# Fast Human Detection using Histogram of Templates and Haar-like Features

Chethan k and ShahlaSohail
Dayanandasagar college of engineering, Bangalore,
Karnataka, 560078, India

***Abstract*—Pedestrian detection in images is still a problem with view and posture variation. In this paper, combination of a novel feature named histogram of templates (HOT) and Haar-like features are used as descriptors for feature extraction in an image. A Haar-like feature considers adjacent rectangular regions at a specific location in a detection window, sums up the pixel intensities in each region and calculates the difference between these sums. This difference is then used to categorize subsections of an image. HOT features are extracted from every pixel of an image which are meeting various templates for a pre-defined formula. Extracted features from both the methods are provided to support vector machine (SVM) classifier for training and classification.**

***Index Terms* — Histogram of templates, Haar-like features, SVM.**

## I. Introduction

Human detection in images is very important in the area of image based sensing, applications such as surveillance, pedestrian detection, robotics[1]-[6]. Many methods have been introduced for human detection in common views and simple background, it is still a problem in the situations like complex background, different views and postures.

The two main problems for designing any human detection system is feature representation and classifier design. The Human detection can be divided into techniques which require background subtraction or segmentation and techniques which can detect humans directly from the input without such pre-processing[7]. Background subtraction techniques usually find the foreground object from the image and then classify it into categories like human, animal, vehicle etc., based on shape, color, or motion or other features. Direct techniques operate on (features extracted from) image patches and classify them as human or non-human. We can also classify techniques based on the features which are used to classify a given input as human or not. These features include shape (in the form of contours or other descriptors), color (skin color detection), motion, or combinations of these.Some of the feature descriptors are Haar-like features[6], HOG[8], Local binary patterns(LBF)[9],HOG-LBF[10], Granularity-tunable Gradients partition(GGP) descriptors[11]. According to most recentresearches performance and accuracy in detection can be improved by combining different feature descriptors[12].

## II. Previous Work

Paul Viola, Daniel Snow, Michael J Jones [6] describes pedestrians detection system that integrates image intensity information with motion information. Use detection style algorithm that scans a detector over two consecutive frames of a video sequence. The detector is trained using Adaboost algorithm to take the advantage of both intensity information and motion information. Intensity information is calculated by finding histogram of image and Motion information can be extracted from pairs or sequences of images by measuring the differences between region averages at various scales, orientations, and aspect ratios. Generalization of the Viola Jones features which operate on the differences between pairs of images in time is used. Using optimized image processing routines the time taken to detect human in an image can be greatly reduced.

Christoph H Lampert [15] introduced Efficient Sub-window Cascade (ESC), a divide and conquer for accelerating the evaluation of classifier cascades for object detection in natural images. The ESC algorithm starts with a single window in stage 1 that contains all possible object locations. Depending on the quality bounds, the window is either accepted as a whole, rejected as a whole, or split into disjoint parts that are separately processed further. Accepted windows are advanced to the next classifier stage or returned as detection if they already were in the last stage. By using an internal representation by set of regions instead of individual regions, ESC can discard large fractions of the potential candidate locations with few classifier evaluations. Thereby it reduces the computational effort compared to the standard way of cascade evaluation for object detection, in which one applies the classifier cascade exhaustively to every candidate region in the images.

Y. Mu, S. Yan, Y. Liu, T. Huang, and B. Zhou [9] introduces Local Binary Patterns (LBP) for human detection. Existing LBP descriptors does not suite for human detection, due to its high complexity and lack of semantics consistency. For this, Paper proposed two variants of LBP which are Semantic LBP and Fourier LBP. Among two popular approaches for human detection this paper uses sub-window based method for feature extraction from an image. Here each neighbor pixel is compared with the center pixel, once whose intensities exceed the center pixels are marked as 1 otherwise 0. From this get binary code and convert it to decimal form for further calculations.

Q. Ye, Jiao, and B. Zhang [16] uses HOG with multi-scale windows for feature extraction. Different sizes of square image blocks are used and slides over entire image to get features. The extracted features are fed into a cascade adaboost to train the classifier, here classifier used was two stage classifier. Drawback of this method is cannot detect pedestrian in crowed scenes.

Shaopeng Tang and Satoshi Goto [17] uses concept of every pixel of image various templates are defined, each of which contains the pixel itself and two of its neighboring pixels. If the texture and gradient values of the three pixels satisfy the pre-defined formula, the corresponding template for this formula.This extracted features are passed to the SVM classifier for classification of pedestrians. Results shows that Histogram of Templates(HOT) feature is more discriminative than HOG feature for the same training method.

## III. Proposed Method

This method gives overview of our feature extraction chain, which is summarized in Fig.1. This section is divided into A. Feature extraction B. Classifier

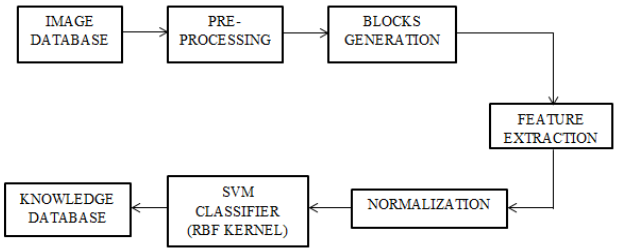


Fig .1. Flow chart for Human detection

### A. *Feature Extraction*

In pattern recognition and in image processing, feature extraction is a special form of dimensionality reduction. When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant then the input data will be transformed into a reduced representation set of features. Transforming the input data into the set of features is called feature extraction. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input.

A Haar-like feature[6] considers adjacent rectangular regions at a specific location in a detection window, sums up the pixel intensities in each region and calculates the difference between these sums. This difference is then used to categorize subsections of an image. For example, let us say we have an image database with human faces. It is a common observation that among all faces the region of the eyes is darker than the region of the cheeks. Therefore a common haar feature for face detection is a set of two adjacent rectangles that lie above the eye and the cheek region. The position of these rectangles is defined relative to a detection window that acts like a bounding box to the target object.

In the detection phase a window of the target size(basic haar set as shown in Fig. 2) is moved over the input image, and for each subsection of the image the Haar-like feature is calculated. This difference is then compared to a learned threshold that separates non-objects from objects. Because such a Haar-like feature is only a weak learner or classifier (its detection quality is slightly better than random guessing) a large number of Haar-like features are necessary to describe an object with sufficient accuracy. In the Viola–Jones object detection framework, the Haar-like features are therefore organized in something called a classifier cascade to form a strong learner or classifier.

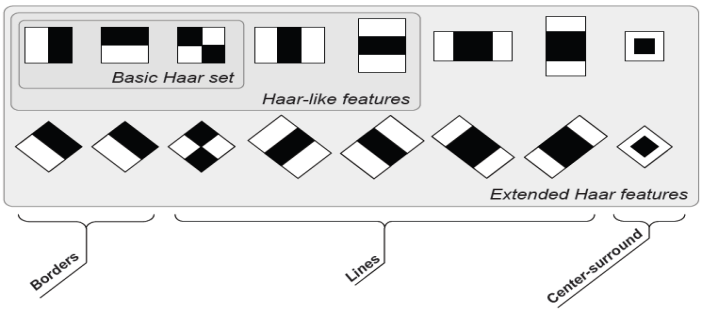


Fig .2. Basic Haar set for feature Extraction

For every pixel of an image, various templates are defined, each of which contains the pixel itself and two of its neighboring pixels [17]. If the texture and gradient values of the three pixels satisfy a predefined formula, the central pixel is regarded to meet the corresponding template for thisformula. Histograms of pixels meeting various templates are calculated for a set of formulas, and combined to be the feature for detection. Some templates are given to define the special relationship of three pixels in Fig. 3.

These templates are used in some formulas. The texture information and the gradient information are also used in these formulas, to give a concrete definition of this feature. The formulas are designed to capture the shape of the human body, and have reasonable computation complexity. For texture information, two formulas are given as following

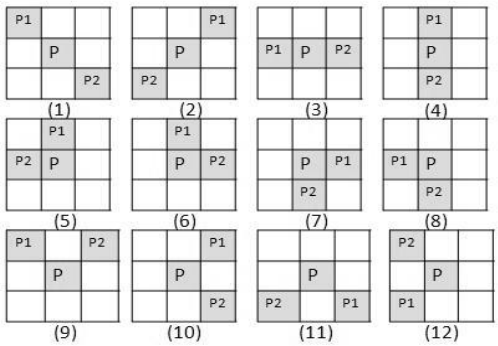


Fig. 3. Templates defining special relationship between three pixels.

$$I(P) > I(P1) \;\&\&\; I(P) > I(P2) \qquad (1)$$

For each template, if the intensity value of *P* is greater than the other two, it is regarded that the pixel *P* meets this template. It can capture the pixels that have the greatest value in one template, and the histogram of pixels that satisfy each template in a sub window can reflect the properties of local part of human body well. For each sub window, the number of pixels meeting each template is calculated to get a histogram as shown in Fig. 4. For example, eight templates are used to extract the feature

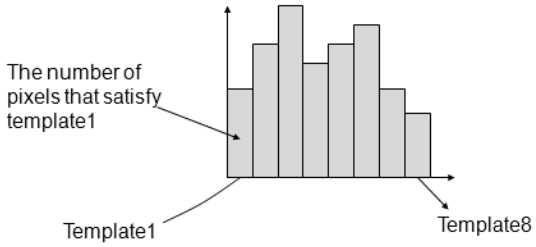


Fig 4. Example of histogram of template for one formula; 8 templates are used, and they correspond to 8 bins. The value of each bin is the number of pixels that meeting corresponding template.

$$k = = \arg_I \max\{ I(P_i) + I(P1_i) + I(P2_i)\} \qquad (2)$$

The sum of intensity values of three pixels in template *k* is greater than the values of other templates; it is can be regarded that *P* meets template *k* . A histogram can be calculated by using formula . By using this formula, we could find the template that has the greatest sum. They can be regarded as the basic unit of human body shape and the shape of human body can be represented well. For the gradient magnitude information, there exist similar formulas.

$$Mag(P) > Mag(P1) \;\&\&\; Mag(P) > Mag(P2) \qquad (3)$$

$$k = = \arg_i \max\{ mag(P_i) + mag(P1_i) + mag(P2_i)\} \qquad (4)$$

Eight templates are usually used to extract the feature, so for each formula, an eight-dimensional vector can be obtained. These vectors are combined together as the final feature as shown in Fig. 5.

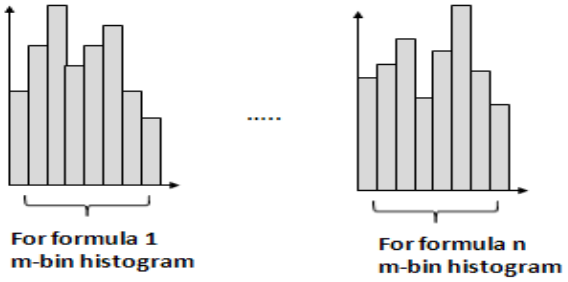


Fig. 5. Final HOT feature for a sub window

B. *Radial Bias Function Kernel SVM*

The inseparable data can be made separable by mapping original input space into a high-dimensional dot product space called the feature space [14], [18]. Mapping is done by a kernel function which is given by

Gaussian RBF kernel : $\Phi(r) = \exp(-r^2 / 2\sigma^2)$ for some $\sigma > 0$

Support vector machines(SVM) are used to find the particular hyper-plane that maximizes the margin of separation and finding such a separating hyper-plane which is optimal. The inner product kernel is a function that is used to find the optimal hyper-plane for SVM network.

Inner product kernel: $K(x, x_i) = \Phi^T(x)\,\Phi(x_i)$

Where $\Phi$ is a set of transformation functions, $x_i$is the i th training sample and x is the input sample. The input sample is mapped onto a feature space and searched for the optimal hyper-plane. The architecture of SVM is as shown in fig 6.

## IV. Results

Database for training SVM is formed from selecting images from INIRIA dataset [1], [8] which consist of

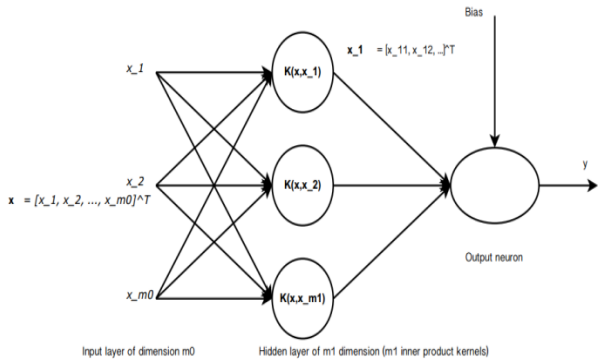


Fig. 6. SVM Architecture

2478 positives and 12180 negatives. For testing purpose images from the same dataset is considered. Figure 7 shows the human examples of two subsets

(a) (b)

Fig. 7. Human samples of two subsets from INRIA Dataset

By training SVM with samples of different subsets and using the knowledge database provided from SVM training for testing results in very good detection of humans in images. Figure 8 shows the detection of humans in images obtained from this experiment.

Fig. 8. Detection examples obtained from this experiment

## V. Conclusion

Human detection in images using SVM as classifier and HOT and Harr-like features as descriptors is implemented using Matlab R2010a software. The proposed method can detect the human in images with various postures, clutter backgrounds, human shadows. By combining two feature descriptors the human detection accuracy in the images is improved.

## References

[1]. Qixiang Ye, Zhenjun Han, Jianbin Jiao and JianZhuang Liu, "Human detection in Images via Piecewise Linear Support Vector Machines", IEEE Transactions on Image Processing, Vol. 22, No. 2, February 2013.

[2]. Y. Xu, D. Xu, S. Lin, T. X. Han, X. Cao, and X. Li, "Detection of sudden pedestrian crossings for driving assistance systems," *IEEE Trans.Circuits Syst. Video Technol.*, vol. 42, no. 3, pp. 729–739, Jun. 2008.

[3]. P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEETrans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.

[4]. M. Enzweiler and D. M. Gavrila, "Monocular pedestrian detection: Survey and experiments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2179–2195, Dec. 2009.

[5]. R. Xu, B. Zhang, Q. Ye, and J. Jiao, "Cascaded L1-norm minimization learning (CLML) classifier for human detection," in *Proc. IEEE Int.Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 89–96.

[6]. P. Viola, M. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," *Int. J. Comput. Vis.*, vol. 63, no. 2, pp. 153–161, 2005.

[7]. Neeti A. ogale, "A Survey of techniques for human detection from video" Department of computer science, university of Maryland, College park, MD 20742.

[8]. N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, pp. 886–893

[9]. Y. Mu, S. Yan, Y. Liu, T. Huang, and B. Zhou, "Discriminative local binary patterns for human detection in personal album," in *Proc. IEEEInt Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8

[10]. Y. Mu, S. Yan, Y. Liu, T. Huang, and B. Zhou, "Discriminative local binary patterns for human detection in personal album," in *Proc. IEEEInt Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.

[11]. Y. Liu, S. Shan, W. Zhang, X. Chen, and W. Gao, "Granularity-tunable gradients partition (GGP) descriptors for human detection," in *Proc.IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1255– 1262.

[12]. M. Enzweiler and D. M. Gavrila, "Multilevel mixture-of-experts framework for pedestrian classification," *IEEE Trans. Image Process.*, vol. 20, no. 10, pp. 2967–2979, Oct. 2011.

[13]. T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, "Object recognition with cortex-like mechanisms," *IEEE Trans. Pattern Anal.Mach. Intell.*, vol. 29, no. 3, pp. 411–426, Mar. 2007

[14]. S. Maji, A. C. Berg, and J. Malik, "Classification using intersection kernel support vector machines is efficient," in *Proc. IEEE Int. Conf.Comput. Vis. Pattern Recognit.*, Jun. 2008, pp.1–8.

[15]. C. H. Lampert, "An efficient divide-and-conquer cascade for nonlinear object detection," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 1022–1029.

[16]. Q. Ye, J. Jiao, and B. Zhang, "Fast pedestrian detection with multi-scale orientation features and two-stage classifiers," in *Proc. IEEE 17th Int.Conf. Image Process.*, Sep. 2010, pp. 881–884.

[17]. Shaopeng Tang, Satoshi Goto, "Histogram of templates for pedestrian detection" in ICICE Trans. Fundamentals /Commun. /Electron. /Inf./&Sysst., Vol. E85 – A//B/C/D, No.xx January 20xx.

[18]. S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, "A fast iterative nearest point algorithm for support vector machine classifier design," *IEEE Trans. Neural Netw.*, vol. 11, no. 1, pp. 124–136, Jan. 2000.