Pedestrian Detection using Infrared images and Histograms of Oriented Gradients

F. Suard, A. Rakotomamonjy, A. Bensrhair
Lab. PSI CNRS FRE 2645
INSA Rouen
avenue de l'université, 76800 Saint Etienne du Rouvray
France
email: frederic.suard@insa-rouen.fr

A. Broggi
Dipartimento di Ingegneria dell'Informazione,
Università di Parma, Parco Area delle Scienze 181A,
I-43100 Parma, Italy
email : broggi@ce.unipr.it

Abstract— **This paper presents a complete method for pedestrian detection applied to infrared images. First, we study an image descriptor based on histograms of oriented gradients (HOG), associated with a Support Vector Machine (SVM) classifier and evaluate its efficiency. After having tuned the HOG descriptor and the classifier, we include this method in a complete system, which deals with stereo infrared images. This approach gives good results for window classification, and a preliminary test applied on a video sequence proves that this approach is very promising.**

I. Introduction

Since the last few years now, the development of driving assistance systems has been very active in order to increase the vehicle and its environment safety. At the present time, the main objective in this domain is to provide the drivers with some information concerning its environment and any potential hazard. One among all useful information is the detection and localization of a pedestrian in front of a vehicle.

This problem of detecting pedestrians is a very difficult problem that has essentially been addressed using vision sensors, image processing and pattern recognition techniques. In particular, detecting pedestrians thanks to images is a complex challenge due to their appearance and pose variability. In the context of daylight vision, several approaches have been proposed and are based on different image processing techniques or machine learning [9], [5], [12].

Recently, owing to the development of low-cost infrared cameras, night vision systems have gained more and more interest, thus increasing the need of automatic detection of pedestrians at night. This problem of detecting pedestrians from infrared images has been investigated by various research teams in the last years. The main methodology is based on extracting cues (symmetry, shape-independent features, ...), pedestrian templates from images and then using these features to perform detection [8], [1], [6].

This paper addresses the problem of detecting pedestrian from infrared images. The proposed approach is based on shape-based cues and a machine learning technique that learns to recognize a pedestrian.

Recent works have shown that efficient and robust shape-based cues can be obtained from histogram of oriented gradient (HOG) in images [7]. For instance, Shashua et al. [10] has developed a complete system for pedestrian detection with monocular acquisition system. Its one-frame classification method is based on a description of images with histograms of gradients, computed over a determined number of regions according to a mask of distribution. Recently, Dalal and Triggs have further developed this idea of histogram of gradient and have achieved excellent recognition rate of human detection in images [4].

In this paper, we introduce a complete pedestrian detection system, applied to infrared images. At first, we propose a single frame pedestrian detection system which follows the path of Shashua and al. and Dalal and al. This detection system is based on histogram of gradients combined with Support Vector Machines for the recognition stage. It has been developed to detect a pedestrian centered in a 128×64 single image. The paper provides a comprehensive study of this system parameters in order to point out its best setting. Then we propose a complete detection system based on a focus of attention approach. This complete system is then able to detect any scale of pedestrians in a large size image.

The paper is organised as follows. In section II-A, we describe the single frame detector and we give details for the HOG descriptor and its parameters. Then, we propose our method to scan a complete image and to detect pedestrians. The results section gives a study of the parameters setting of the HOG descriptor and also presents some performances of the full system. Conclusions and perspectives are presented in the final section.

II. Overview of the method

A. *Histogram of Oriented Gradients based Detector*

In the context of object recognition, the use of edge orientation histogram has gain popularity [10], [4]. However, the concept of dense and local histograms of oriented gradients (HOG) is a method introduced by Dalal et al.[4]. The aim of such method is to describe an image by a set of local histograms. These histograms count occurences of gradient orientation in a local part of the image. In this work, in order to obtain a complete descriptor of an infrared image, we have computed such local histograms of gradient according to the following steps :

1) compute gradients of the image,
2) build histogram of orientation for each cell,
3) normalize histograms within each block of cells.

The following paragraphs give more details on each of these steps.

1) Gradient computation: The gradient of an image has been simply obtained by filtering it with two one-dimensional filters :

- horizontal : $\begin{pmatrix} -1 & 0 & 1 \end{pmatrix}$
- vertical : $\begin{pmatrix} -1 & 0 & 1 \end{pmatrix}^T$

An example of gradient is shown in figure 1. Gradient could be signed or unsigned. This last case is justified by the fact that the direction of the contrast has no importance. In other words, we would have the same results with a white object placed on a black background, compared with a black object placed on a white background. In our case, we have considered unsigned gradient which values going from 0 to π.

The next step is orientation binning, that is to say to compute the histogram of orientation. One histogram is computed for each cell according to the number of bins.

2) Cell and block descriptors: The particularity of this method is to split the image into different cells. A cell can be defined as a spatial region like a square with a predefined size in pixels. For each cell, we then compute the histogram of gradients by accumulating votes into bins for each orientation. Votes can be weighted by the magnitude of a gradient, so that histogram takes into account the importance of gradient at a given point. This can be justified by the fact that a gradient orientation around an edge should be more significant than the one of a point in a nearly uniform region. Examples of histograms of the square region given in the middle image of figure 1 is shown in figure 2. As expected, the larger the number of bins, the more detailed the histogram is.

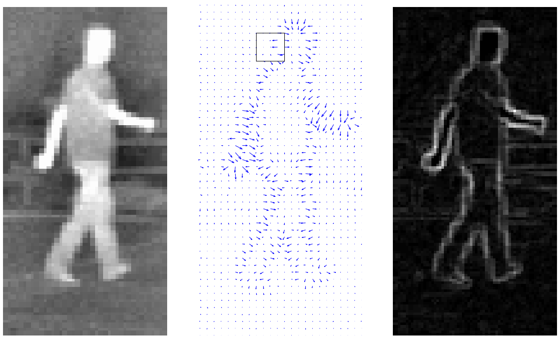


Fig. 1. This figure shows the gradient computation of an image. (left) is the original image, (middle) shows the direction of the gradient, (right) depicts the original image according to the gradient norm.

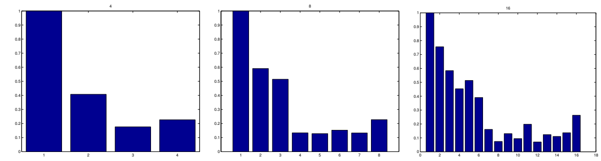


Fig. 2. This figure shows the histograms of gradient orientation for (left) 4 bins, (middle) 8 bins (right) 16 bins.

When all histograms have been computed for each cell, we can build the descriptor vector of an image concatenating all histograms in a single vector.

However, due to the variability in the images, it is necessary to normalize cells histograms. Cells histograms are locally normalized, according to the values of the neighboured cells histograms. The normalization is done among a group of cells, which is called a block.

A normalization factor is then computed over the block and all histograms within this block are normalized according to this normalization factor. Once this normalization step has been performed, all the histograms can be concatenated in a single feature vector.

Different normalization schemes are possible for a vector V containing all histograms of a given block. The normalization factor nf could be obtained along these schemes :

- none : no normalization applied on the cells, $nf = 1$.
- L1-norm : $nf = \frac{V}{\|V\|_1 + \varepsilon}$
- L2-norm : $nf = \frac{V}{\sqrt{\|V\|_2^2 + \varepsilon^2}}$

ε is a small regularization constant. It is needed as we sometime evaluate empty gradients. The value of ε has no influence on the results.

Note that according to how each block has been built, a histogram from a given cell can be involved in several block normalization. In thus case, the final feature vector contains some redundant informations which have been normalized in a different way. This is especially the case if blocks of cells have overlapping.

B. SVM Classifier

As we have stated in the introduction, the recognition system is based on a supervised learning technique. Hence, we have used a set of training image examples with and without pedestrians, and described by their HOG, to learn a decision function. In our case, we have used a Support Vector Machines classifier.

The Support Vector Machines classifier is a binary classifier algorithm that looks for an optimal hyperplane as a decision function in a high-dimensional space [2], [11], [3]. Thus, consider one has a training data set $\{\mathbf{x}_k, y_k\} \in \mathcal{X} \times \{-1, 1\}$ where $\mathbf{x}_k$ are the training examples HOG feature vector and y_k the class label. At first, the method consists in mapping $\mathbf{x}_k$ in a high dimensional space owing to a function Φ. Then, it looks for a decision function of the form : $f(\mathbf{x}) = \mathbf{w} \cdot \Phi(\mathbf{x}) + b$ and $f(\mathbf{x})$ is optimal in the sense that it maximizes the distance between the nearest point $\Phi(\mathbf{x}_i)$ and the hyperplane. The class label of $\mathbf{x}$ is then obtained by considering the sign of $f(\mathbf{x})$. This optimization problem can be turned, in the case of L_1 soft-margin SVM classifier (misclassified examples are linearly penalized), in this following way :

$$\min_{\mathbf{w},\xi} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{k=1}^{m} \xi_k \tag{1}$$

under the constraint $\forall k, \quad y_k f(\mathbf{x}_k) \geq 1 - \xi_k$. The solution of this problem is obtained using the Lagrangian theory and it is possible to show that the vector $\mathbf{w}$ is of the form :

$$\mathbf{w} = \sum_{k=1}^{m} \alpha_k^* y_k \Phi(\mathbf{x}_k) \tag{2}$$

where α_i^* is the solution of the following quadratic optimization problem :

$$\max_{\alpha} W(\alpha) = \sum_{k=1}^{m} \alpha_k - \frac{1}{2}\sum_{k,\ell}^{m} \alpha_k \alpha_\ell y_k y_\ell K(\mathbf{x}_k, \mathbf{x}_\ell) \tag{3}$$

subject to $\sum_{k=1}^{m} y_k \alpha_k = 0$ and $\forall k, 0 \leq \alpha_k \leq C$, where $K(\mathbf{x}_k, \mathbf{x}_\ell) = \langle \Phi(\mathbf{x}_k), \Phi(\mathbf{x}_\ell) \rangle$. According to equation (2) and (3), the solution of the SVM problem depends only on the Gram matrix K.

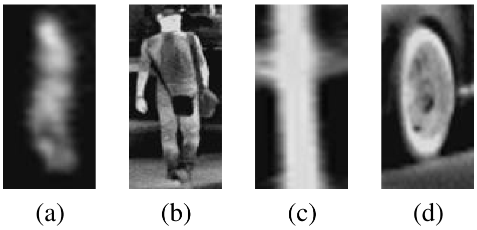


Fig. 3. This figure shows some examples of images in the learning set. (a) and (b) are pedestrians, (c) and (d) are non-pedestrians but are potential objects that could be detected in the image.

III. Setting parameters

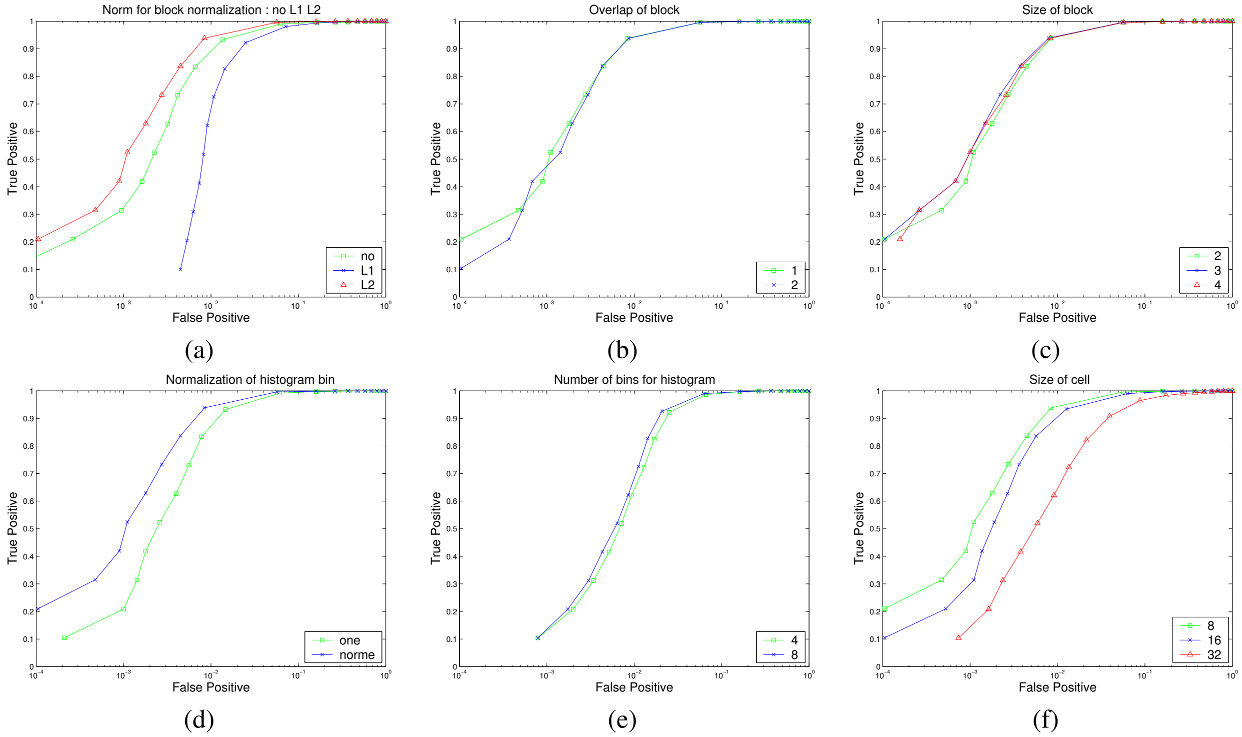
In this section, we will describe a method to choose the optimal parameters for the HOG descriptors. As we have seen in section II-A, HOG descriptors involve many parameters concerning the cells, blocks, or cells histograms that need to be treated.

- Cell
  - size of the cell, that is to say the number of pixels contained in a cell.
- Blocks
  - size : number of cells contained in a block,
  - shift : number of cells overlapped by block,
  - norm : normalization scheme.
- Histogram
  - number of bins,
  - sign : gradient signed or unsigned,
  - weighting vote method.

To evaluate the most efficient set of parameters, we have set up a complete test. This test has been realised with 4400 infrared images with a size of 128×64 pixels : 2200 pedestrians, and 2200 non-pedestrians. Figure 3 shows some examples of images used for learning. These images are obtained by selecting manually in original images different boxes containing a pedestrian or any kind of object. Images are then resized to comply with the requested size of 128×64 pixels.

We tested a large variety set of parameters :

- Size of cell : 4×4, 8×8 or 16×16 pixels,
- size of block : 1×1 , 2×2 or 4×4 cells,
- overlap of block : 1, 2,
- number of bins for histogram : 4,8 or 16,
- vote method for histogram : weigthed with gradient magnitude or no,
- normalization factor for block : no, L1 or L2,



(a) (b) (c)

(d) (e) (f)

Fig. 4. This figure shows main results obtained for different set of HOG parameters. All figure have been obtained for a 2-classes linear svm with 100 elements for learning. For HOG descriptor, here are the default parameters that have been retained : size of block=2, number of bins = 4, size of cell = 8, overlap of blocks = 1, adding values in histogram = normalized, normalization factor for block = L2. (a), (b) and (c) shows results for block parameters. (d) and (e) shows parameters for histogram parameters. (f) shows the cell parameter.

To complete the test, we also tested different parameters for SVM classifier :

- size of learning set : 10, 100, 1000 object per class,
- weight for misclassified points C : 0.01, 1, 100.

First, we compute a dataset for a given HOG set of parameters. Then we evaluate its efficiency with the classifier. The classifier was run 10 times on different combination of data for learning and test. It should be noticed that all combinations have been fixed at the beginning of the test, and for different sets of parameters, we took the same elements for classification.

We present here some results of our test. Results in figure 4 highlights the parameters setting. All results are given with respect to default parameters which are :

- size of block=2,
- number of bins = 4,
- size of cell = 8,
- overlap of blocks = 1,
- adding values in histogram = normalized,
- normalization factor for block = L2.

Figure 4 shows different results obtained for setting HOG parameters. We can see that some parameters are increasing performance significantly, like block factor normalization or cell size. On the other hand, some parameters are less significant but participate also to the global performance.

We deduced the optimal set of parameters :

- size of block=2,
- number of bins = 8,
- size of cell = 8,
- overlap of blocks = 1,
- adding values in histogram = normalized,
- normalization factor for block = L2.

A result should be pointed out. Graphic 4-(f) seems to be better for a shortest size of cell. Indeed, results are better for a size equal to 4, but with these parameters, size of HOG descriptor becomes too large for our machine and the test could not be run.

		True	
		P	N
Prediction	P	2096	54
	N	71	2079

detection	0.9749
accuracy	0.9709
precision	0.9672

Fig. 5. Confusion matrix obtained with a learning set of 1000 examples, tested on 4400 examples.

In fact, that size of a vector varies for 128 up to 100000, depending on parameters. With a small vector, computation of HOG descriptor is fast and does not require a lot of memory. In the contrary, largest vector requires more time, but detection rate is higher. In pratice, a compromise is done between time computation and high detection rate.

IV. Results

A. *Windows classifier*

Here are some results for the single windows classifier. We use the optimal HOG parameter set that have been found in section III. We test 3 sizes for the learning set : 10, 100 and 1000 for each class. The total number of images is 2200 positive and 2200 negative examples. For each test, we use the given learning set, and test the classifier with all other images.

Results have been averaged over 10 trials with random splitting of learning and testing data. This random splitting has been performed prior to parameter testing so that results are comparable.

Figure 6 presents some results of ROC Curve obtained with the classifier. An example of Confusion Matrix obtained during our test is shown on figure 5.

The ROC curve enables us to compare different result obtained for the prediction function $f(x)$ when $f(x) > \theta, \theta \in \mathbb{R}$. For a high value of θ, false prediction are rejected. At the contrary, when θ is low, the classifier becomes more permissive and some misclassification appear.

As we can see on figure 6, with 1000 examples in the learning set, for 90% of detection rate, we have one false alarm for 330 computed images. The accuracy obtained is up to 99%.

As shown on figure 6, size of learning set is an important parameter. It clearly shows that when the learning set covers the largest variety of pedestrians, the recognition is easier. But it should be noticed that, even for 100 pedestrians in the learning set, detection rate is already good. Concerning the weight of misclassified C, we have tested some different values (0.01, 1 and 100), but this change has little effects on the results.

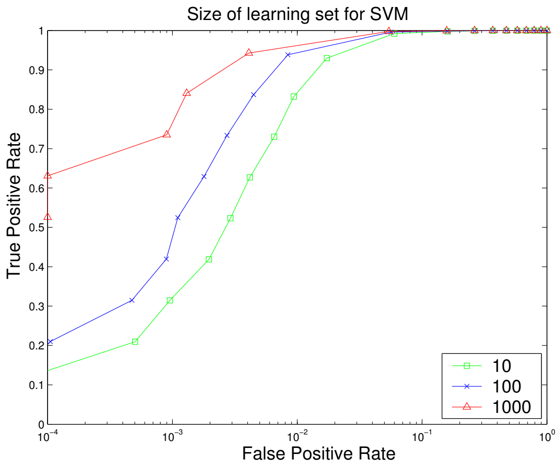


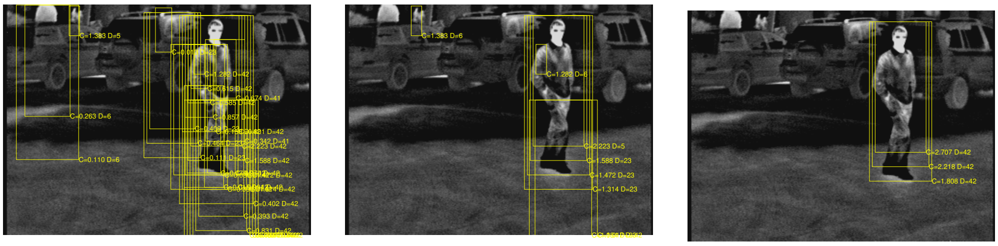
Fig. 6. This figure shows the ROC Curve of the classifier when the size of the learning set varies.

Now we will present some preliminary results for the complete system. We test the system on a video sequence containing infrared stereo images. Note that the sequence is completely different with the sequence used during the HOG test.

We tested a two-classes SVM, with a learning set of 100 pedestrians, and 100 non-pedestrians. These examples are extracted from the current video sequence, as well as the test examples.

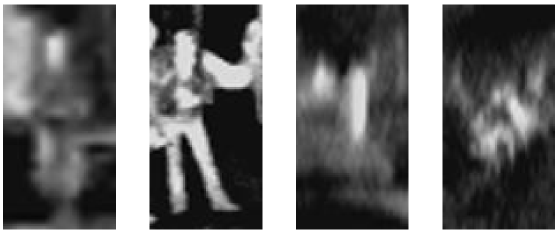
Figure 7 is an example of results. Usually, we consider the sign of the prediction $f(x)$ to classify the object x (see sectionII-B) . The prediction value can be interprated as a distance with the margin. If the distance is over 0, it means that this is near the pedestrian class, but could be rejected according with the ambiguity of the prediction. So, if we want to keep only windows which represents a pedestrian with strong confidence, we can set a threshold for the prediction rate $fx) > \theta$. Figure 7 shows clearly that when the threshold θ is higher, we have less false prediction or ambiguity. If we come back to the ROC Curve (6), it means that when θ is high, the ratio between good classification and misclassification is high.

On figure 8, we can see some misclassified objects examples. The reasons why our system fails can be explained as follows: generally, misclassification is either due to the poor quality of images, since the camera definition is only 320 × 240 pixels or due to the pedestrian location in the image.



(a) (b) (c)

Fig. 7. This figure shows the pedestrian detection. The threshold prediction value for (a) is 0, 1 for (b) 1.5 for (c). C is the prediction rate, D is the disparity of the window.



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

Fig. 8. Examples of misclassified pedestrians (a and b) and non-pedestrians (c and d).

a - pedestrian is blurred, so it does not fit exactly a pedestrian shape.
b - pedestrian is not centered.
c,d- non-pedestrians are confused with a small pedestrian.

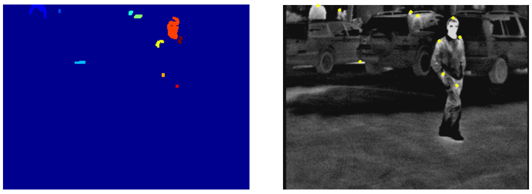
V. Complete system

In this part, we will describe a proposal for a complete system.

In the section II-A, we have studied a classification method for a single window. Now, a complete system is implemented to use this classifier for an image of a scene, that is to say containing many objects, which of them could be pedestrians. The HOG descriptor enables us to caracterize a window with a feature vector. The brute method would be to test all possible windows in the given image, in order to be the most exhaustive. But we could easily conclude that the number of windows becomes rapidly too large, and the large majority of the scan is useless. Our aim is now to select potential windows of the image, that could contained a pedestrian.

Our application concerns FIR images : infrared images. One specificity of this kind of images, is that warm objects appears lighter than cold objects which are dark. We propose to use FIR images during night, so a pedestrian appears lighter than its environment.

One way to extract potential areas of the scene is to look at each area whose pixel values are above a defined threshold. For each area, we extract some windows around this area, resize it at 128×64 pixels, compute HOG descriptor for each windows and classify vectors. Figure 9 shows an example of potential areas detected in an image.



(a) (b)

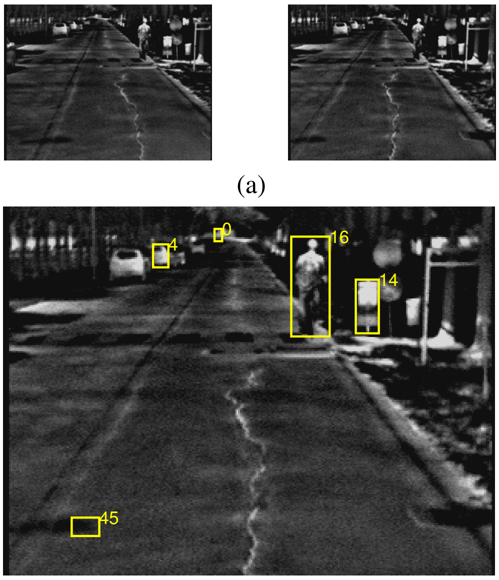
Fig. 9. This figure shows points for potential pedestrian location

The proposed system is made of two infrared cameras, that is to say stereovision. For the moment, we use only one image (right or left) to detect pedestrians in the observed scene. Using stereovision provides us information concerning the pedestrian position. To obtain the depth information we simply compute the disparity for the detected frames which are containing a pedestrian. Figure 10 shows an example of disparity computation, for some windows in the image.

In the future, we propose to run with another advantage given by stereovision. Since we could dispose of a second image, we could reinforce the detection obtained on the first image, with a second detection. This point could help us to reduce the false alarm rate.

VI. Conclusion

We have presented a new method for pedestrian detection using infrared images. The main characteristic of this



(a)

(b)

Fig. 10. This figures shows an example of disparity computation. (a) shows the right and left images, (b) shows result disparity for some windows.

method is its single frame based classification method. Indeed, the classifier deals with a 128 × 64 window containing a single object. From this window, we have extracted a feature vector composed of local histograms of oriented gradients. Combined with a SVM classifier, such system yields to very good results for single frame performance. We have integrated this classifier into a complete system of pedestrian recognition, using an infrared stereovision system. In FIR images, a pedestrian has some caracterics which help us to localize all potential pedestrians in the scene. Then, we look precisely through a sliding window if the image contains or not a pedestrian. If a pedestrian is found, we add another functionality, with help of the stereovision, to locate in real world the position of the pedestrian.

Results are very encouraging, but there is still some perspectives for our future search. Firstly, we will develop a coarse-to-fine approach to localize pedestrians in large images. Furthermore, we plan to enhance the performance of the global system by developing a multiple classifier system, where each classifier is devoted to a given pedestrian pose. Besides, when dealing with image sequences, motion information can be used for still improving the detection performance.

VII. Acknowledgement

We would like to thank Mike Del Rose (TACOM) for the availability of the FIR cameras.

This work was supported in part by the IST Program of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors views.

References

[1] Massimo Bertozzi, Alberto Broggi, Alessandra Fascioli, Thorsten Graf, and Marc-Michael Meinecke. Pedestrian detection for driver assistance using multiresolution infrared vision. *IEEE Trans. on Vehicular Technology*, 53(6):1666–1678, nov 2004. ISSN 0018-9545.

[2] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In D. Haussler, editor, *5th Annual ACM Workshop on COLT*, pages 144–152, Pittsburgh, PA, 1992. ACM Press.

[3] N. Cristianini and J. Shawe-Taylor. *Introduction to Support Vector Machines*. Cambridge Univeristy Press, 2000.

[4] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In Cordelia Schmid, Stefano Soatto, and Carlo Tomasi, editors, *International Conference on Computer Vision and Pattern Recognition*, volume 2, pages 886–893, INRIA Rhone-Alpes, ZIRST-655, av. de l'Europe, Montbonnot-38334, June 2005.

[5] D. Gavrila and J. Geibel. Shape- based pedestrian detection and tracking. In *Proceedings of IEEE Intelligent Vehicles Symposium*, pages 215–220, 2000.

[6] Y. Fang K. Yamada Y. Ninomiya B.K.P. Horn and I. Masaki. A shape-independent method for pedestrian detection with far-infrared images. 53(5), September 2004.

[7] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[8] A. Broggi A. Fascioli P. Grisleri T. Graf M. Meinecke. Model-based validation approaches and matching techniques for automotive vision based pedestrian detection. In *Intl. IEEE Wks. on Object Tracking and Classification in and Beyond the Visible Spectrum, San Diego, USA*, page in press, June 2005.

[9] C. Papageorgiou and T. Poggio. Trainable pedestrian detection. In *Proceedings of the 1999 International Conference on Image Processing*, pages 35–39, 1999.

[10] A. Shashua, Y. Gdalyahu, and G. Hayon. Pedestrian detection for driving assistance systems: Single-frame classification and system level performance. In *Proceedings of IEEE Intelligent Vehicles Symposium*, 2004.

[11] V. Vapnik. *Statistical Learning Theory*. Wiley, 1998.

[12] P. Viola, M. Jones, and D. Snow. Pedetrian using patterns of motions and appearance. In *IEEE Int. Conf on Computer Vision*, pages 734–741, 2003.

[13] F. Xu and K. Fujimura. Pedestrian detection and tracking with night vision. In *IEEE Intelligent Vehicles Symposium*, 2002.