International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Human Detection with Non Linear Classification Using Linear SVM

Vaibhav Janbandhu

Savitribai Phule Pune University, Maharashtra, India

Abstract: ***Human detection is one of the major topics of vision research and has lots of application such as surveillance, robotics, pedestrian detection etc. Detecting human beings expeditiously is the major problem faced in many research works. Support vector machine is one of the techniques used for human detection and among them linear and non linear SVM are most popular. The performance of linear SVM decreases significantly for large set data. Non linear SVM are more promising but they are more computationally expensive. This paper presents the study of classifying the human posture like non linear SVM classification using number of linear SVM. The target space is divided into sub space by each linear SVM which result in non linear classification boundary of human posture. This paper is a review of pricewise linear support vector machine for detecting human being in various environment.***

Keywords: Histogram of orientation (HOG), Human detection, Non linear SVM, Piecewise linear support vector machine (PLSVM,) Support vector machine (SVM).

1. Introduction

Human detection corresponds to the process of detecting human bodies, either as a full posture or a part of human posture. Human detection is the first step for a number of applications such as smart video driving assistance systems, surveillance, and intelligent digital content management. The aim of smart video surveillance is to analyze the video data real-time and alert security officers when predefined events such as burglary happen. Human detection, tracking, and activity recognition are key techniques for the application. The safety of footers in road has become a worldwide problem with the popularity of vehicles. Correctly detecting pedestrians using a camera and warning the drivers before a crash happens will greatly increase the safety of pedestrians. With the high popularity of digital cameras, personal photos have increased exponentially. Searching and locating these images manually are very tiresome. Intellectual digital content management software that automatically adds tags to images to facilitate search is thus an important research area. The number of the images taken is of human, so human detection will form fundamental part of such tools.

Due to this number of applications there are several challenges that should be considered through the detection process. The challenges mostly associated with problem of variance of illumination, color, scale, pose, and so forth. This is also a difficult problem due to the monitoring conditions and the variability of poses and orientations that the human body can adopt. The response of a well-organized person detector provides a bounding polygon or box at the location of human occurrence as shown in figure 1.

Figure 1: showing human detection system output

There are lot of factor due to which human detection becomes a demanding problem. First, the within-class magnitude of changes is very large. A powerful human detector must address the issues such as vary of viewpoint, illumination, pose, clothing, etc. Second, background clutter is common and varies from image to image. The detector must be capable of distinguishing the object from complex background regions. Third, partial occlusions create further difficulties because only part of the object is visible for processing. The first two difficulties present conflicting challenges, which must be tackled simultaneously. A detector that is very specific to one type of human instance will give less false detections on background regions, while an overly general detector can handle large intra-class variations but will generate a lot of false detections on background regions. Given a single image, an ideal human detector should be able to identify and locate all the present humans regardless of their position, scale, or pose. However, because of the articulations of the human body, it will be a very difficult problem to detect humans of all poses and viewpoints; most existing systems only deal with stand-up humans and use learning-based methods. Within learning-based methods, the processing is done as follows: an input image is scanned at all possible locations and scales by a sub-window. Human detection is posed as classifying the pattern in the sub-window as either human or non human.

2. Visual feature for human detection

Local filters are operated on pixel that intensively used in feature set. The most famous method in both dense and spare representation is Histogram of oriented gradient (HOG). To diminish the dimensionality of the feature HOG were represented using local fix block at single fix scale and performance can be increased by this method. Spatiotemporal features for human posture detection can also be used in which information such as motion is present.

HOG calculates the histogram of gradients on orientation and magnitude in an image. First on input image we normalize gamma and color. The input image is tested with RGB (Red, Green and Blue), LAB and gray scale model. The RGB is the basic model which represents the large percentage of visible spectrum. LAB model is represents on the human color. Images produce using black-and-white or grayscale scanners are typically displayed in grayscale. Then gamma normalization and compression is done with square root and log function. The gradients can be computed with filter mask in x-direction as shown in figure 2.

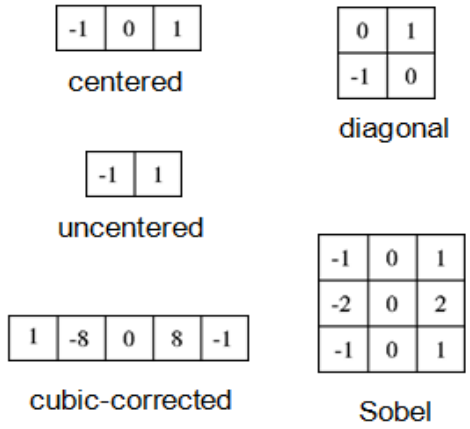

Figure 2: Computes gradients for HOG

The orientation can be calculated with $\theta = arc\tan(\frac{S_y}{S_x})$ and magnitude with $S = \sqrt{S_x^2 + S_y^2}$ as shown in figure 3. The result is weighted for each spatial and orientation cell. The HOG can be calculated either using R-HOG (Rectangular) or C-HOG (circular) as shown in figure 4. Contrast normalization over overlapping spatial block are used for invariance of condition of spiritual awareness and shadowing.

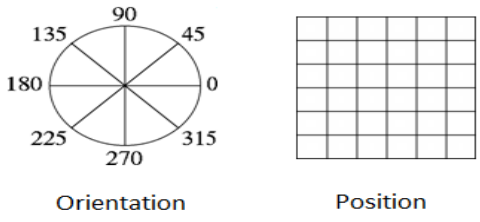

Figure 3: Histogram of gradient orientations

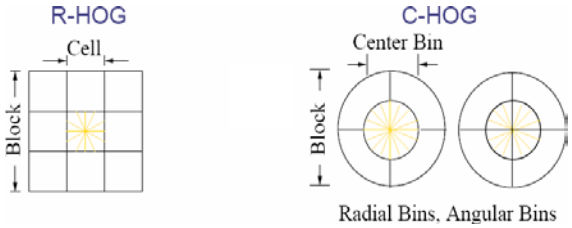

Figure 4: R-HOG and C-HOG

L1-norm: $v \to v/(||v||_1 + \epsilon)$ L1-sqrt: $\sqrt{v/(||v||_1 + \epsilon)}$

L2-norm: $v \to v/\sqrt{v/(||v||_2^2 + \epsilon^2)}$ L2-hys: L2-norm, plus clipping at .2 and renormalizing.

This L1 norm and L2 norm are used for contrasting normalization of an image. Then histogram of over detection windows are collected and feed to SVM classifier. Based on this features of histogram the classifier detect the human posture in an image.

Histogram of oriented gradient (HOG) descriptor [2] allows excellent performance relative to other existing feature sets including wavelets. The gradients are computed over the normalized color and gamma of input image. Then HOG features are gathered from imbrications of spatial blocks. The detector window is scanned all over the image and collects all information in a sliding window in a pyramid form. Then the vectors are feed to linear SVM for human detection. A combination of HOG and Haar like feature is used in [1] to obtain low false positive rate. For calculations of Harr like feature it sanctions to compute sum of rectangular areas in the image, at any position or scale, utilizing four or more than four lookups, depending on how it was defined. It is calculated sum = I(C)+I(A)-I(B)-I(D) where A, B, C, D are points belong to the integral image I. The authors find that the system performs better compared to Haar like & Adaboost method, HOG, HOG & NMS and HOG & Haar classifier.

For human representation features of variable size block i.e. v-HOG is used in [3]. The variable block size, are extracted by varying scales and location with a ratio of 1.0, 0.5 or 2.0. The size of the image is also varies from 12x12 to 16x128 in width and height. The cascade-of-rejectors method is integrated with the Histograms of Oriented Gradients (HoG) features in [4] for quick and precise human detection. HoGs of variable-size blocks are used to capture the features of humans. A novel approach is presented in [5] by combining Histograms of Oriented Gradients (HOG) and Local Binary Pattern (LBP) as the feature set. The LBP feature is denoted as $LBP_{n,r}^{u}$, here *n* and *r* denoted the sample points and radius. *u* represents the number of transitions of 0-1 and it must be less than *u*. In LBP the uniform patterns are represented in different bin and all non uniform patterns are represented in on bin. l_∞ represents distance to the central pixel i.e. d_∞ ((x1,y1),(x2,y2)) = max(|x1-x2|,|y1-y2|). In figure 5 the LBP feature extraction is shown with 8 sample points and with 1 radius using l_∞ distance.

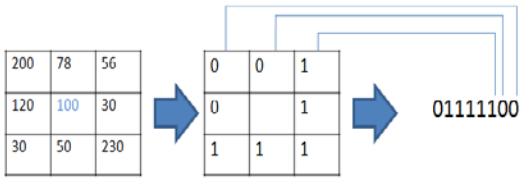

Figure 5: LBP feature extraction

In [6] the MSO (multi-scale orientation) feature is derived from HOG and Harr like feature. MSO is derived through coarse and fine feature in which coarse are the unit orientation and fine is pixel orientation of histograms. For calculating the coarse orientation square blocks are used instead of rectangular box. In coarse gradient the unit is divided into left right sub unit and the vertical gradient is represented as:

$$Dv = |\sum_{x \in left\ subunit} I(X) - \sum_{x \in right\ subunit} I(X)| \quad (1)$$

Here I(X) represent the image color value and horizontal gradient can be calculated using the similar operation. Then the orientation of unit feature can be calculated as $f(n) = Q(aectan(Dv/Dh))$ where Q converts the continuous value into discrete feature bin. Fine feature represents unit orientation histogram. For a pixel the horizontal gradient can be calculated as:

$$dx_\sigma = \max(dx_\sigma^c), c = r, g\ or\ b \quad (2)$$
and $$dx_\sigma^c = (\sum_{i \in \Omega_L} G_\sigma * I_c(i) - \sum_{i \in \Omega_R} G_\sigma * I_c(i)) \quad (3)$$

Here Ω_L and Ω_R represents the left and right neighbor region, G_σ represents the gaussian filter. The vertical gradient dy_σ is calculated in similar way. The pixel orientation in scale σ is obtained by $\theta_\sigma(i) = \arctan(dy_\sigma/dx_\sigma)$. In [7] the limitation of HOG feature is described as it cannot describe the actual object shape as it computes the gradient distribution in a rectangle. Block orientation (BO) feature is presented in [21] that is used to remove stroke and region pattern with noise. BO is derived from Haar like feature and calculated by dividing a cell into left-rignt and up-down sub cells. The horizontal gradient is shown as :

$$Bh = \max_{c \in R,B,G} \{|\sum_{x \in left\ subcell} I(X) - \sum_{x \in right\ subcell} I(X)|\} \quad (4)$$

$$Bv = \max_{c \in R,B,G} \{|\sum_{x \in up\ subcell} I(X) - \sum_{x \in down\ subcell} I(X)|\} \quad (5)$$

Here I(x) represent the R, G, B color value and BO feature is represented in normalization form. Here to reduce noise effect ε is used.

$$BO_h = Bh/\sqrt{Bv^2 + Bh^2 + \varepsilon} \quad (6)$$
$$BO_v = Bv/\sqrt{Bv^2 + Bh^2 + \varepsilon} \quad (7)$$

3. Classifiers for Human Detection

Support vector machines are the potential mechanism for pattern classification problem. SVMs maximize the decision boundary to reach the maximum separation between the object classes. Both linear and non linear SVMs are used for human detection in which non linear SVM requires more computational cost.

3.1 Linear SVM influence

In [8] a new framework for robust object recognition is described. Gabor filter is used for human detection with simple linear SVM classifier. A linear SVM is represented as an optimization problem as:

$$Minimize\ \frac{1}{2} w^T w \quad (8)$$

Subject to: $y_n(w^T x_n + b) \geq 1$ for n=1,..N

Here w represents the unit vector, x_n represents the data point in a plane and $w^T x_n + b$ represent the classification line with y_n output label class. Linear SVM with soft margin for human detection is presented in [2] and better performance is observed by author by using HOG futures. The linear SVM with the soft margin can be represented by an optimization problem as:

$$Minimize\ \frac{1}{2} w^T w + C \sum_{n=1}^{N} \xi_n \quad (9)$$
Subject to: $y_n(w^T x_n + b) \geq 1 - \xi_n$ and $\xi_n \geq 0$ for n=1,..N

Here C is a constant that gives a relative importance to ξ_n and $w^T w$, ξ reprents the slack variable to manipulate the error function and it must be non negative.

A novel approach is presented in [5] by combining Histograms of Oriented Gradients (HOG) and Local Binary Pattern (LBP) as the feature set. Two kind of detector is used i.e. global detector and part detectors. Global detector is used for scanning complete windows and part detector is used for detecting local region. For every ambiguous scanning window global detector is used. If partial occlusion indicates promising detection then part detector is used for final detection process. Linear SVM is used for the classification purpose. The system performs best on INRAI dataset as compared to other methods.

A view-related and sample-dependent combination of multi-cue or multi-feature pedestrian classifiers [9] is involved in multilevel Mixture-of-Experts framework insoles. It separates the complex human classification problem into better sub-problems and does not endure from over fitting effects in high-dimensional spaces. This liner SVM reported landmark works for human detection.

3.2 Non Linear SVM influence

In [10] authors shows that histogram of intersection kernel SVM can be build with the runtime logarithmic complexity in number of support vector. For feature vectors x,z ϵ Rn, the intersection kernel is k(x, z) is represented as:

$$k(x,z) = \sum_{i=1}^{n} \min(x(i), z(i)) \quad (10)$$

and the classification can be computed as :

$$h(x) = \sum_{l=1}^{m} \alpha_l y_l\, k(x, x_l) + b$$
$$= \sum_{l=1}^{m} \alpha_l y_l \sum_{i=1}^{n} \min(x(i), x_l(i)) + b \quad (11)$$

An approximate classifier with constant space and time requirement, independent support vector numbers can be constructed by pre-computing auxiliary table. A kernelized SVM amounts the resemblance in histogram that reporting the feature. The space and time complexity for storing the support vectors and for classification is O(mn). The IKSVM

takes the O(n log m) time complexity and O(mn) space complexity. The main concept is to decompose the classifier as sum of the function for every histogram bin.

3.3 Divide and Conquer influence

Tree structured classifier models are used in [11] for multi-view multi-pose object detection. An iterative boosting based learning method is proposed i.e. Cluster Boosted Tree (CBT) to automatically construct tree structured object detectors. For CBT the input is represented by x ∈ X, where X is the sample space and the tree classifier is represented as: H(x)=[$H_1(x)$,.. $H_C(x)$]. Here c represents the number of sub categories. Each classifier is represented as $H_k(x) = \sum_{t=1}^{T} h_{k,t}(x), k = 1,..,C$. This is the week classifier that is used to represent part of CBT. Based on image features selection the sample space is divide by unsupervised clustering. The refining of classification functions of the ancestors is done to sub-categorization information of the children's in the tree. Edgelet feature is used for this purpose.

In [12] FloatBoost is used for learning a boosted classifier for accomplishing the lowest amount of error rate. A new arithmetical model is provided for stage-wise approximation needed for learning weak classifiers. A classifier which requires fewer weak classifiers than AdaBoost is proposed based on this novel model. The FloatBoost is composed of M number of weak classifier i.e. H_M={h_1,..,h_m} and represented by $H_M(x) = \sum_{m=1}^{M} h_M(x)$. A divide-and conquer procedure in the space of candidate regions is presented in [13] for object detection. It requires fewer evaluations of the classifier functions. ESC (efficient sub-window search) can reject large fractions of the candidate locations with few classifier evaluations. ESC in this way combines the advantages of two current trends for fast object detection i.e. global optimization and cascades. The global optimization exploits spatial correlation of the detection scores and cascades provide that acceleration by approximating the actual detection function with increasing precision. This divides and conquers classifiers reported noteworthy work on human detection.

3.4 Deformable part based model (DPM) influence

Root filter and part filters are used in deformable part based model [14]. Root filter is used to capture the textures that roughly cover full object. It captures the coarse resolution edges for example boundary of an object. Part filter is a high resolution filter that captures the finer resolution features of small parts that root filter fails to capture. The matching of an object is done by calculating the overall score by root location and applying dynamic programming and generalized distance transformation to compute the best part location of the function for the root location. In [14] the approach is based on pictorial structure framework that represent target by collection of arranged parts in deformable configuration. The deformable configuration seizes connections between pair of the parts. Star-structure part-based model with a root filter and with set of part filters and deformation models are defined. Partially labeled data is trained with the help of latent SVM where each example x is scored by a function:

$$f_\beta(x) = \max_{z \epsilon Z(x)} \beta.\phi(x,z) \quad (12)$$

Where β represents vector of model parameters, z represents latent values and $\phi(x,z)$ represents the feature vector. Object model are defined by filters to make sub-windows of feature pyramid and lower dimensional features are discovered which can efficiently computed and easily interpreted.

Extension to the DPM is presented in [15] in which multiple mixture components and object classes are allowed to share object part models. This outcome in added compact models that improves the outcome of incomplete size training set by allowing training examples to be shared by multiple components. The learning is provided by an energy function E as:

$$E(W,V,\beta) = \lambda R(W,\beta) + \sum_{k=1}^{n} L(y^k, h(x^k)) \quad (13)$$

Here to make certain good generalization a regularization term R is used. Loss term is used for detecting how well training data is predicted. Also {W,V} defines appearance and spatial configuration of the parts and how parts are linearly combined is denoted by β. A better detection with a linear classifier can be obtained by using this deformable part based model that consider that each part has smaller deformation, lower dimensionality and non-linearity.

3.5 Piecewise and localized SVMs influence

In [16] author concentrates on multi-category discrimination of sets or objects by applying multi-classification SVM (MSVM) model. This MSVM is represented by an objective function to produce piecewise MSVM as:

$$\min_{w,\gamma} \frac{1}{2}\sum_{i<j}^{k} \|w^{(i)} - w^{(j)}\| + \frac{1}{2}\sum_{i<j}^{k} \|w^{(i)}\|, \quad (14)$$

Subject to $A^i(w^{(i)} - w^j) - (\gamma^{(i)} - \gamma^{(j)})e - e \geq 0$ for i, j=1,..,k and $i \neq j$.

Here the difference between the w(i) and w(j) is that of the normal vector. The location of original and optimal hyperplane is differentiated by $\gamma^{(i)}$ and $\gamma^{(j)}$. Both linear and non linear piecewise classification can be computed using this method. Linear programming is used to represent multi class problem as single optimization problem. Study shows that it performs great with kernel SVM.

In [17] the feature space is divided into a number of sub spaces and then piecewise classification is done on each sub-space by SVM. Localized Support Vector Machine (LSVM) is presented in [18] for classification of non linear decision surface. The localized SVM is build up by the following optimization problem as:

$$\min_{w,b,\xi} \frac{1}{2}\|w\| + \beta \sum_{i=1}^{n} \sigma(\bar{x_s}, x_i)\xi_i \quad (15)$$

Subject to: $y_i(w^T x_i + b) \geq 1 - \xi_i$, for $\xi_i \geq 0$, i=1,2,..,n.

Here $\sigma(\bar{x_s}, x_i)$ represents the similarity between test and training example. The first term in the objective function is used for classification where as second term is used for add a penalty for each misclassification. Multiple liner SVMs are constructed by LSVM to correctly classify the test example. LSVM uses the strength of both SVM and KNN. As LSVM is computationally costly Profile SVM [17] is proposed by

author and represented by supervised clustering algorithm i.e. Magkmeans as:

$$\min_{C,Z} \sum_{j=1}^{k} \sum_{i=1}^{n} Z_{i,j} \|X_i - C_j\| + R \sum_{j=1}^{k} \left|\sum_{i=1}^{n} Z_{i,j} y_i\right| \quad (16)$$

The class labels of training example is denoted by $Y=(y_1,..y_n)^T$, X_i represents the i^{th} row in the matrix and C represents the centroid of the cluster. Non negative scaling parameter is shown by R and Z represents the elements of cluster membership matrix. PSVM prepare the example set into cluster and then apply PSVM on each partition. PSVM performs computationally stronger then LSVM.

Problems in pattern classification are how to design decision functions that can classify a set of observations correctly with the highest possible level of generalization. Cross distance minimization algorithm [19] is an iterative algorithm. It uses the nearest point pair between two polygons by using non-kernel to compute hard margin. From CDMA support conlitron algorithm (SCA) and the support multiconlitron algorithm (SMA) are derived. They are the non-kernel extension of SVM and can maximize the margin by classifying two classes in an SVM. Multiconlitron is a combination of numerous conlitrons that consist of a set of hyperplanes or linear functions surrounding a convex region. A study of non kernel extension of SVM by piecewise linear classifier (PLC) is put forward by author. A framework to construct PLCs from multiconlitron is put forward for assorting two convexly separable data. A convexly separable concept is presented to separate complex nonintersecting classes from a “conlitron” and a “multiconlitron.”

For multi-category case and to extended binary SVM a multi-category support vector machine (MSVM) is proposed in [20]. The standard version of multi-category support vector machine is represented as:

$$\frac{1}{n}\sum_{i=1}^{n} L(y_i).(f(x_i) - y_i)_+ + \frac{1}{2}\lambda \sum_{j=1}^{k} \|h_j\|_{H_K}^2 \quad (17)$$

Here the euclidean ineer product is represebted by “.” Operation. Reproducing kernel Hilbert space (RKHS) is represented by H_K. L(.) maps the class labels to the row of the matrix and f(x) represents the separating function. A loss function is used for multi-category classification problems that target the implicit class with the maximum conditional probability. MSVM shows the output for non representative training set and unequal or equal training set.

3.6 Other classifiers influences

In [1] human pose and activity detection is done on videos. The main aim is to find out the crossing footer as early as possible. Firstly for this purpose spare sliding window with local binary pattern (LBP) is used for apace detection containing footer motion. Secondly further verification at the frame level is done with generic training data of half size of footer sample. Then spatiotemporal refinement is done to speck the location of footer location. This is three-level coarse-to-fine framework proposed by the author. The proposed system is focused on partly visible footer before they enter in the full view of the video frame and quickly process the input to detect the footer posture as soon as possible. A LBP difference-based motion filter is used to chuck out region that lack motion so that detection process can be speed up. A mixture of HOG and Haar like feature is used to obtain low false positive rate.

Classifier construction is used with feature selection to detect human posture in [3]. To train a series of weak classifier LML (L1-norm Minimization Learning) and min-max penalty function models are used. To construct a strong classifier, L1-norm minimization selects the weak classifier and integer optimization models are used to find the minimal VC dimension. To achieve higher detection rate and accuracy cascade of LML is used. For human representation features of variable size block i.e. v-HOG is used. The system performs great in INRIA human test.

4. Learning and Detection Phase

The learning phase consists of training the classifier for proper and correct detection of human posture. Data set is required for training the SVM classifier. The data set must consist of various images of different views and posture of human figure. Normalization can be done on data set so that the proper representation of figure gamma and color can be presented. On this dataset the features are extracted and histogram is calculated on this feature. SVM classifier utilizes this feature to correctly classify the human posture and human is detected and trained with SVM classifier. The flowchart learning phase is shown in figure 6.

The detection phase is consists of testing an image file for correctly detecting a human posture. Firstly the test image is scanned in a sliding window fashion in all scale and in all location in the image. Based on this the required feature is extracted and histogram is calculated on this feature. SVM classifier is run on this test image, based on the training SVM predicts whether human is present or not in the image. If human is present in the image then it is represented by bounding box in the image. The flow chart of detection process is shown in figure 6.

Sliding window is the most popular technique where detector window at several position and scale are changed over the image. Sliding window is computationally expensive for real time processing. It can be speed up cascade classifier or with the prior information about the target object. In real world environment the moving cameras with changing pitch can be handled by relaxing scene constrains or by approximating geometry. Background subtraction can be done on static camera. Moving cameras assume translator motion that calculates the deviation of the observed optical flow from the expected ego-motion flow field. Interest point detectors can be used to identify the region of interest with high detailed data on local discontinuities of the picture brightness function.

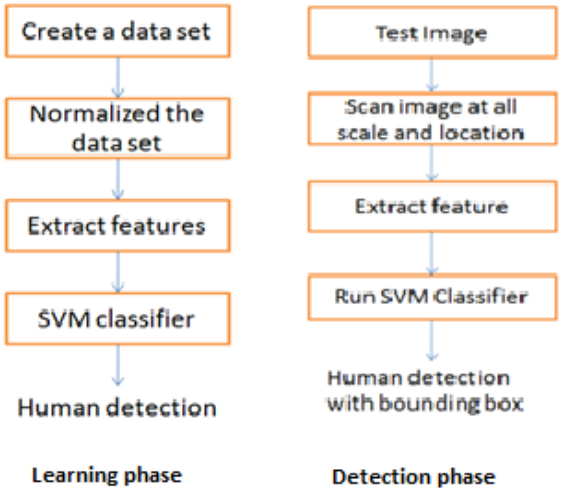

Figure 6: Learning and detection phase

5. Strength of N-linear SVM

With one SVM classifier line two classes can be classified, with two SVM classifier line three classes can be classified and similarly for three SVM classifier line four classes can be classified as shown in figure 7.

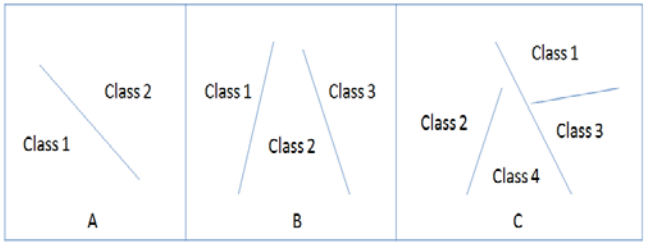

Figure 7: A- shows 1 SVM 2 classes, B- shows 2 SVM 3 classes and C- shows 3 SVM 4 classes

The figure 7 shows the number of SVM used and the number of classes it classify. It shows the representation of different SVM classifier lines for linearly classifiable and linearly separable data. As the number of classifier increased the power of classification also increased with it. This technique is used for classifying the human posture in an image file. With the N-linear SVM, N number of SVM classifier line can be drawn to classify human figure. This N number of classifier line is represented as non linear classification boundary. Thus using a linear SVM a non linear classification boundaries are obtained. Thus the classifier produces a non-linear classifier boundary around human posture utilizing the power of N-linear SVM. The SVM classifier lines are produce around the pose of person, which can be represented as drawing the SVM classifier line around the polygon, as shown in figure 8.

The points (1,2,3,a) and (4,5,6,7,b) forms a positive convex polygon which shows the promising result for human posture parts and rest of the part is represented with negative polygon. The nearest point (a,a') and (b.b') are used to classify the SVM line with a large classification margin. In [A32] it is shown that the problem of finding the maximum margin in SVM is same as the problem of finding the nearest point problem (NPP). It shows that SVM problem can be easily transferred to the problem of finding nearest point between two convex polytopes. By applying this idea N-linear classifier are utilized to construct the non linear classification margin around polytopes, in this case a human posture.

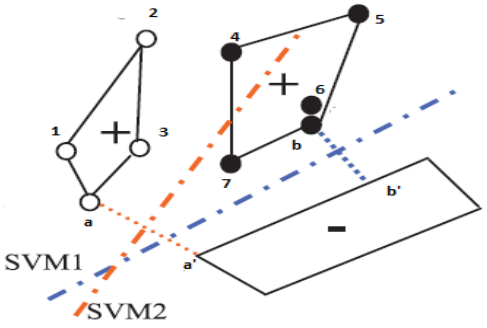

Figure 8: Convex polygon and their SVM

6. Existing System

6.1 Human Detection

The proposed PL-SVM in [21] is contained with two kinds of features for human detection. For this a cascade detector is designed to improve detection performance. A sample of 64 × 128 pixels is divided into cells of size 8 × 8 pixels. Then each group of 2 × 2 cells is integrated into a block in a sliding fashion. The blocks overlap with each other. The gradient orientations of the pixels in the cells are calculated to extract HOG features. Nine dimensional HOG orientations as the features are calculated for each cell. Each block is corresponded by a 36-dimensional feature vector. By dividing each feature bin with the vector module each block is normalized [B6]. Each sample is represented by105 blocks i.e. 420 cells. It corresponding to 3780-dimensional HOG feature vector. In PL-SVM models training is done with the BO features [21] and the HOG features for training samples. As the preprocessing Histogram equalization and median filtering of radius equal to 3 pixels is applied on testing image as preprocessing. An image pyramid is constructed by repeatedly reduced the test image in size by a factor of 1.1. From every layer of the pyramid sliding windows are extracted. For each window BO features are extracted in the first stage and tested with PL-SVM. In the second phase if the window is classified as human then it is tested with PL-SVM with the HOG features.

6.2 PL-SVM Model

A PL-SVM (Piecewise linear support vector machine) [21] is formed from K linear SVMs. It is represented as a piecewise linear function as follows:

$$f(x) = \underset{f_k(x),\ x\in\Omega_k}{\arg max}\ \{C_k(x)\} \quad (18)$$

Here, $f_k(x) = w_k^T.x + b_k, k = 1, \ldots\ldots K$, represents the k^{th} local linear SVM. w_k^T represents normal vector. The threshold is represented by b_k. In (18) $\Omega_k = \Omega_k^+ \cup \Omega^-$denotes the k^{th} subspace of the training samples and it is shown in Fig. 9 [21].

In (18), $C_k(x)$ represents the membership degree of a sample x to Ω_k . From the vantage point of probability the membership degree is defined as,

$$C_k(x) = P_k(y = 1|x) \quad (19)$$

Here, P_k (y = 1|x) is the outputted probability of a sample x. The function of the support vector machine is represented by probability as follows:

$$P_k(y = 1|x) = \frac{1.0}{1.0 + \exp(-(A_k \cdot f_k(x) + B_k))} \quad (20)$$

Here A_k and B_k represents the two parameters computed with a maximum likelihood approximation on the training subset [22], and the parameterized sample to hyper plane distance is represented by $A_k \cdot f_k(x) + B_k$.

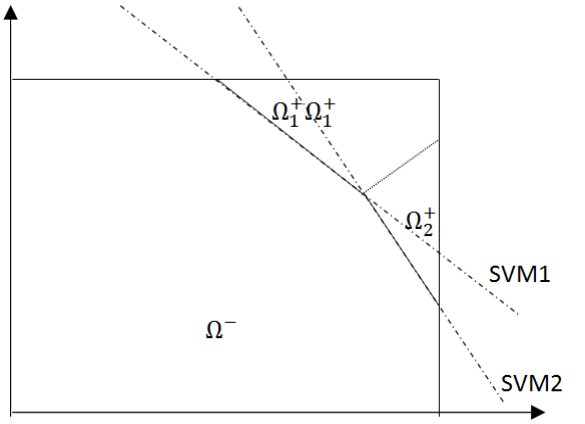

Figure 9: Show the PL-SVM and feature space division. Subspaces are confined with dotted lines and Ω_1^+,Ω_2^+ denote the positive subspaces corresponding to linear SVMs 1, 2, respectively, with Ω^- denoting negatives. Dissimilar positive subspaces are related to samples of different views and postures. Classification boundary of the PL-SVM is noted by bold line segments.

The PL-SVM discriminative function can be converted to a sign function following form for discrimination and detection from (18) while performing classification:

$$F(x) = \text{Sign}\left(f(x)\right) \quad (21)$$

For a given training set $X = \{(x_n, y_n)\}$, $n = 1, \ldots, N$, to train the PL-SVM the multi-objective programming problem is needed to be solved:

$$\min\left(\|\omega\|^2 + \lambda \cdot \sum_{n_1}^{\xi_{n_1}}\right), \ldots, \min\left(\|\omega_k\|^2 + \lambda \cdot \sum_{n_k}^{\xi_{n_k}}\right)$$
$$s.t.\ y_n \cdot F(x_n) - 1.0 + \xi_n \geq 0,\ \xi_n \geq 0, n = 1,2,\ldots,N. \quad (22)$$

The above objective function presumes that all of the local SVMs in a PL-SVM are evenly important. Here n_k denotes the sample index in the k^{th} sample subset. Also λ represents the parameter for training error and the SVM margins. ξ represents the slack factor and $F(x)$ represents the PL-SVM discriminative function defined in (22).

6.3 PL-SVM Training

The human samples are divided into subsets with a K-mean clustering algorithm in a manifold embedded space before training. When clustering is done to these initial subsets then human samples allotted to the same subset have smaller dissimilarities. This heads to an improved sample division than an arbitrary one.

To build up the human manifolds local linear embedding (LLE) algorithm [17] is employed. The feature space is divided into a number of sub spaces and then piecewise classification is done on each sub-space by SVM. By mapping high-dimensional samples to the low-dimensional space LLE computes the low-dimensional and neighborhood-preserving embedding. When a set of human samples in the high-dimensional feature space is given, LLE begins with detecting nearest neighbors based on the Euclidean distance. LLE discovers the optimal local convex combinations of the nearest neighbors to represent each original sample. Then LLE identifies the optimal local convex combinations of the nearest neighbors to represent each original sample. Then it obtains an embedded space by figuring out a sparse eigenvector problem.

6.4 Training Convergence Analysis

Sequential minimization optimization (SMO) is used for every linear SVMs of the PL-SVM for training and can be following objective function:

$$\min_{\omega_k} \frac{1}{2}\|\omega_k\|^2 + \lambda \cdot \sum_{n_k}^{\xi_{n_k}}$$
$$s.t.\ y_{n_k} \cdot Sign\left(f_k(x_{n_k})\right) - 1.0 + \xi_{n_k} \geq 0,\ \xi_{n_k} \geq 0, n_k = 1,2,\ldots,N_k. \quad (23)$$

Here n_k denotes the sample index in the k^{th} subset. Also N_k denotes the number of the samples of the subset. The convergence of the PL-SVM training is analyzed by the nearest point algorithm (NPA) [23]. Let us consider the positive convex hull U_k and the negative convex hull V_k for the k^{th} subset. Also let $\widetilde{u_k} \in U_k$ *and* $\widetilde{v_k} \in V_k$ such that

$$\|\widetilde{u_k} - \widetilde{v_k}\| = \min_{u \in U_k, v \in V_k} \|u - v\| \quad (24)$$

Then the problem of finding $\widetilde{u_k} \in U_k$ *and* $\widetilde{v_k} \in V_k$ is equivalent to finding the solution of k^{th} SVM [23]. According to (19)–(20), the parameterized sample to hyper plane distance of this sample to the hyper plane of this SVM is also the largest. [23] shows that the problem of finding the maximum margin in SVM is same as the problem of finding the nearest point problem (NPP). It shows that SVM problem can be easily transferred to the problem of finding nearest point between two convex polytopes.

7. Conclusion

There are numerous challenges that should be considered through the human detection process. The major problem that faced in human detection is of variation of views and postures. Another difficulty that is faced is to use which feature and classifier for human detection. A study of features and classifier is presented. HOG feature is found out to be most popular features. SVM classifier has also reported as an effective classifier for human detection. PL-SVM shows a promising result in the process of human detection. Detecting human posture with PL-SVM is studied and found out to be promising for detecting human views and posture.

References

[1] 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.
[2] 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.
[3] 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.
[4] Q. Zhu, S. Avidan, M. Yeh, and K. Cheng, "Fast human detection using a cascade of histograms of oriented gradients," in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., Jul. 2006, pp. 1491–1498.
[5] X. Wang, T. X. Han, and S. Yan, "An HOG-LBP human detector with partial occlusion handling," in Proc. IEEE Int. Conf. Comput. Vis., Oct. 2009, pp. 32–39.
[6] 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.
[7] W. Gao, H. Ai, and S. Lao, "Adaptive contour features in oriented granular space for human detection and segmentation," in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., Jun. 2009, pp. 1786–1793.
[8] 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.
[9] 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.
[10] 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.
[11] B. Wu and R. Nevatia, "Cluster boosted tree classifier for multi-view, multi-pose object detection," in Proc. IEEE Int. Conf. Comput. Vis., Oct. 2007, pp. 1–8.
[12] S. Z. Li and Z. Zhang, "Floatboost learning and statistical face detection," IEEE Trans. Pattern Anal. Mach. Intell., vol. 26, no. 9, pp. 1112–1123, Sep. 2004.
[13] 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.
[14] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan,"Object detection with discriminatively trained part based models," IEEE Trans. Pattern Anal. Mach. Intell., vol. 32, no. 9, pp. 1627–1645, Sep. 2010.
[15] P. Ott and M. Everingham, "Shared parts for deformable part-based models," in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., Jun. 2011, pp. 1513–1520.
[16] O. Oladunni and G. Singhal, "Piecewise multi-classification support vector machines," in Proc. Int. Joint Conf. Neural Netw., Jun. 2009, pp. 2323–2330.
[17] S. Q. Ren, D. Yang, X. Li, and Z. W. Zhuang, "Piecewise support vector machines," Chin. J. Comput., vol. 32, no. 1, pp. 77–85, 2009.
[18] H. B. Cheng, P.-N. Tan, and R. Jin, "Efficient algorithm for localized support vector machine," IEEE Trans. Knowl. Data Eng., vol. 22, no. 4, pp. 537–549, Apr. 2010.
[19] Y. Li, B. Liu, X. Yang, Y. Fu, and H. Li, "Multiconlitron: A general piecewise linear classifier," IEEE Trans. Neural Netw., vol. 22, no. 2, pp. 276–289, Feb. 2011.
[20] Y. Lee, Y. Lin, and G. Wahba, "Multicategory support vector machines," Dept. Stat., Univ. Wisconsin-Madison, Madison, Tech. Rep. 1063, 2001.
[21] Qixiang Ye, Zhenjun Han, Jianbin Jiao, Jianzhuang Liu, "Human Detection in Images via Piecewise Linear Support Vector Machines," IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 22, NO. 2, FEBRUARY 2013
[22] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularization likelihood methods," in Proc. Adv. Large Marg. Classifiers, 1999, pp. 61–74.
[23] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, "Afast iterative nearest point algorithm for support vector machine classifierdesign," IEEE Trans. Neural Netw., vol. 11, no. 1, pp. 124–136, Jan.2000.