Pedestrian detection using a feature space based on colored level lines

Pablo Negri[1,2], Pablo Lotito[1,3]

[1] CONICET, Av. Rivadavia 1917, Capital Federal, Argentina
[2] Instituto de Tecnologia, UADE, Lima 717, Capital Federal, Argentina
[3] PLADEMA-UNCPBA, Campus Universitario, Tandil, Argentina

Abstract. This work gives the guidelines to develop a pedestrian detection system using a feature space based on colored level lines, called Movement Feature Space (MFS). Besides detecting the movement in the scene, this feature space defines the descriptors used by the classifiers to identify pedestrians. The multi-channel level lines approach has been tested on the HSV color space, which improves the one-channel (gray scale) level lines calculation. Locations hypotheses of pedestrian are performed by a cascade of boosted classifiers. The validation of these regions of interest is carry out by a Support Vector Machine classifier. Results give more than 78.5 % of good detections on urban video sequences.

1 Introduction

This work aims to detect pedestrians in street video sequences using a pattern recognition system. Our main contribution is the development of the Movement Feature Space (MFS) based on level lines [6,2]. Using an adaptive background model, the MFS identify moving level lines of objects, preserving their gradient orientation and a factor similar to the gradient modulus. This MFS has two objectives in this system. First, it generates a descriptor of the moving objects in the scene and then, it becomes the input of the detector to classify between pedestrian and non-pedestrian classes.

Working with the MFS has interesting advantages. For instance, it adapts well to slow changes in the scene while it is robust to rapid variations (i.e. illumination changes or weather conditions). In these situations, the people appearance on the MFS does not change significantly compared to normal conditions and thus, they are easily detected by the classifiers. On a transformed HSV color space, called Texton Color Space (TCS) [1], we compute the level lines of each channel. The performance of this color approach has been contrasted with that from the B&W monochromatic MFS (hereafter called MFS-TCS for the former and MFS-gray for the latter).

The pedestrian detection algorithm has three steps. The first one consists in identify the movement on the scene with our MFS. Then, a cascade of boosted classifiers generates several hypotheses. Finally and using a Support Vector Machine (SVM) classifier, these hypotheses are thus confirmed.

The paper is structured as follows. Section 2 gives the guidelines to obtain the MFS in the video sequences. In section 3, we introduce changes to extend the MFS to color images. Section 4.1 describe the input descriptors and the classifiers used in the pedestrian detection. Experimental results over a dataset built for this particular purpose are given in section 5. Finally, we give our final remarks and future extensions in section 6.

2 Movement feature space based on level lines

2.1 Definition of level lines

Let I be a monochromatic image with $h \times w$ pixels, where $I(p)$ is the intensity value at pixel p whose coordinates are (x, y). The (upper) level set X_λ of I for the level λ is the set of pixels $\mathbf{p} \in I$, so that their intensity is greater than or equal to λ, $X_\lambda = \{\mathbf{p}/I(\mathbf{p}) \geq \lambda\}$.

For each λ, the associated level line is the boundary of the corresponding level set X_λ, see [6]. Finally, we consider a family of N level lines $\mathcal{C}$ of the image I obtained from a given set of N thresholds $\Lambda = \{\lambda_1, ..., \lambda_N\}$. From these level lines we compute two arrays S and O of order $h \times w$ defined as follows:

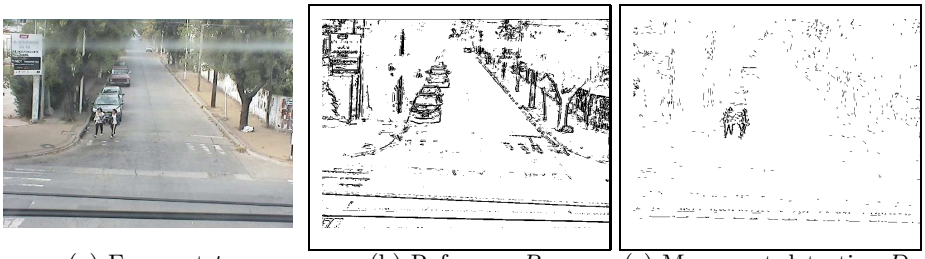
- $S(p)$ is the number of level lines C_λ superimposed at p. When considering all the gray levels, this quantity is highly correlated with the gradient module.
- $O(p)$ is the gradient orientation at p. In this paper, it is computed in the level set X_λ using a Sobel filter of 3×3 pixels, then, orientations are quantized in η values. Here, we do not make difference between a dark-bright transition and a bright-dark one in order to be robust to the high variability in human appearance. For each pixel p, we have a set of $S(p)$ orientation values, one for each level line passing over p. The value assigned to $O(p)$ is the most repeated orientation in the set.

Generally, in the practical implementation, only those pixels for which $S(\mathbf{p})$ is greater than a fixed threshold δ are considered, simplifying the analysis and preserving meaningful contours. In our system, best results where obtained with $N = 48$ and $\delta = 2$.

2.2 Movement detection

As described in [4], level lines have many properties, being invariant to contrast changes. It means that a regular contrast change (monotonic and upper semi-continuous) can either create or remove level lines from a pixel, changing the $S(p)$ quantity, but it could never create a new level line intersecting the original ones [2]. This is crucial because we will use level line intersections to detect movements. The last assertion means that our method will be robust to regular contrast variations.

Now, let two consecutive images I_{t-1} and I_t obtained at times $t-1$ and t. When looking for scene changes at pixel p, the variation of $S(p)$ could correspond



(a) Frame at t (b) Reference R_t (c) Movement detection D_t

Fig. 1. Movement detection of the same sequence, showing the background model reference and the movement detection.

to a movement but also to a change in contrast. A more reliable indicator is a variation on $O(p)$, i.e., $O_{t-1}(p) \neq O_t(p)$. However, the number of points verifying that condition between two consecutive images could be very few. Bouchafa and Aubert [3,2] showed that it is better to work with background reference. They defined an adaptive background reference model, composed of the set of pixel p which are stable over an horizon of time, together with the corresponding values S^R and O^R. More precisely, given a horizon of time T we define R_t as the set

$$R_t = \{p \in I_t : O_{t-1}(p) = O_{t-2}(p) = \cdots = O_{t-T}(p)\},$$

together with O_t^R whose value at p is the preserved orientation, i.e., $O_t^R(p) = O_{t-1}(p)$ for any $p \in R_t$. In practice, the equality constraints in the definition of the reference space R_t can be mollified to allow for small variations of orientation due to noise or other perturbations.

Thus, at time t, the set of pixels p that are not in the reference or have an orientation other than the reference: $O_t(p) \neq O_t^R(p)$, are more likely to correspond to moving objects. These pixels will make up the detected set D_t. Figure 1 shows an example of the reference model of the video sequence at time t. Detected set D_t is presented in fig. 1(c). Note that for this frame, parked cars belong to the reference model and do not appear in D_t.

Below, we will focus the analysis only on pixels in the detected set D_t, and their values of S_t and O_t. This set can be considered as a virtual image with two associated scalar fields, or a kind of feature space referred to as *Movement Feature Space,* or MFS.

3 Colored level lines

Gray scale limitation

The color of an object in the scene is the result of the body reflection. It depends on two characteristics related to the physical properties of the material: the penetration of the light and the scattering of the body's pigments [11]. Clothes are opaque bodies and reflect light in a way that difficult the detection of color transitions between the person and the background. This difficulty becomes harder

for detecting small-sized objects and when the capture is converted into a gray scale.

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

Fig. 2. RGB and Graylevel color spaces. Level line comparison.

Fig. 2(a) shows a person wearing a red t-shirt walking in front of a green hedge. If we transform the color image in its gray-scale representation (see fig. 2(b)), the RGB average approaches both color and it is impossible to find a transition between them without generating any level line (see fig. 2(c)).

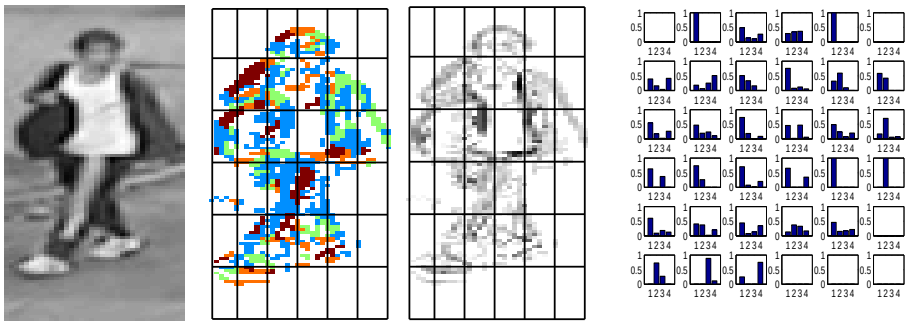
HSV Transformed Space

We thus propose to work with a transformed HSV color space, where it is possible to recover the transition between the body (clothes) and the background finding the level lines showed in fig. 2(d). In this space, Hue (H) is the color feature, Saturation (S) measures the degree of purity of the Hue, and Intensity (V) is the average gray level. Carron [5] proposes this transformation scheme because color features are less sensitive to non-linear effects, being less correlated than the RGB color space.

If Saturation has high values, the Hue is very pertinent. In contrast, when Saturation has small values, Hue is noisy or unstable, and thus it may be irrelevant [5]. The last means that Hue is ill-defined in the unsaturated cases, and this channel can generate irrelevant level lines [11].

To overcome this, Alvarez et al. [1] introduce a simplification of the Carron's method called Color Texton Space (CTS). In this space, two new channels are generated: $S \cdot cos(H)$, and $S \cdot sin(H)$. The intensity V remains unchanged. In this way, in a pixel where H is not relevant because of the low value of S, those channels have not an important value.

First, we calculate for each channel of the Texton Space, $S_t^x(p)$ and $O_t^x(p)$, where $x = \{SsinH, ScosH, V\}$. Then, in order to obtain $S_t^{CTS}(p)$ and $O_t^{CTS}(p)$, it is choose, for each pixel p, the orientation and the modulus of the greatest $S_t^x(p)$. Finally, we obtain D_t, as was explained early, from $S_t^{CTS}(p)$ and $O_t^{CTS}(p)$.



(a) Person (b) O_t grid (c) S_t grid (d) HOG-MFS features

Fig. 3. HOG-MFS calculation on defined grids. In (b), each color correspond to a one of the four directions of the gradient. In (c), darker pixels have highest S_t values.

4 Pedestrian detector system

Features and descriptors extract information from the image. The choice of a good representation space (feature space), helps the classifier to discriminate whether a person is present inside a region of interest (RoI) or a pattern. Using a learning dataset, positive samples (patterns with a person inside) and negative samples (patterns without persons) are grouped into two classes by the classifier by a decision boundary.

4.1 Feature space

The chosen pattern is a rectangle of 12x36 pixels size. The information inside the pattern generates a set of descriptors called feature space.

Two types of feature spaces are used by the classifiers. First ones are calculated from the MFS and the second ones are the Haar-like features [13], which are computed from the gray scale representation of the image.

Feature space from the MFS First set is a HOG-like features, called HOG-MFS, calculated on rectangular patches inside the RoI using S_t and O_t information of the D_t pixels (the MFS space).

HOG-MFS results in a concatenated set of Histograms of Oriented Gradients (HOG) [9]. A RoI is subdivided into a grid of rectangular patches. Within each patch, we compute HOG descriptors as follow:

- a histogram has $\eta = 8$ bins (one for each direction),
- the value for a bin is the sum of the $S(p)$ for the p with this orientation,
- the histogram values are normalized.

Figure 3 shows the 6x6 grid and the HOG-MFS calculated in each grid on figure 3(d). For the sake of simplicity, we consider four gradient directions, where the histogram bin 1 corresponds to the vertical direction, bin 3 corresponds to horizontal direction, and the other two are the diagonals directions.

The second set of features, called MAG-MFS computes the sum of the S_t values inside each patch. This feature helps with the fact that HOG descriptors can make difference between a strong edge in the patch, generating a one in the corresponding bin, and only one pixel of noise, which generates the same histogram after the normalization.

Haar-like filters Rectangular filters or Haar-like features provide information about the gray-level distribution of two adjacent regions in an image. These filters consist of two, three or four rectangles, as proposed by Viola [13]. To compute the output of a filter on a certain region of image, the sum of all pixels values in the gray region is subtracted from the sum of all pixels values in the white one (and normalized by a coefficient in case of a filter with three rectangles). The total number of Haar-like features calculated on the person pattern is 4893.

4.2 Pedestrian classification

Datasets The dataset is composed of seven video sequences. Five of them are employed to train the classifiers, having 3793 labeled pedestrian in more than 5000 frames. Negative samples are obtained from captures without pedestrians. Test sequence have 1324 labeled pedestrians in 4200 frames.

Hypothesis generation In the hypothesis generation, the whole image is analyzed using a sliding window approach [9] to identify pedestrians in these regions.

The detector is an Adaboost Cascade of 20 boosted classifiers [13] discriminating pedestrian and non-pedestrian hypothesis. The implemented methodology is analogous to the one presented in [12][4]. This classifier has a good performance, it evaluates about 25.000 RoIs in some milliseconds and deliver to the next detection step only most probable hypothesis.

The input descriptors of the cascade will be: MAG-MFS, HOG-MFS as a generative function (calculated using a pedestrian model, see [12]), HOG-MFS as a discriminant function, and Haar-like features. At each iteration of the learning process of a strong classifier, the learner chooses between one of those sets of features and their weak classification function associated. The HOG-MFS and the MAG-MFS descriptors are computed on a dense grid of 3707 overlapped square and rectangular patches with different sizes on the pattern.

It is important to note that in the learning process, initially the boosted classifiers of the cascade are chosen among HOG-MFS and MAG-MFS descriptors, which discriminate the movement in the scene. Then, at later steps, classifiers choose Haar-like features, they are highly discriminant and help to identify pedestrians from others moving objects, as circulating vehicles.

[4] `http://pablonegri.free.fr/Downloads/RealAdaboost_PANKit.htm`

(a) Hypothesis generation (b) Hypothesis validated (c) Final RoIs

Fig. 4. The three steps of the detection algorithm.

Hypothesis validation The hypothesis validation is carried out by a SVM classifier. Once the Adaboost classifiers have finished their work, only remains hypothesis which can be considered as harder samples.

The SVM classifier has better discriminant properties than the Adaboost cascade, but it is very time consuming. To speed it up, we use a limited set of HOG-MFS. The RoI is divided in a set of 2x2, 4x4 and 6x6 non-overlapped patches (as show fig. 3(b)). In addition, we have calculated three more grids overlapping the others of 1 patch size (overlapping the grid of 2x2), 3x3 patches, and 5x5 patches. This set of 91 HOG-MFS features is the input of the SVM classifier. Positive samples are the labeled pedestrians of the training datasets, and negative samples are those RoIs validated by the Adaboost cascade. To train the classifier we use the LIBSVM [7] and their default parameters.

Hypothesis filtering Validated RoIs, as shown in fig. 4(b), are then grouped using a Mean Shift Clustering method [8,10]. This is an iterative algorithm that chooses a mean position from the position of neighbouring RoIs. Returned clusters (RoI positions) are considered the system response (see figure 4(c)). Finally, the system output, i.e., the estimated pedestrian positions, is given by the position of those resulting RoIs.

5 Experimental results

Video sequences were recorded by a Vivotek SD7151 camera, filming a street in the city of Tandil (Argentina). The recording format is MJPEG of 640x480 pixels size. We have chosen the minimum JPEG compression to reduce blocking artifacts in captures, moreover a bilinear interpolation is applied to input frames: on each HSV channel and on the gray scale image. With this JPEG resolution, the limited network bandwidth reduces the recording process to one capture every three seconds on average.

Two approaches are compared: a MFS-gray system calculated on monochromatic images, and a MFS-TCS system using the Texton Color Space. Detection results, in percentage, are 76.9 % for the MFS-Gray system, and **78.6 %** for the

MFS-TCS system. The former made 833 false alarms on the test dataset (4200 frames), and the later 899 false alarms. System performance is given by the mean values of individual systems obtained by a 3-fold training. As we see, MFS-TCS outperforms the system using monochromatic images in detection, having few more false alarms at the same time.

6 Conclusions and future work

This article presented a pedestrian detector system, which obtains promising results on urban video sequences. It proposed a Movement Feature Space that help to detect movements and to generate descriptors to identify pedestrians in the scene. The MFS calculated on color images, using a Texton Color Space, improves a system which employs the MFS calculated on monochromatic images.

These are preliminary results and there is a work in progress to improve the detection. This detection system will be employed later to analyze the behavior of pedestrians crossing the street in an intersection and their interaction with moving vehicles.

Acknowledgments This work was supported by the PICT-2283 of ANPCyT, the ACyT R11020 of UADE and CONICET (Argentina).

References

1. Alvarez, S., Salvatella, A., Vanrell, M., Otazu, X.: 3d texton spaces for color-texture retrieval. In: Image Analysis and Recognition. pp. 354–363 (2010)
2. Aubert, D., Guichard, F., Bouchafa, S.: Time-scale change detection applied to real-time abnormal stationarity monitoring. Real-Time Imaging 10, 9–22 (2004)
3. Bouchafa, S.: Motion detection invariant to contrast changes. Application to detection abnormal motion in subway corridors. Ph.D. thesis, UPMC Paris VI (1998)
4. Cao, F., Musse, P., Sur, F.: Extracting meaningful curves from images. Journal of Mathematical Imaging and Vision 22, 1519–181 (2005)
5. Carron, T., Lambert, P.: Color edge detector using jointly hue, saturation, and intensity. In: ICIP. pp. 977–981 (1994)
6. Caselles, V., Col, l.B., Morel, J.: Topographic maps and local contrast changes in natural images. International Journal on Computer Vision 33, 5–27 (1999)
7. Chang, C.C., Lin, C.J.: Libsvm: a library for support vector machines, `http://www.csie.ntu.edu.tw/~cjlin/libsvm`, accessed november 2011
8. Comaniciu, D.: Mean shift: A robust approach toward feature space analysis. IEEE Transactions on pattern analysis and machine intelligence 24(5), 603–619 (2002)
9. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition. pp. 886–893 (2005)
10. Finkston, B.: `http://www.mathworks.com/matlabcentral/fileexchange/10161-mean-shift-clustering`, accessed on march 2012.
11. Gouiffes, M., Zavidovique, B.: A color topographic map based on the dichromatic reflectance model. EURASIP JIVP 2008, 1–14 (2008)
12. Negri, P., Clady, X., Hanif, S., Prevost, L.: A cascade of boosted generative and discriminative classifiers for vehicle detection. EURASIP JASP 2008, 1–12 (2008)
13. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: CVPR. vol. 1, pp. 511–518 (Decembre 2001)