between [-180°,180°]. Since unsigned orientations are desired for this implementation, the values which are less than 0° is added up with 180°.

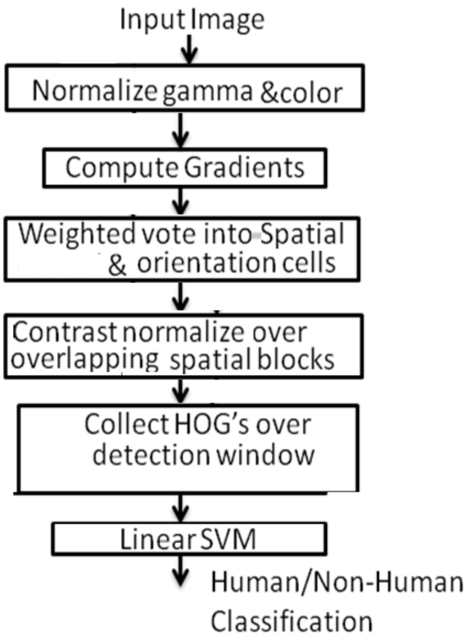

Figure 4: HOG algorithm

B. Orientation Binning

The next step is to compute histograms for each cell. Cell histograms later use at descriptor blocks. 8x8 pixel size cells are computed with 9 orientation bins for [0°,180°] interval. For each pixel's orientation, the corresponding orientation bin is found and the orientation's magnitude |***G***| is voted to this bin.

C. Descriptor Blocks

To normalize the cells orientation histograms, they should be grouped into blocks. From the two main block geometries, the implementation uses R-HOG geometry. Each R-HOG block has 2x2 cells and adjacent R-HOGs are overlapping each other for a magnitude of half-size of a block i.e., 50%.

D. Block Normalization

Although there are three different methods for block normalization, L2-Norm normalization is implemented using the formulae [1][9]: $V \rightarrow V/\sqrt{V_2^2 + \epsilon^2}$

E. Classifier

In machine learning, support vector machines (SVMs, also support vector networks) are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis [12]. Given a set of training examples, each marked as belonging to one of two classes, an SVM training algorithm builds a model that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier. When a test image given to the classifier [13]. It extracts the Histogram of oriented features and compute the distance between linear discriminant function (vector) and computed feature vector. If the distance is greater than zero(>0), human detected in the image otherwise not.

5. Experimental Results

Human detection system is successfully implemented on DM3730 by taking input as image (.jpg) and giving output as image with rectangular bounded box when human is detected. An operating system Ubuntu 12.04 is ported on to the Beagleboard-xM with DM3730 processor [10]. A Linux kernel image (uImage) is created using Linux kernel 2.6.32 which is compatible with DM3730. USB Webcam and keyboard devices are interfaced with Beagleboard-xM through USB ports. Monitor is connected to Beagleboard-xM through HDMI/DVI-D. Figure 7 shows the hardware setup of the Beagleboard-xM DM3730 with connections and µSD card. After Ubuntu OS loaded, enter the commands to initialize the webcam, capture the image and display the output result.

$sudo fswebcam –r 320*240 –S 20 test.jpg
$sudo ./humandetection test.jpg
$sudo fbi detection.jpg

Figure 5: Hardware setup

In the above commands, the first command is to initialize the webcam and capture the image of 320*240 pixel size. Second is to run the .exe file (here humandetection is the .exe file of source code). Third is to display the output result.

Figure 6: Output of the project

5.1 Small Training Dataset and Analysis

A small dataset for training is constituted from 25 positive and 25 negative images.

5.2 Testing Results

Table 1: Testing results on small training dataset

Test Image Database (Total No.of Positive images:288)	
Human	229
No Human	59 (0.208) false negative
Test Image database (Total No.of Negative Images:454)	
Human	34 (0.07) false positive
No human	420

5.3 Large Training dataset and Analysis

A large dataset for training is constituted from 100 positive and 100 negative images.

Testing results:

Table 2 Testing results on small training dataset

Test Image Database (Total No.of Positive images:288)	
Human	268
No Human	18(0.04) false negative
Test Image database (Total No.of Negative Images:454)	
Human	10 (0.02) false positive
No human	434

5.4 Performance Measures:

Precision and recall measures are widely used for evaluation of the classification tasks. They are defined as follows:

$$p = \frac{TP}{TP+FP}, \qquad r = \frac{TP}{TP+FN}.$$

Precision p is the number of correctly classified positive examples divided by the total number of examples that are classified as positive. Recall r is the number of correctly classified positive examples divided by the total number of actual positive examples in the test set.

Where,

TP: The no.of correct classifications of the positive examples (True Positive)

FN: The no.of incorrect classifications of the positive examples (False Negative)

FP: The no.of incorrect classifications of the negative examples (False Nositive)

TN: The no.of correct classifications of the negative examples (True Negative)

Table 3 Performance measures

	Classified Positive	Classified Negative
Actual Positive	0.93 (TP)	0.04 (FN)
Actual Negative	0.02 (FP)	0.95 (TN)

precision p = 0.978
recall r = 0.958 (from formulae)
accuracy = (TP+TN)/(TP+TN+FP+FN)
 = 0.97

Table 4 represents the execution times on Intel PC and DM3730 Processor with two different test image sizes. From the table results we can analyze that the algorithm is optimized on hardware with 1GHz processor and 512MB of RAM.

Table 4 Comparisons between HOG with SVM and SIFT

Execution Time (640*480px)	On PC (Intel pentium, 2Gb RAM)	On Board (1Ghz processor,512Mb RAM)
	16-21ms	100-150ms
Execution time (320*240px)	On PC	On Board
	8-15ms	70-85ms

Table 5 represents the comparison between performance measures of proposed algorithm i.e., HOG with pre trained SVM features and SIFT features. The proposed algorithm shows the best results interns of precision, recall and accuracy.

Table 5 Comparison of performance measures

	HOG+SVM	SIFT
Precision	0.978	0.77
Recall	0.958	0.82
Accuracy	0.97	0.8

Analysis:

The results assist in assessing the increase in performance of the HOG descriptor, when a large dataset is used for training the classifier. A change froms a smaller training set to a larger training set, while using the histogram of HOG output vector, certainly increases the evaluation parameters precision and recall. Comparison results shows that HOG descriptor performs well in Human detection than SIFT features.

6. Conclusion & Future Scope

To detect the people in static images HOG is an overall suitable feature descriptor, since it can describe an object without the need to detect smaller, individual parts of a person (i.e. the face). Additionally, the HOG features of an image are not affected by varying illumination conditions. In this project, an implementation of the Histogram of Oriented Gradients in static images has been described. A dataset was built for training an SVM classifier.

We have successfully implemented Human detection system on Beagleboard-xM DM3730 which is very useful and basic step in Automatic digital content management, Video surveillance, Driver assistance systems and can be used in many other applications. The test results showed that the detecting humans in the static images using HOG-SVM is quite satisfactory

Possible extensions to this study include expanding the dataset used by the SVM with additional images. The amount of errors in the detection process can be further reduced by adding more training images. The use of pre-processing algorithms to eliminate or reduce the amount of noise in the images will also improve the detection process.

References

[1] Navneet Dalal , Bill Triggs, “Histograms of Oriented Gradients for Human Detection,” in Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1, p.886-893, June 20-26, 2005.

[2] David G. Lowe, “Distinctive image features from scale-invariant keypoints,” International journal of computer vision, 60, 2004.
[3] P. Viola, M. J. Jones, and D. Snow, “Detecting pedestrians using patterns of motion and appearance,” *The 9th ICCV, Nice, France*, volume 1, pages 734–741, 2003.
[4] Andriluka, M., Roth, S., & Schiele, B. (2008), “People-tracking-by-detection and people-detection-by-tracking,” In IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 1{8).
[5] V. de Poortere, J. Cant, B. Van den Bosch, J. de Prins, F. Fransens, and L. Van Gool., “Efficient pedestrian detection: a test case for svm based categorization,” in *Workshop on Cognitive Vision*, 2002.
[6] R.C. Gonzalez and R. E, Woods, *Digital Image Processing*, Pearson Education, 3rd Edition, 2008.
[7] G. Bradski and A. Kaehler, *Learning OpenCV*, OReilly Publications,2008.
[8] *BeagleBoard-xM Rev C System Reference Manual*, http://beagleboard.org, Revision 1.0, April 4, 2010.
[9] N.Dalal, B.Triggs and C.Shmid, “Human detection using oriented histograms of flow and appearance,” *in proc European Conference on Computer Vision*, 2006.
[10] www.ti.com
[11] M. Enzweiler and D.M. Gavrila, “A mixed generative-discriminative framework for pedestrian classification,” IEEE Int’l Conf. Computer Vision and Pattern Recognition, 2008.
[12] O.Tuzel, F.Porikli, and P.Meer, “Human detection via classification on Riemannian manifolds,” IEEE Int’l Conf. Computer Vision and Pattern Recognition, 2007.
[13] E. Osuna, R. Freund and F. Girosi, “An Improved Training Algorithm for Support Vector Machines,” To appear *in Proc. of IEEE NNSP’97*, Amelia Island, FL, 24-26 Sep., 1997.

Author Profile

Amaraneni Srilaxmi, received B.Tech. Degree in Electronics and Communications Engineering from Anurag Engineering College., affiliated to JNTU Hyderabad in 2012. She is pursuing M.Tech in Embedded systems at VNR VignanaJyothi Institute of Engineering & Technology, Bachupally, Hyderabad. Her research interests include Image Processing and Embedded systems

Shaik Khadar sharif, received the B.E and M.E. degrees in Electronics and communications engineering from Osmania University in 1991 and 2003 respectively. During 1995-2000, he stayed in System integration Division, Research Center Imarat(RCI), Min. of Defence, Govt of India, Hyderabad. He is having 22 years of industry and acadeic experience. His research interest includes Signal and Image processing. Presently he is working as Associate professor in the Department of ECE, VNRVJIET, Hyderabad, since 2003.