Pedestrian detection is current research topic in computer vision mainly because of its extensive usage in video surveillance application e.g. elder monitoring, activity, intrusion, event detection and video content analysis. Outdoor low resolution (352x288) pedestrian detection is a challenging task and requires handling many difficulties [8]. The state of art is more focused to handle problems like occlusion, handling challenging cluttered backgrounds[10], the trivial problem of detecting medium to small scale (of size between 30-80 pixels) cloth invariant pedestrian in wide range low resolution visual surveillance camera is yet an open issue[1]. This paper presents a technique to find pedestrian of small to medium scale and invariant clothing(loose cloth) using proposed heuristic scan window formed using motion cues and edgelet based contour detection and SVM classifier trained over a dataset of 5000 negative and 2500 positive 35x20 images using HOG and statistical shape based features. We also provide a dataset of positive and negative examples of real scenarios of surveillance captured from the camera placed on the roof top covering a wide range area. Our experiments indicate that motion cues and edgelet based contour detection (ECD) provide a powerful heuristic scan window for pedestrian detection and HOG along with other statistical shape based features results in more accurate result reducing false positive to 15% than using HOG features alone.

II. RELATED LITERATURE

Wherever Pedestrian detection is a mature area of research in computer vision and a lot of work has been carried out on pedestrian detection [1]-[14]. The current state of art can be divided into three main areas as shown in Fig. 1. First in this category is contour cues, these can be used as a heuristic to obtain initial object location based on some hypotheses [3]. Techniques used for generating this initial hypotheses range from simple sliding window technique[1][2][13] to more complex integration of sliding window and classifier cascade based approaches[2][6]. All approaches [1][2][13] takes an assumption that pedestrian should be 100 pixels and above. Classifier based pedestrian detection techniques use combination of various feature set and a classifier to search for the presence of pedestrian in images or videos. [1], [11][13] introduced a breakthrough in the state of art of pedestrian detection by introducing histogram of oriented gradient (HOG) features to target the contour information of the pedestrian and training the SVM as a classifier. The algorithm proved to be highly accurate for the pedestrians in upright position and of size greater than 100 pixels.

Part based classification [3] gained less success over appearance based overall detection. However, the images are low resolution where pedestrian body parts are not visible so part based detection is not possible.
Temporal cues play an important role especially when the frames under consideration are of low resolution [1]. Many of the current state of art [22] utilize motion information by using either background subtraction [2][6], optical flow[13] and tracking information[10].

III. PROPOSED FRAMEWORK

This paper presents a method to detect pedestrian of small to medium scale (30-80 pixels) from a low resolution and low frame rate static CCTV video camera, placed on the top of the roof 40 feet above the ground with the tilt of 45 degree focusing ground thus capturing a wide range. We search the image segments to find the probability that is a pedestrian given by (probability of being a pedestrian) as shown in eq. (1).

\[
P(ped) = \sum_{i=1}^{k} P(E_i) \times P(ped | E_i) \tag{1}
\]

Where \( E_i \in \{fg, Ap\} \), \( fg \) denotes the foreground mask and \( Ap \) denotes the appearance map generated by BS and ECD module respectively. \( P(E_i) \) is the probability that \( E_i \) is a representation of pedestrian, while \( P(ped | E_i) \) is the conditional probability of pedestrian given \( E_i \). Multiple heuristic windows are obtained as a result of above process where \( i \in \{1, 2, ..., n\} \) where \( n \) is number of heuristic windows. Feature vectors are extracted from each window and send to SVM classifier prediction module that returns a decision of whether \( Hwin \in P \) or \( Hwin \in F \) where \( P \) corresponds to pedestrian. Multiple bounding boxes detecting the presence of pedestrian are the final output of the framework. The block diagram of complete framework is shown in Fig 3.

A. Background Subtraction

In low resolution videos where the appearance features fail to classify the object correctly due to lack of pixel details. In-cooperating motion confidence value will provide dominant information about the location of small to medium sized pedestrian [1]. One of the problem addressed by this work is also low frame capture rate due to network latency. Only two frames are available at a time for processing, considering these limitations we propose a background subtraction technique inspired from[2][6], that take advantage of shifted image to obtain better result than frame by frame differencing. Let \( f(t) \) denotes the current frame and \( f(t-1) \) denotes the previous frame. The foreground mask \( f_{mask} \) is obtained as following

\[
f_{mask} = U + D + L + R \tag{2}
\]
\[
f_{diff} = f(t) - f(t-1) \tag{3}
\]
\[
U = f(t-1) + f_{diff} \tag{4}
\]
\[
D = f(t-1) + f_{diff} \tag{5}
\]
\[
R = f(t-1) + f_{diff} \tag{6}
\]
\[
L = f(t-1) + f_{diff} \tag{7}
\]

\( U, L, R, D \) are the images obtained by shifting the difference image \( f_{diff} \) up, down, left and right by one pixel and adding the result into the previous frame.[2] and [6] uses next frame \( f(t+1) \) to capture motion in all direction to get the motion gradient information however we have used this frame difference and added the shifted version in all direction to get the enhanced version of small scale motion. It can be seen from fig 4 that \( f_{mask} \) provides more information than \( f_{diff} \) along with some noise. This noise is removed using dilation and thresholding. After the \( f_{mask} \) is produced shape based features that include orientation eccentricity [20] and aspect ratio \( A_i \) of the blob \( B_i \) are obtained from the blobs extracted from the \( f_{mask} \) where \( i \) denotes number of blobs extracted. The votes determine the conditional probability \( P(ped | E_i) \) of pedestrian given as shown equation (8).

\[
P_{ped} = \sum_{i=1}^{k} P(E_i) \times P(ped | E_i) \tag{8}
\]
Where $P(fg)$ is the probability the $f_{mask}$ is a significant change or not. $P(fg)$ and $P(ped|f_g)$ are later used to calculate the $P(ped)$ eq. (1).

Fig 4. (a) Reference frame  (b) current frame  , (c)frame difference and (d)foreground mask

![Reference frame](image1)

![Current frame](image2)

![Frame difference](image3)

![Foreground mask](image4)

\[
P(ped|fg) = P(ped|O) + P(ped|E) + P(ped|A)
\]

\[
P(ped|O) = P(O|ped) \times P(O|ped)
\]

\[
P(ped|E) = P(E|ped) \times P(E|ped)
\]

\[
P(ped|A) = P(A|ped) \times P(A|ped)
\]

\[
P(O|ped) = \begin{cases} 0.3 & \text{abs}(O) < 10 \\ 0 & \text{otherwise} \end{cases}
\]

\[
P(E|ped) = \begin{cases} 0.3 & (E) < 0.5 \\ 0 & \text{otherwise} \end{cases}
\]

\[
P(A|ped) = \begin{cases} 0.4 & A_i > 1.25 \\ 0 & \text{otherwise} \end{cases}
\]

\[
o_i = \frac{1}{2} \arctan\left(\frac{2M_{13}}{M_{22} - M_{13}^2}\right)
\]

\[
E_i = \frac{\left(M_{23} - M_{32}\right)^2 + 4M_{13}^2}{M_{22}^2 - M_{13}^2}
\]

\[
M_{13} = \sum (x-x') (y-y')
\]

\[
M_{22} = \sum (x-x')^2
\]

\[
M_{13}^2 = \sum (y-y')^2
\]

\[
A_i = \text{height/width}
\]

\[
P(fg) = \begin{cases} 0.2 & f_{ad} < 128 \\ 0.5 & f_{ad} \geq 128 \end{cases}
\]

**B. Edgelet base contour detection**

The basic idea lies in modeling pedestrian as geometric combination of small curves called edgelets and targeting the contour information of pedestrian as it is the main information targeted by HOG features [15]. The edge map $E$ is obtained by using motion gradient obtained from [2][6]. The main assumption is when seen from the top view pedestrian appear to be belled shaped where the peak of curve corresponds to head of the pedestrian; the peak descends equally in both directions as shown in Fig 2. The problem of curve detection is considered at low pixel level because the derivative and curvature don’t provide sufficient information [15]. In edge map, edge linking starts from the top left of the image and gap size is considered in a heuristic search window of size $wh \times w$ that finds the next pixel to be linked. We are concerned with the left and right part of the edgelets as the curve is traced from midpoint. To ensure symmetry of the curve on both sides. Let $H(x_n, y_n), n \in \{1, 2, 3, \ldots, n\}$ denotes the head curve pixels and $n$ is the number of head pixels, $A_{H,k}$ denotes the angles that the left and right side of head edgelets makes with the midpoint of head, $H(x_n, y_n)$and $l_{kn}$ denotes the positions of the edgelets where $n$ denotes the subscript of line segments combined to make an edgelet then $A_{H,k}$ obtained in radians is given as

\[
A_{H,k} = \arctan^{-1}\left(\frac{y_n - y_{P(n)}}{x_n - x_{P(n)}}\right)
\]

\[
H(x_n, y_n) = \left(x_n - \frac{\sum y_i \cdot x_i}{w}\right)
\]

Let $l_{kn}$ denotes the line segments and $m(l_{kn})$ denotes the angle of the line segment makes with horizontal axis.

\[
m(l_{kn}) = \arctan\left(\frac{y_{l_{kn}} - y_{l_{kn-1}}}{x_{l_{kn}} - x_{l_{kn-1}}}\right)
\]

\[
l_{kn} = (y_{l_{kn}} - y_{l_{kn-1}}) = m(l_{kn})(x_{l_{kn}} - x_{l_{kn-1}})
\]

Let $M_{L,k}$ denotes the mean of angles of the left side and right side edgelets of the body, then $M_{L,k}$ is given by

\[
M_{L,k} = \frac{1}{n} \sum m(l_{kn})
\]

\[
M_{L,k} = \left[l_{l_{kn}}, l_{r_{kn}}, l_{l_{kn}}, \ldots, l_{l_{kn}}\right]
\]

Then $s(x, y)$ belongs to the pedestrian shape curves $P_s$ is given by

\[
P_s = \left[u_{l_{kn}}, u_{r_{kn}}, u_{l_{kn}}, u_{r_{kn}}\right]
\]

The probability $P_b$ that $P_s$ represents a pedestrian curve is given by

\[
P(ped|A_b) = P(ped|A_b) + P(ped|M_b) + P(ped|A_i) + P(ped|A_i)
\]

\[
P(ped|M_b) = P(M_b|ped) + P(M_b|ped)
\]

\[
P(ped|A_b) = P(A_b|ped) + P(A_b|ped)
\]

\[
P(ped|A_b) = P(A_b|ped) + P(A_b|ped)
\]
Where $f$ denotes $A_{hk}$, $B_{hk}$ and $O_i$ respectively. $P(Ap)$ is 0.5 as the chances of pedestrian appearing in appearance map is 0.5. This module output the labeled image containing shapes similar to pedestrians. Edge map obtained by canny operator is scanned from top left for a pixel which has a value of 1 indicating it belongs to edge pixels is selected as a seed pixel, let’s say $s(x, y)$ denotes the belongs to a head curve or not, let $D_x$ denoted the decision of pixel belonging to the head curve or not. $D_x$ is given as seed pixel:

$$\begin{align*}
D_x = \begin{cases} 
1, & E(x, y) = 1 \\
0, & \text{otherwise}
\end{cases}
\end{align*}$$

(33)

if the pixel belongs to the head curve then the curve is being labeled with the current label $L_b \in \{1, -1\}$ and last two pixel linked $s_k(x, y)$ are returned back where $k \in L, R$ representing the left and right pixels to be traced next and in the next step $s_k(x, y)$ are traced to determine whether they belong to pedestrian body or not. If the pixel doesn’t belong to the head curve then latter step is skipped, all the pixels labeled during the process are given previous value, and $(\text{number of times pixel is visited as pedestrian})$ is decremented by 1 and new seed-pixel $s(x, y)$ is selected according to eq. (33) i.e. if $D_x = 1$ then $s(x, y)$ is a seed-pixel.

1) Head Curve Detection:

This step takes a seed-pixel $s(x, y)$ as input. It has a heuristic search window of size $w_x \times w_y$ that finds the next two parallels pixel to be linked. The size of this search window determines the gap size to be considered on both vertical and horizontal axis. The algorithm divides the search for curve into two parts $(x_k, y_k)$ where $k \in L, R$ denotes the left and right part as shown in eq below.

$$\begin{align*}
(x_k, y_k) = \begin{cases} 
(s(x), s(y) + 1), & (x, y) + w, s(x) + w) \\
(\ldots (x, y) + w, s(x) + w, s(x) + w) & 
\end{cases}
\end{align*}$$

(34)

$$\begin{align*}
(x_k, y_k) = \begin{cases} 
(s(x) + 1, s(y)), & \ldots (s(x) + w, s(x) + w) \\
\ldots (s(x) + w, s(x) + w, s(x) + w) & 
\end{cases}
\end{align*}$$

(35)

Let $D_h$ denotes the decision whether the pixel should be linked to the head curve or not. The decision depends on two factor and first $(x_k, y_k) \neq 0$ and number of times the pixel is visited $v$ is less than 2 which means that each pixel is given two chances to be visited, considering it may belong to the exterior boundary of two overlapping regions.

$$\begin{align*}
D_h = \begin{cases} 
1, & (x_k, y_k) \neq 0, v < 2 \forall k \\
0, & \text{otherwise}
\end{cases}
\end{align*}$$

(36)

Let $D_R$ denotes the decision whether $(x_k, y_k)$ form a head curve or not

$$\begin{align*}
D_R = \begin{cases} 
1, & D_x = 1, D_x = 1 \\
0, & \text{otherwise}
\end{cases}
\end{align*}$$

(37)

If $(D_R) = 1$ then assign current label $L_b$ to the point $(x_k, y_k)$ and check for the left part of the head curve, else $(D_R) = 0$ if $(D_R) = 1$ and $(D_R) = 1$ then trace left part until $x_L = x_R$(to get parallel points). If $D_R = 1$ then it is a head curve $D_h = 1$ and the next seed-pixels $s_k(x, y) = (x_k, y_k) \forall k$.

2) Body Curve Detection

After the pixels $H(x_k, y_k)$ belonging to the head curve are labeled and $D_h = 1$, then the two pixels $s_k(x, y)$ are traced to find whether the head curve is followed by the body curve or not. Like head curve search, body search is also divided into two parts $(x_k, y_k)$ and $D_x$ denotes the left and right part as shown in eq below.

$$\begin{align*}
(x_k, y_k) = \begin{cases} 
(s_k(x) + 1, s_k(y) + M_i), & \ldots (x_k(x) + w, s_k(y) + w) \\
\ldots (s_k(x) + w, s_k(y) + w, s_k(x) + w) & 
\end{cases}
\end{align*}$$

(38)

$$\begin{align*}
(x_k, y_k) = \begin{cases} 
(s_k(x) + 1, s_k(y) + M_i), & \ldots (s_k(x) + w, s_k(y) + w) \\
\ldots (s_k(x) + w, s_k(y) + w, s_k(x) + w) & 
\end{cases}
\end{align*}$$

(39)

The head curve where right and left part are sequentially traced, in this module the left and right part of pedestrian body are traced in parallel and mid value is updated after the left and right deviates from the mid value. $M_i$ denotes the midpoint of $s_k(x, y)$ and $D_x$ denotes the absolute distance between $s_k(x, y)$ and $s_k(y)$ where $i$ denotes the body pixels. $D_B$ is final decision is dependent on $D_x$. The termination criterion is when no pixel is left to be linked or maximum height of the pedestrian is achieved or distance $D_x$ deviates from the mean distance $D_m$.

$$\begin{align*}
M_i = \frac{s_k(x) + s_k(x)}{2}, \frac{s_k(y) + s_k(y)}{2} \forall i \in P
\end{align*}$$

(40)

$$\begin{align*}
D_i = \text{abs}(s_k(y) - s_k(y))
\end{align*}$$

(41)

$$\begin{align*}
D_x = \begin{cases} 
1, & D_x = 1, D_x = 1 \\
0, & \text{otherwise}
\end{cases}
\end{align*}$$

(42)

$$\begin{align*}
D_m = \sum_{i=1}^{n} D
\end{align*}$$

(43)

Where $n$ denotes number of pixels linked. Fig 5 shows the complete procedure diagrammatically.
C. Appearance Based Features Classifier (APF)

Although ECD gives 95% accuracy but it gives a lot of false positives. In order to cater this, classical technique of pedestrian detection is applied. Appearance features refers to the appearance based features obtained without considering time domain. They are extracted from a single frame in case of video.

![Fig 5. Body curve detection](image)

The feature set selected by this work contains histogram of oriented gradient (HOG) [1] because it is considered most successful appearance based feature so far, along with other contour based features the compare ellipse fitting parameters[14] and straightness of the object. Let $E_3^2$ denotes the orientation of the ellipse, $E_2^3$ denotes the eccentricity of the ellipse calculated using moments of $M^4$ calculated similarly as eq 17-19, the difference is instead of blob it is calculated on whole heuristic window and $st^4$ denotes the straightness of image and s being calculated using principle component analysis (PCA) [21]Here $i$ denotes the curve under consideration, $(x, y)$ denotes the pixels having the same label and $(x', y')$ denotes the center of the region. Let $E_g$ denotes the eigen matrix, $E_g(1)$ denote the rows and $E_g(2)$ denotes the columns of $E_g$, the columns determine the vectors pointing along and across the object of interest. $C_y$ denotes the covariance along each component. $E_g(1)$ Compared to $E_g(2)$ is a measure of the aspect ratio of a the rectangle containing the object. Where $N$ denotes the number of curve pixels. Linear support vector machine (SVM) is used as classifier.

$$C_y = \sum \frac{(x-x')(y-y')}{N}$$ (44)

$$st^4 = \frac{E_g^2(2)}{\sum E_g}$$ (45)

We tested our dataset on the self-created dataset of 35x20 size images some images are shown in fig 6, NICTA pedestrian dataset of 8x20 size images [16] and ut-tower dataset [18]. Caltech [17] and INRIA [1] datasets are not suitable for our proposed work because the top view is not captured in the datasets and low resolution cameras are not used.

IV. EXPERIMENTAL RESULTS AND COMPARISON

Experimentation results shown are processed CCTV footage of complete 3 days obtained from three different locations varying illumination conditions (morning, noon and evening) and collected at different instant of time (different frames of same CCTV footage) and also some low resolution surveillance video available online. Result are evaluated by comparing the results of proposed algorithm with ground truth as comparison with the current state of art is not possible directly as most of these research works [1]-[15] consider pedestrian size to be greater than 100 pixels and of medium resolution i.e. greater than 600 x 600. The results are studied on 125 images taken from these different scenarios using two accurate measures: FP (false positive) and FN (false negative) in fig 10 and second by finding the computational time shown graphically in fig 11.

V. CONCLUSION

We have presented a technique to detect pedestrian in low resolution images and video of size between 30-80 pixels in triad manner targeting all levels of detection from low level edgelets based contour detection (ecd), to medium level appearance detection and high level motion detection. Medium to small size pedestrian detection in low resolution cameras captured from the top view is a challenging task especially because of noise accumulation and lack of pixel detail. Our experiments indicate that edgelet based contour detection (ecd) along with appearance and motion features can provide a new dimensionality to the research on small to medium scale object detection and is a powerful representation of pedestrian which can be combined into existing lines of research.

![Fig 7 Testing video samples obtained at different time and day. Yellow box indicates the heuristic search windows obtained from probabilistic voting of Bs and ECD modules, blue box shows the heuristic window with highest probability while green box show heuristic window the is classified as pedestrian by the SVM classifier](image)
Fig 8. Overall results (red shows probabilistic voting assign a high probability to this heuristic window and green shows the vote of appearance features

Fig 9. Testing video samples obtained at different time and day. Green box shows the overall detection achieved by integrating probabilistic voting and classifier based detection.

Fig 10. Frame No versus FP and FN. Where green shows the FP and red shows the FN trend shows the FN trend.

REFERENCES


Fig 11. Frame NO versus Computational Time over 125 frames


[22] Rodrigo Benenson, Mohamed Omran, Jan Hosang , Bernt Schiele “Ten Years of Pedestrian Detection, What Have We Learned?” ECCV 2014.