

An Approach to Vision-Based Person Detection in Robotic Applications

Carlos Castillo and Carolina Chang

Grupo de Inteligencia Artificial
Universidad Simón Bolívar
Caracas 1080, Venezuela

carlos@gia.usb.ve, cchang@ldc.usb.ve

Abstract. We present an approach to vision-based person detection in robotic applications that integrates top down template matching with bottom up classifiers. We detect components of the human silhouette, such as torso and legs; this approach provides greater invariance than monolithic methods to the wide variety of poses a person can be in. We detect borders on each image, then apply a distance transform, and then match templates at different scales. This matching process generates a focus of attention (candidate people) that are later confirmed using a trained Support Vector Machine (SVM) classifier. Our results show that this method is both fast and precise and directly applicable in robotic architectures.

1 Introduction

Detection and recognition of objects from images disregarding orientation, scale and view is a very important research subject in computer vision. People detection in images and video sequences is a research subject in this area. We are interested in this problem from a robotic application point of view since we are currently in early development stages of a robotic application for search and rescue operations [2].

The problem of people detection is very complex and has not been solved in its generality, but there have been advances where the pose is fixed, such as in the case of pedestrians [1, 9, 14]. However not much attention has been given to the problem when the camera cannot be assumed stationary (therefore not having an explicit scene model).

Our approach uses fast template matching as a focus of attention. Basically it discards locations where there is no silhouette matching the human body. And from those candidate locations (ideally, a very reduced set), we query a full scale SVM.

The contributions pretended are two-fold: first the design and implementation of a vision system that integrates top-down template matching with bottom-up classifiers; and second a concrete implementation on board a robot in an embedded application.

The rest of the paper is organized as follows, first we describe distance transforms for template matching and then support vector machines for pattern recognition, after that we describe the system details, then the results are presented. Finally, the discussion and conclusions are presented and then ideas for future work are given.

2 Distance Transform for Template Matching

A distance transform (DT) converts a binary image (containing values 0 and ∞) to an image where each pixel value denotes the distance to the nearest feature pixel. From this definition of the distance transform problem, a $O(n^4)$ algorithm can be readily constructed (for an $n \times n$ image). However, over the last 20 years the state of the art has advanced either approximating the EDT in a $O(n^2)$ time or providing an exact solution in a $O(n^3)$ time.

Many DT algorithms exist, the differing characteristic is the distance metric and the propagation of local distances. In particular we use Euclidean distance and Maurer's line-column scanning method [10].

After the image has been adequately preprocessed the template matching step begins. As described in by Gavrilu [7], a given image I is said to be matching a template T when:

$$D(T, I) \leq \theta \quad (1)$$

where θ is a user defined threshold on the maximum acceptable dissimilarity between the DT image and the template, and $D(T, I)$ is given by:

$$D(T, I) = \frac{1}{|T|} \sum_{t \in T} d_I(t) \quad (2)$$

where $|T|$ is the number of features in T and $d_I(t)$ is the distance between feature $t \in T$ and the closest feature in I .

3 Support Vector Machines for Pattern Classification

Support vector machines (SVMs) is a principled machine learning technique that is well founded in statistical learning theory.

SVMs have two outstanding characteristics: (1) they have a solid mathematical foundation and (2) strong practical results in large-scale, real-world problems.

Traditional machine learning methods such as backpropagation, minimize the training error, while SVMs minimize a bound on the empirical error and the complexity of the classifier, simultaneously. Therefore, SVMs are likely to perform better than conventional techniques, such as backpropagation trained neural networks. The decision surface of an SVM is given by:

$$f(x) = \text{sgn} \left(\sum_{i=0}^{N_s} \alpha_i y_i K(x, x_i) + b \right) \quad (3)$$

where N_s is the number of support vectors (points closest to the separating hyperplane, in terms of which the decision boundary is defined); x is the point to be classified, x_i is a support vector, and α_i is the corresponding Lagrangian multiplier. K is a kernel satisfying Mercer's conditions. For a complete review of SVMs for pattern recognition (see [4]).

4 System Details

At its core, our system for person detection uses template matching employing Euclidean distance transform (EDT) to evaluate candidate people by independent components (such as torso, leg, arm, head). These matched components are immediately verified using a SVM specialized for that component. If valid, the component is adequately marked on the image. The very first step is preprocessing. Each input image is grayscale and contour-filtered using the Marr-Hildreth method[11]. After that, the contoured and grayscale (CG) image is transformed using an EDT. Figure 1 shows the result of running the preprocessing step on three example images.

We have devised two simple methods for image scanning:

- Using exhaustive scanning. In an $X \times Y$ image with an $N \times M$ template, we first try to match the window defined by the rectangle $(0, 0, N, M)$; after that the one defined by $(1, 0, N + 1, M)$, and so on until reaching the end of the image at that scale.
- Using random sampling. In an $X \times Y$ image with an $N \times M$ template, we select a fixed number of samples proportional to the size of the image. This scanning method accelerates the process with a sacrifice in precision.

In the offline experiments we use exhaustive scanning because runtime performance is not an issue. However, the online version uses the randomized method.

After experimentation we settled with 12 templates. More templates means a better definition of the class of interest but also translates into a slower matching process. The templates are taken from photographs of the object of interest after contour filtering it and obtaining the relevant connected components.

When an image window matches a template, a previously trained and bootstrapped SVM is queried. If the SVM classifies the window as a valid component, the component is then marked in the original image taking into account the scale. Compared to template matching, SVM query phase is very slow. We have looked into simplifying the verification and use Burges's method [3] but later noticed that a homogenous quadratic kernel does not perform well on some of these component datasets.

This approach is not new. Heisele et al. [9] and Gavrila [7], both use some type of hierarchical quick discard method. However, our method is very simple and uses a small amount of templates compared to the results reported by Gavrila [7].

The initial prototype of the current system was written in Python. It uses the LIBSVM support vector machine library [5]. For image processing, we used

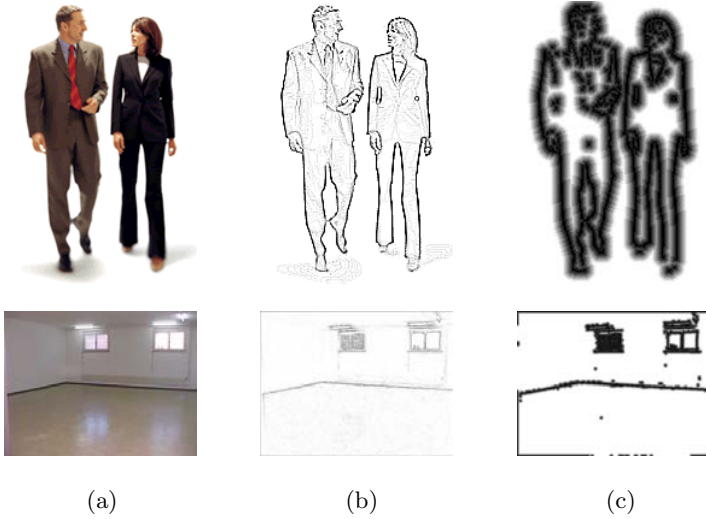


Fig. 1. (a) is the original image, (b) is the contoured and grayscale image and (c) is the distance transformed image ready for template matching.

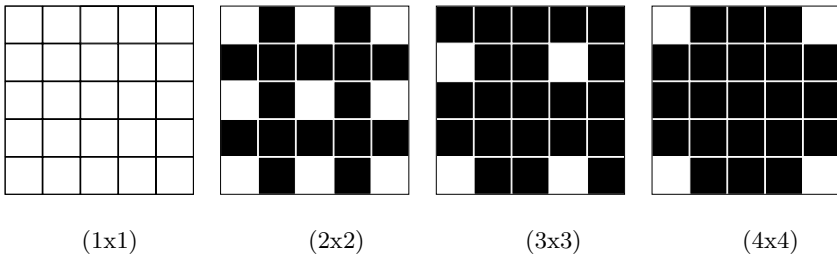


Fig. 2. Chessboard feature selection for various sizes. White squares represent selected pixels, black squares represent non-selected pixels.

the Python Imaging Library (PIL). The production version of the system is written in C++ and uses LIBSVM and ImageMagick. The main difference in the two implementations is mainly performance. On our 1.6 GHz Pentium IV machine, the C++ version runs at 3 frames per second. The system does not use movement as a focus of attention; using movement our system should be considerably faster.

5 Results

We use a chessboard sampling of the pixels in the input image, as presented in Fig. 2. The ROC (Receiver Operating Characteristic) curves in Fig. 3 (right) show that the loss in accuracy is not significant while this feature selection method makes real time performance feasible for our approach. The fact that

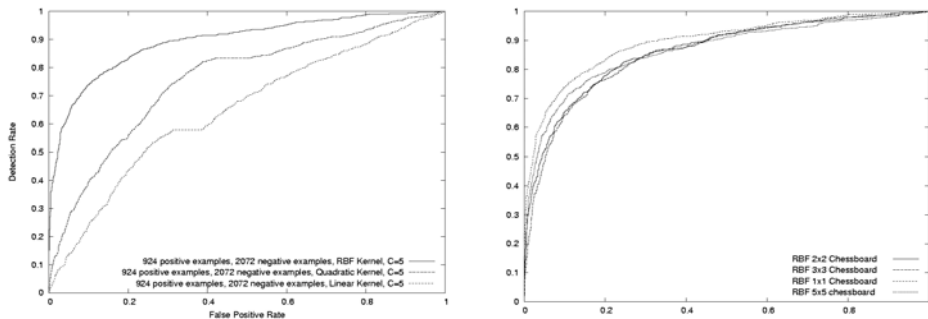


Fig. 3. Left: ROC Graphic of the SVM classifier with an RBF (Radial Basis Function), quadratic and linear kernel and C=5. Both classifiers are of similar complexity. Notice the poor performance of the linear and quadratic kernels. Right: ROC Graphic of the SVM classifier with an RBF (Radial Basis Function) and different chessboard intervals. The loss in accuracy can also be observed in Table 1.

Table 1. Chessboard feature: selection and mean and standard deviation of the classification rate doing a 5-piece cross-validation of the torso classifier.

Features	Mean \pm Std. Dev.
1x1	89% \pm 1%
2x2	86% \pm 1%
3x3	86% \pm 1%
5x5	87% \pm 1%
7x7	83.5% \pm 1%

this type of very simple feature selection approach works shows that the training data are highly redundant.

We applied a 5-piece cross-validation of the training set and report the mean and standard deviation of the classification accuracy rate of the torso classifier in Table 1. Results show that we obtained high accuracy rates on a very large complex dataset.

In Fig. 3 (left) it can be clearly observed that the linear and quadratic kernel perform very poorly in this domain. While using a quadratic kernel, Burges’s method [3] can be readily applied, as reported by Papageorgiou and Poggio [13] after results reported by Osuna et al. [12] in another domain. We consider the precision loss to make this approach prohibitive.

We present several examples of the output of the offline version of the system in Fig. 4. Notice that kids are detected by the system. We consider this to be encouraging since their characteristic proportions are different to those of an adult. The system is also able to correctly classify a naked torso. This is remarkable since the torso of a naked person is considerably different to the torso of a dressed person.

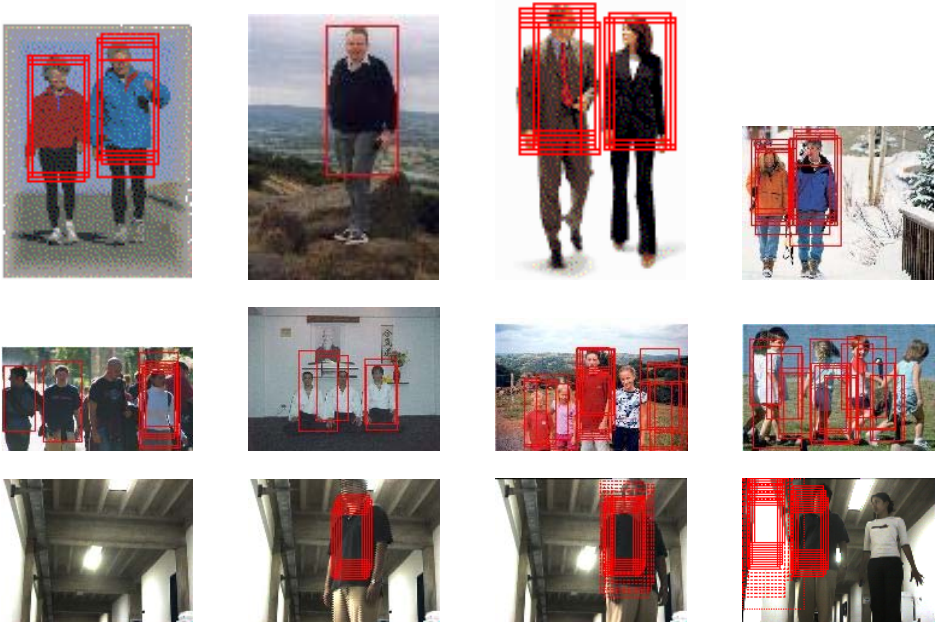


Fig. 4. The first two rows contains examples of the system running on several images offline. The last row shows results obtained by the online version of the system in our office environment.

6 Robotic Application

We tested the system onboard an ActivMedia Robotics Pioneer 2 mobile robot. The online version (onboard the robot) uses the randomized scanning method previously described.

It is important to note that because the camera is not stationary and the background is constantly varying, simple techniques of background subtraction cannot be used for getting the foreground objects. We execute multi-scale exhaustive scanning at each frame.

Because a robotic application usually needs to be run on hardware that is not last generation, we found the querying an SVM on every candidate quickly becomes a crippling bottleneck. We eliminated the SVM querying step from the online version.

The performance (as measured by false positives and false negatives) degenerated significantly. To handle this we adjusted (downwards) the value of Θ in the template matching step. Further, to enhance the precision of the system in our office environment, we measured the correlation of the value pixels on the DT image over the template as described in equation 2 and called this value β and measured the percentage of matching non-data points in the template compared to the contoured image and called this value α . So the matching criteria is:

$$\frac{\alpha}{\beta} > \gamma \quad (4)$$

where γ is an experimentally set threshold value. The matching criteria seeks a balance of many matched points with low matching error (derived from the distance measure of the EDT image). This refinement of the matching criteria significantly decreases the false-positive rate and eliminates the need of querying an SVM to have acceptable results.

The online version of the system works at 3 Hz.

7 Conclusions

We have presented an approach to vision-based person detection in robotic applications that integrates top down (high speed) template matching with bottom up classifiers. We detect components of the human silhouette such as torso and legs; this approach provides greater invariance than monolithic methods to the wide variety of poses a person can be in.

The torso detection methodology presented currently works very well even though each pattern contains more than 1400 features. We have found that the torso can be characterized as very noisy data due to the presence of clothes. The trained SVM classifier correctly captures the relevant information to classify a torso from CG image data, yet querying it is a bottleneck that makes unfeasible to run the system in real time. We presented an alternative using only template matching.

We believe this shows the wide range of applicability of our approach. Our torso dataset contains 924 torsos (from the MIT Pedestrian dataset) and 2072 non-torsos (the non-torsos were generated after a bootstrapping process).

Developing classifiers and templates for other components of the human body (more important in other poses) for use by this method constitutes promising future work. By detecting components of the human body our method is more resilient to occlusion than monolithic approaches.

Our system is not ready for mission critical applications. Performing a principal component analysis (instead of the described chessboard) for feature selection would be a challenging future direction with this large-scale dataset. In the future, we intend to automatically construct shape models using techniques such as described by Duta et al. [6] and Gavrilu et al. [8] to generate a larger template set before continuing on to the development of the classifiers for search and rescue poses.

References

1. M. Bertozzi, A. Broggi, R. Chapuis, F. Chausse, A. Fascioli, and A. Tibaldi. Shape-based pedestrian detection and localization. *Procs. IEEE Intl. Conf. on Intelligent Transportation Systems*, pages 328–333, 2003.
2. A. Brando and C. Chang. Firefighter-robot interaction during a hazardous materials incident exercise. In *11th International Conference on Advanced Robotics*, volume 2, pages 658–663, 2003.

3. C. J. C. Burges. Simplified support vector decision rules. In *International Conference on Machine Learning*, pages 71–77, 1996.
4. C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.
5. C. C. Chang and C. J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
6. N. Duta, A. K. Jain, and M. P. Dubuisson-Jolly. Automatic construction of 2d shape models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(5):433–446, 2001.
7. D. Gavrilu. Pedestrian detection from a moving vehicle. *Proc. of the European Conference on Computer Vision*, 2(8), 2000.
8. D. Gavrilu, J. Giebel, and H. Neumann. Learning shape models from examples. In *Pattern Recognition, 23rd DAGM-Symposium, Munich, Germany, September 12-14, 2001, Proceedings*, volume 2191 of *Lecture Notes in Computer Science*. Springer, 2001.
9. B. Heisele, C. Nakajima, M. Pontil, and T. Poggio. People recognition in image sequences by supervised learning. Technical Report CBCL-188, MIT Artificial Intelligence Laboratory, June 7 2000.
10. C.R. Maurer Jr. and V. Raghavan. A linear time algorithm for computing the euclidean distance transform in arbitrary dimensions. In *IPMI*, 2001.
11. D. Marr and E. Hildreth. Theory of edge detection. *Proc Roy. Soc. London*, page B207:187, 1980.
12. E. Osuna, R. Freund, and F. Girosi. Training support vector machines: an application to face detection. In *1997 Conference on Computer Vision and Pattern Recognition (CVPR '97), June 17-19, 1997, San Juan, Puerto Rico*. IEEE Computer Society, 1997.
13. C. Papageorgiou and T. Poggio. Trainable Pedestrian Detection. In *Proceedings of the 1999 International Conference on Image Processing (ICIP-99)*, pages 35–39, Los Alamitos, CA, October 24–28 1999. IEEE.
14. C. R. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfunder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785, 1997.