}, \bar{h_j}) < l$ **then**
      Include the class i e j as candidates to grouping;

---

one-hand wave; 9 - two-hands wave). Figure 2 shows samples of some humans actions that can be founded in [24].

Through Algorithm 1 (Section IV-A), with the variable l set to 0.07, a clustering of classes 8 and 9, as well as 4 and 5, was obtained, indicating that these classes have similar optical flow profile and the classification process using only the optical flow technique would not be successful. In this case it was necessary to use features collected from the object contour analysis step.

To validate the results we used the complete database previously described, and performed the leave-one-out cross validation (LOOCV) method [25].

In Tables I, III and IV we show the parameters used in the execution of the algorithms KLT, SOM and LVQ, respectively, and in Table II the parameters for the object contour feature extraction, when it was required. All these variables were adjusted empirically.

Figure 3 depicts classification error with varying bin size for the HOOF, ranging from 5 to 100. Best results were obtained with 15 bins (3.6% error) and 60 bins (4.8% error).

Increasing the number of bins results in slower training for the ANN, due to the higher dimension of the inputs. By varying the number of bins, we did not find a significant

TABLE I
PARAMETERS USED TO EVALUATE THE OPTICAL FLOW THROUGH THE KLT ALGORITHM.

Parameter	Value
Pyramidal images	with five levels
Rectangular neighborhood window	5×5 pixels
Iterations of the optical flow evaluation	up to 20
Good features to track	up to 40.000
Eigenvalue threshold to point inclusion	set at 0.01
Minimal distance between points	set at 0.01

TABLE II
PARAMETERS USED WHEN THE OBJECT CONTOUR FEATURE EXTRACTION WAS REQUIRED.

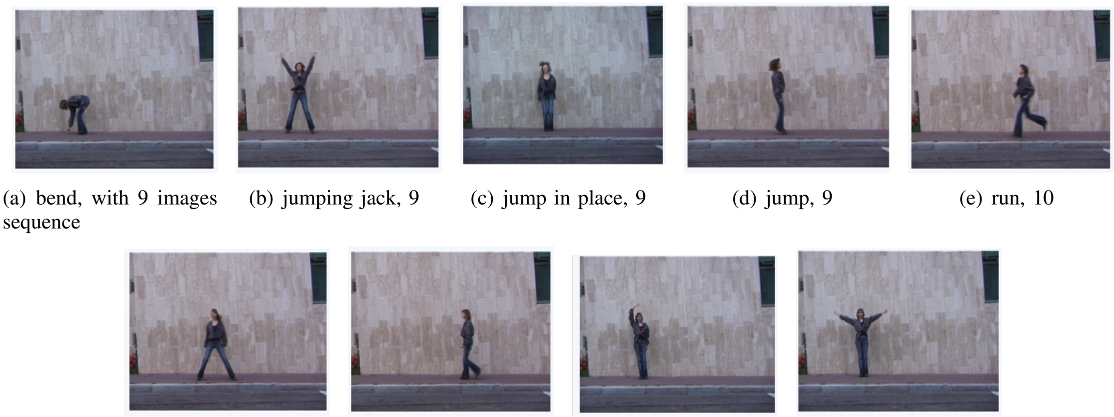
Parameter	Value
Threshold for binary image Conversion (step 3 of Method in section II-C)	equal to 0.1
Edge detection algorithm	Moore-Neighbor Tracing algorithm
Edge detection stopping criteria	Jacob's stopping criteria
Connectivity	8-connectivity
Standard edges	with 100 points

TABLE III
PARAMETERS USED TO EXECUTE THE SOM ALGORITHM.

Parameter	Value
Iterations of the SOM training	up to 100
Initial learning rate	set at 0.9
Neighborhood function	Gaussian function
Distance	Euclidean
Topology map	rectangular, with 9×9 neurons, to the HOOF analysis. rectangular, 4×4 neurons, to the contour analysis.
Order input data	random order
Input data normalization	not performed, to the HOOF analysis. performed, to the contour analysis.
Initial weights	randomly initialized (between 0 and 1)

TABLE IV
PARAMETERS USED TO EXECUTE THE LVQ ALGORITHM.

Parameter	Value
Iterations of theLVQ training	up to 100
Initial learning rate	set at 0.3
Distance	Euclidean
Order input data	random order
Initial weights	obtained from SOM
Input data normalization	not performed, to the HOOF analysis. performed, to the contour analysis.



(a) bend, with 9 images sequence (b) jumping jack, 9 (c) jump in place, 9 (d) jump, 9 (e) run, 10

(f) gallop sideways, 9 (g) walk, 10 (h) one-hand wave, 9 (i) two-hands wave, 9

Fig. 2. Natural actions performed in the database provided by Computer Vision Laboratory of Weizmann Institute of Science.

improvement over 15 bins, and thus we chose that number in all further experiments. Table V and VI show virtually identical confusion matrix on the final classification when the HOOF algorithms operates with 15-bin histograms and with 60-bin histograms. Moreover, we note that most of the experiments achieved error values smaller than 10.0 % error rate. Table VII shows a comparison between our results and those found in the literature (section I-A).

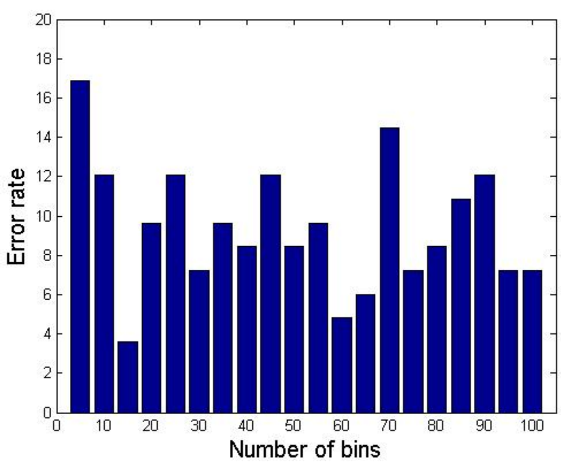


Fig. 3. Experimental results using the techniques HOOF and standard edges coordinates, with the numbers of bins varying from 5 to 100 and the number of points on the standard edges equal 100. The results were validated with LOOCV method.

Wang *et al.* [4] also used optical flow for representation, which was complemented with statistical informations about optical flow, contour and trajectory path of the object during the motion. The main advantage of our proposed method is a smaller computational effort. While we used only three neural classifiers, Wang *et al.* used classifiers created from the multi-class AdaBoost algorithm. This requires training several weak classifiers, that together reach a improved final result. Thurau and Hlavac [9] used a technique that requires pre-processing to locate the region that contains a person. Viola and Jones

TABLE V

CONFUSION MATRIX FOR THE CLASSIFICATION USING HOOF WITH 15-BIN HISTOGRAMS.

	bend	jumping jack	jumping in place	jump	run	gallop sideways	walk	one-hand wave	two-hands wave
bend	1.0								
jumping jack		1.0							
jumping in place			1.0						
jump				.89				.11	
run					1.0				
gallop sideways						.89	.11		
walk						.10	.90		
one-hand wave								1.0	
two-hands wave									1.0

TABLE VI

CONFUSION MATRIX FOR THE CLASSIFICATION USING HOOF WITH 60-BIN HISTOGRAMS.

	bend	jumping jack	jumping in place	jump	run	gallop sideways	walk	one-hand wave	two-hands wave
bend	1.0								
jumping jack		1.0							
jumping in place			1.0						
jump				.89				.11	
run					.90	.10			
gallop sideways						.89	.11		
walk						.10	.90		
one-hand wave								1.0	
two-hands wave									1.0

[26] present a method for such pre-processing, not required in our approach.

Chaudhry *et al.* [10] achieved slightly worse results than ours. Their method also does not need a preprocessing step. Including the object contour analysis step we improved the results, especially in cases when distinct actions were represented by ambiguous HOOF, such in the classes 8 and 9.

TABLE VII
CLASSIFICATION PERFORMANCE COMPARISON BETWEEN OUR METHOD AND ANOTHER WORKS WHO USED THE WEIZMANN DATABASE.

Method	Hit rate (%)
Hoai *et al.* [11]	87.7
Huang and Wu [12]	88.8
Lassoued *et al.* [13]	91.6
Wang, Huang and Tan [4]	93.3
Thurau and Hlavac [9]	94.4
Chaudhry *et al.* [10]	94.4
Our method (HOOF with 60 bins)	**95.2**
Our method (HOOF with 15 bins)	**96.4**
Wang and Leckie [7]	96.8
Huang *et al.*[14]	97.4
Masood *et al.*[15]	98.0
Jhuang *et al.*[8]	98.8

In the Wang and Leckie [7] method we note some misclassification when trying to classify the actions one-hand wave and two-hands wave. Following our proposed method these classes are grouped in the optical flow profile analysis step and then we generated a expert ANN able to classify it using standard edges coordinates representation and reaching 100% of hit rate.

VI. CONCLUSIONS AND FUTURE WORKS

This work presented a method for human action classification from a sequence of images. The approach is based on two stages, that involves a combination of Computer Vision and Computational Intelligence techniques. The kernel of Computer Vision stage is based on Histogram of Oriented Optical Flow (HOOF) technique and object contour analysis when ambiguous humans actions profiles are found.

The processed data is fed to a self-organized map (SOM), optimized through Learning Vector Quantization (LVQ) to perform the final classification decision. An algorithm was also developed to automatically determine which classes require a separate object-contour analysis to disambiguation.

We note that the method achieved acceptable error rates and compatible with those already published in the literature. The results showed its robustness in a database with several kinds of human action. The tests of variation in the quantity of bins showed that it is not necessary to increase this quantity beyond 15 and 60, in our case, to get better results. Good results were obtained despite the absence of preprocessing stages for background extraction or subject location.

We attribute the success of this approach to the integration of different techniques to solve a complex problem. This still requires, however, an evaluation of real-time capability. An immediate proposal for future work is the implementation of our algorithm in a high-level language and its operation in real time for embedded systems.

REFERENCES

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.*, vol. 38, December 2006.

[2] D. M. Gavrila, "The visual analysis of human movement: a survey," *Comput. Vis. Image Underst.*, vol. 73, pp. 82–98, January 1999.

[3] "Computer vision group - weizmann institute," http://www.wisdom.weizmann.ac.il/~vision/, january 2012.

[4] S. Wang, K. Huang, and T. Tan, "A compact optical flowbased motion representation for real-time action recognition in surveillance scenes," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, nov. 2009, pp. 1121 –1124.

[5] B. Lucas, "Generalized image matching by the method of differences," Ph.D. dissertation, Robotics Institute, Carnegie Mellon University, July 1984.

[6] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the 7th international joint conference on Artificial intelligence - Volume 2*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1981, pp. 674–679.

[7] L. Wang and C. Leckie, "Encoding actions via quantized vocabulary of averaged silhouettes," in *Pattern Recognition (ICPR), 2010 20th International Conference on*, aug. 2010, pp. 3657 –3660.

[8] H. Jhuang, T. Serre, L. Wolf, and T. Poggio, "A biologically inspired system for action recognition," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, oct. 2007, pp. 1 –8.

[9] C. Thurau and V. Hlavac, "Pose primitive based human action recognition in videos or still images," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, june 2008, pp. 1 –8.

[10] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal, "Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, june 2009, pp. 1932 –1939.

[11] M. Hoai, Z.-Z. Lan, and F. De la Torre, "Joint segmentation and classification of human actions in video," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, june 2011, pp. 3265 –3272.

[12] W. Huang and Q. Wu, "Human action recognition based on self organizing map," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, march 2010, pp. 2130 –2133.

[13] I. Lassoued, E. Zagrouba, and Y. Chahir, "Video action classification: A new approach combining spatio-temporal krawtchouk moments and laplacian eigenmaps," in *Signal-Image Technology and Internet-Based Systems (SITIS), 2011 Seventh International Conference on*, 28 2011-dec. 1 2011, pp. 291 –297.

[14] L. Huang, X. gan Chen, Z. Gao, and H. Liu, "Human action recognition by imitating the simple cells of visual cortex," in *Intelligent Computation and Bio-Medical Instrumentation (ICBMI), 2011 International Conference on*, dec. 2011, pp. 313 –320.

[15] S. Masood, A. Nagaraja, N. Khan, J. Zhu, and M. Tappen, "Correcting cuboid corruption for action recognition in complex environment," in *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, nov. 2011, pp. 1540 –1547.

[16] C. Tomasi and T. Kanade, "Detection and tracking of point features," International Journal of Computer Vision, Tech. Rep., 1991.

[17] J. Shi and C. Tomasi, "Good features to track," in *1994 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94)*, 1994, pp. 593 – 600.

[18] J. Bouguet, "Pyramidal implementation of the lucas kanade feature tracker: Description of the algorithm," 2002.

[19] D. Zhang and G. Lu, "A comparative study of fourier descriptors for shape representation and retrieval," in *Proc. of 5th Asian Conference on Computer Vision (ACCV)*. Springer, 2002, pp. 646–651.

[20] A. Ghunein, "Contour tracing," http://www.imageprocessingplace.com/downloads_V3/root_downloads/tutorials/contour_tracing_Abeer_George_Ghuneim/index.html, june 2011.

[21] T. Kohonen, "The self-organizing map," *Proceedings of the IEEE*, vol. 78, September 1990.

[22] ——, "Learning vector quantization for pattern recognition," Helsinki University of Technology, Finland, Technical Report TKK-F-A601, 1986.

[23] S. Haykin, *Neural Networks: a comprehensive foundation*, 2nd ed. Prentice Hall, Jul. 1999.

[24] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *The Tenth IEEE International Conference on Computer Vision (ICCV'05)*, 2005, pp. 1395–1402.

[25] S. Arlot and A. Celisse, "A survey of cross-validation procedures for model selection," *Statistics Surveys*, vol. 4, pp. 40–79, 2010.

[26] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *Proc. CVPR*, vol. 1, pp. 511–518, 2001.