IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 20, Issue 4, Ver. II (Jul - Aug 2018), PP 01-11
www.iosrjournals.org

Hog Feature - A Survey

Suman Mondal[1]
[1](*Department of Computer Science, Yogoda Satsanga Palpara Mahavidyalaya/ Vidyasagar University, India*)
Corresponding Author: Suman Mondal

Abstract: *Detecting a specified object in a given image is a difficult task due to its possible widely variable appearance. An essential requirement for this task is a robust feature set that allows the appearance of the object in the image to be discriminated cleanly, even in complex backgrounds under all possible illumination effects. Study of the existing literature shows that Histogram Oriented Gradient(HOG) feature is one of the most powerful feature for object recognition in scene images. In this report, we made a survey of the use of HOG feature in image classification tasks. This report also covers different variations of HOG features proposed at a later period. Results of the use of HOG feature along with Support Vector Machine (SVM) in scene text recognition tasks on several standard databases like ICDAR2003 robust reading dataset, Street View Text (SVT) etc. are available in the literature and these are comparable with the classification results provided by the state-of-the-art OCR software.*

Keywords: *HOG, Co-HOG, PHOG, Word-HOG*

Date of Submission: 09-07-2018 Date of acceptance: 25-07-2018

I. Introduction

Recognition of the text in natural scenes has attracted increasing research attention in recent years due to its crucial importance in scene understanding. It has become a very promising tool in different applications such as unmanned vehicle/robot navigation, living aids for visually impaired persons, content based image retrieval, etc. Though the optical character recognition (OCR) of scanned document images has achieved great success, recognition of the scene text by using existing OCR systems still has a large space for improvements due to a number of factors. First, unlike scanned document where texts that usually appear against a simple background having uniform color, texture and controlled lighting condition, scene texts often have a much more variation in the background that could have arbitrary color, texture, and lighting conditions as illustrated in Fig. 1. Second, unlike scanned document texts that are usually printed in some widely used text font and text size, the scene texts usually appear in arbitrary size and often printed in some fancy but infrequently used text fonts as illustrated in Fig. 1. Even worse, the font of the scene text may even change within a single word, for the purpose of special visual effects or attraction of the human attention. Third, unlike scanned document texts that usually have a fronto-parallel view, scene texts captured from arbitrary viewpoints often suffer from the perspective distortion as illustrated in Fig. 1. All these variations make OCR of scene texts a very challenging task. A robust OCR technique is urgently needed that is tolerant to the variations of scene texts as well as their background.

Existing Scene text recognition approaches vary widely. However, these approaches usually perform certain preprocessing operations like binarization, slant correction, perspective rectification before passing the processed text portion to the OCR engine. Chen et al. [1] performed a variant of adaptive binarization algorithm [2] on the detected text region before passing to OCR for recognition. An iterative binarization method based on k-means algorithm was proposed in [3] for single character images producing a set of potential binarized characters and then Support Vector Machines (SVM)[4][5][6] was used to measure the degree of character likeness and the one with maximum character-likeness is selected as the optimal result.

II. Literature Survey

In the literature, there exists various features for recognition of scene texts, human, and other objects. Among them HOG feature has shown the better performance than others. After reviewing existing edge and gradient based descriptors, available experimental results show that grids of Histograms of Oriented Gradient (HOG) [7] descriptors significantly outperform existing feature sets for human detection [8] and text classification [9]. For face recognition [10][8], HOG feature is very adorable for recognizing the face. In a recent research, Gradient Field HOG [11] is more suitable Sketch Based Image Retrieval(SBIR). GF-HOG is shown to consistently outperform retrieval versus SIFT[12], multi-resolution HOG, Self Similarity, Shape Context and Structure Tensor.Most of these object-recognition-based works simply use off-the-shelf HOG-like features for character recognition at the first stage. A closer look shows that HOG divides input image data into several equally spaced square grids and then extract oriented gradient information in those predefined sub-regions. Several other research areas have utilized low-level image features computed using linear and non-linear transformations of the input image [13] and then extract midlevel feature representations using sub-regions based pooling schema. Perona and Malik [14] used Gaussian smoothing kernel to perform spatial integration/pooling from lowlevel features in early 90s. Methods in [15][16] aggregated low-level statistics via histogram representation. Viola and Jones [17] proposed integral image technique to compute more expensive bandpass kernels for Haar features. The idea of using integral image to naturally integrate different sources of low-level information has been executed in several systems such as object recognition [18], image categorization [19], and pedestrian detection [10][20], etc. Such feature mining approaches try to automatically find meaningful feature spaces to improve the system performance.

Fig. 1: Example characters taken from ICDAR2003 (first and second rows) and SVT (third and fourth rows) datasets. First row: 'E', 'S', 'S', 'N', 'N', 'G', 'G', 'R', 'A'. Second row: 'A', 'A', 'f', 'M', 'H', 'R', 'T', 'T'. Third row: 'E', 'S', 'L', 'M', 'b', 'J', 'o', 'R', 'R'. Fourth row: 'P', 'M', 'K', 'E', 'h', 'M', 'T', 'A', 'n'.

Text detector is based on the cascade of boosted ensemble. A novel contribution of feature complexity in AdaBoost feature selection algorithm is stated in [21]. In case of heterogeneous feature set, the integration of feature complexity in feature selection algorithm helps in reducing the overall complexity of strong classifier and also the computational load which is an important consideration in real time applications.

For the color image, it is first converted into the gray-level image, on which image pyramids are built with nearest interpolation to capture texts with various sizes. Motivated by the work [22], a text region detector is designed by integrating Histograms of Oriented Gradients (HOG) feature extractor and boosted cascade classifier. For each local region in one image of pyramids,HOG features are extracted as an input to a variation of cascade boosting classifier, WaldBoost [23], to estimate whether this region contains texts. The major difference between WaldBoost and other cascade boosting classifiers is that it directly ensemble weak learners to build a strong classifier and each of them can be used to filter out negative objects individually. Now we address the problem of combining LSHOG [24] and LSLBP [24] and training part based model with learnt LSHOG-LSLBP [24]. This work is different from [25], in which a rigid template model is trained for human detection using concatenated basic HOG-LBP. Many popular methods have been proposed to tackle the feature combination problem. They are Multiple Kernel Learning [26][27], Boosting [28] and subspace learning [29], etc. HOG feature is also used in selecting Gene selection for cancer classification in [30].

After calculating the HOG feature, we need to classify text, human or object. For this, we observe that SVM(Support Vector Machine) [31][32][33] classifier is used for machine learning [34] over the decades. For multi-class classification, LIBSVM [35] with HOG feature can get the better result.

III. Different Types of HOG Features

A. HOG Feature

The HOG (Histogram Oriented Gradient) [7] was originally designed for the task of human detection. Similar to SIFT [12], HOG computes a histogram of gradient orientations in a certain local region. One of the main differences between SIFT and HOG is that HOG normalizes such histograms in overlapping local blocks and makes a redundant expression. Another difference is that SIFT describes the scale and orientation normalized image patch around the specially deleted key point,

while HOG is computed in a rigid rectangular window without scale/orientation normalization. In the HOG calculation, first we have to divide the image window into smaller rectangular regions called cells. Suppose we divide the image into HW cells. Second, we must decide the number of the bins into which the weighted votes of the gradient vectors should be accumulated. In HOG, the orientation bins are evenly spaced over 0-180^0 ("unsigned gradient") or 0-360^0 ("signed gradient"). Note that SIFT always uses 8 bins over 0-360^0. Let π denote the number of the bins over orientation. To make the descriptor robust to small deformation, tri-linear interpolation should be applied. Thus we could obtain the histogram with $HW\pi$ bins. HOG does not use this histogram as a descriptor. Instead, HOG uses "Block Normalization". A block is defined as a group of hw cells. The block slides inside the window image, that means $(H - h + 1)(W - w + 1)$ unique blocks exist. The HOG descriptor is a concatenation of the normalized block descriptors. Block descriptors are $hw\pi$ dimensional vectors each of which is a concatenation of histogram components of the cells. Consequently, HOG descriptor has $(H - h + 1)(W - w + 1)$ hw# dimensionalities. This is a redundant expression in a sense that $HW\pi$ components in the original histogram composes a vector with $(H - h + 1)(W - w + 1)$ $hw\pi$ dimensions. This redundancy is the major characteristic of the HOG feature.

B. Slit Style HOG Feature

On modification of HOG feature, Slit Style HOG feature is introduced for word spotting task. Different from the human detection window used in [7], window image (called "slit image") is a narrow rectangle as in Fig. 2. Here, it restricts the width of the block to be the same as the width of the slit. The horizontal overlapping of the original HOG could be well relized by the sliding window and sequential representation of the vectors. Fig. 3 represents the relationship between slit, blocks and cells in this feature extraction method. Suppose that there is a slit image denoted as S1. The slit image is divided into H×Wcellsash11 ,. . . ,h24 in the figure. In this case, H = 4 and W = 2. Then we define the block as HW group of cells. The width is limited to W; which is the difference to the original HOG. In the figure, it sets the block size as 2×2. So, in this case we have 3 blocks as b11, b12, b13 with each block composed of 4 cells. So, the dimensionality of slit style HOG feature becomes $3\times4\times\pi$ for this case.

Fig. 2: Example of Window

Unlike the reference [7] used the unsigned gradient, here it uses the signed gradient for the orientation binning. The reason why unsigned showed better result in human detection is estimated that the clothes of the human are sometimes brighter and sometimes darker than the background. The detection system cannot determine which model should be used in advance. We cannot imagine the manuscripts in which characters darker than the background and characters brighter than the background are mixed. Therefore, a signed gradient should be appropriate rather than an unsigned gradient.

C. Word-HOGs

The Word-HOGs is a descriptor that is based on gradient orientation histograms. The descriptor is a concatenation of gradient orientation histograms from sub-blocks within a word patch. From the Word-HOG descriptor, we generate SIFT-like [12] descriptors, which is known as WSIFT, and use the Vocabulary Tree

(VT)-based approach [36] to perform word patch matching. Word-HOG is similar to [37] [38] , however, we differ in how gradient orientation histograms are generated and how word patch matching is performed. It uses lattice coding [39] to quantize the descriptor and use a context-based arithmetic coder to compress the query. Compressed Word-HOGs performs word patch matching at a high accuracy with only a few tens

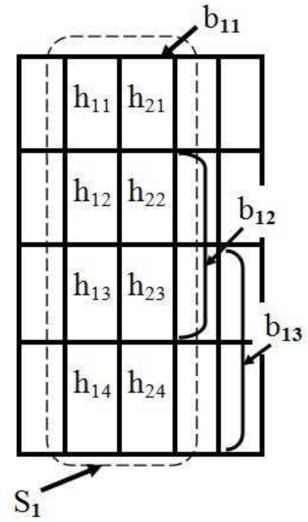
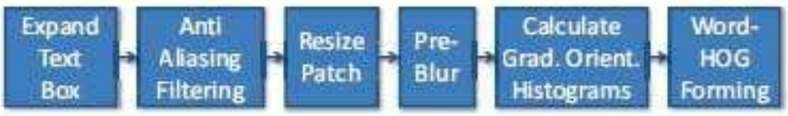
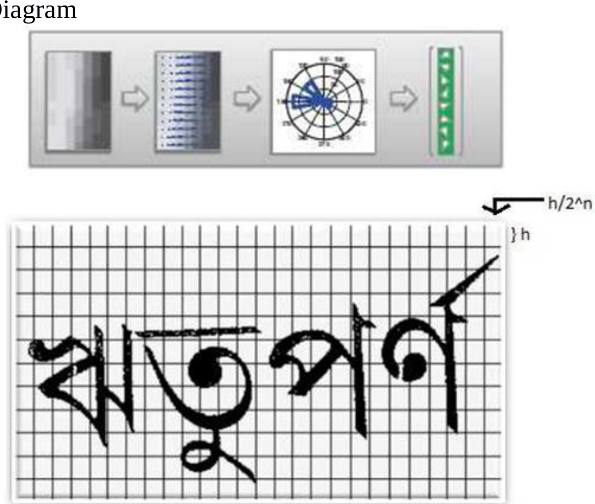


Fig. 3: Block Normalization for SSHOG

of bytes. Fig. 4a shows a diagram of the Word-HOG extraction process. Given an image, we assume that a rectangular box is tightly fit around the text, where the box is typically found using a text detection algorithm [40]. The box is expanded by a factor of 1/m, and a word patch is extracted from the location of the expanded box in the image. Then, the patch is resized to a height of H pixels while the aspect ratio is kept the same. Pre-blurring is applied to the word patch using a Gaussian filter, similar to [12]. After that, the patch is divided vertically into 4 rows, as shown in Fig. 4b, and each row is horizontally divided into sub-blocks that are $1/2^n$ of the row height, where n is an integer and greater than 0. For each sub-block, image gradients are calculated and used to generate a gradient orientation histogram with 8 bins. Both the number of rows and the dimensions of the histogram were determined from patch matching experiments. Once the gradient orientation histogram of each sub-block is generated, the Word-HOG is formed by concatenating the histograms starting from the top-left sub-block and following an order from top to bottom and left to right.



(a) Word HOG Extraction Diagram



(b) Gradient histograms are extracted from sub-blocks

Fig. 4: The Word-HOG descriptor is generated from gradient orientation Histograms of sub-blocks in the word patch.

The Word-HOG descriptor is a variable length descriptor; its length depends on the width of the word patch. Additionally, since the descriptor is formed by gradient orientation histograms, the descriptor can be efficiently compressed using lattice coding techniques [39] [41] as described in the next section.

1) Lossy Compression of Word-HOGs: To compress the Word-HOG descriptor, lattice coding is first used to lossily compress the histogram of each sub-block. Then, context based arithmetic coding is used to encode the lattice indices.

2) Lattice Coding: The gradient orientation histogram is the distribution of the image gradient directions within a sub-block. For a normalized gradient orientation histogram of dimension m, the histogram vector lies on a probability simplex in m-dimensional space. Lattice coding quantizes these histogram vectors to lattice points in the probability simplex using the method described in [41], where a quantization parameter, n, controls the density of points on the simplex. The lattice points within the probability simplex are enumerated, hence, only an index is needed to represent the quantized distribution.

3) Context-based Arithmetic Coding: Once the Word-HOG descriptor is quantized using lattice coding, a set of lattice indices is produced. These indices are compressed using entropy coding to further reduce the query size. Since the sub-block size is typically smaller than the character strokes, a single stroke would appear in several sub-blocks. For sub-blocks that contain the same stroke, the direction of the image gradients is similar. We exploit this relationship in sub-blocks by using context-based arithmetic coding. Starting from the top-left, the indices of each sub-block are entropy coded while using the previously coded horizontal sub-block as context, as shown in Fig. 5

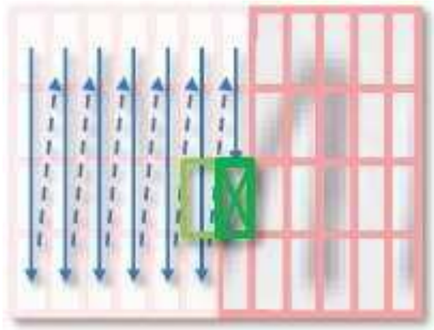


Fig. 5: The horizontal neighboring sub-block is used as context for the context-based arithmetic coder for Word-HOGs.

D. Co-occurrence of Histogram of Oriented Gradients

Co-HOG [42] captures spatial information by counting frequency of co-occurrence of oriented gradients between pixel pairs. Thus relative locations are stored. The relative locations are reflected by the offset between 2 pixels as shown in Fig. 6a. The yellow pixel in the center is the pixel under study and the neighboring blue ones are pixels with different offsets. Each neighboring pixel in blue color forms an orientation pair with the center yellow pixel and accordingly votes to the co-occurrence matrix as illustrated in Fig. 6b. Therefore, HOG is just a special case of Co-HOG when the offset is set to (0;0), i.e., only the pixel under study is counted. The frequency of co-occurrence of oriented gradients is captured at each offset via co-occurrence matrix as shown in Fig. 6b. Co-occurrence matrix at a specific offset (x; y) is given by:

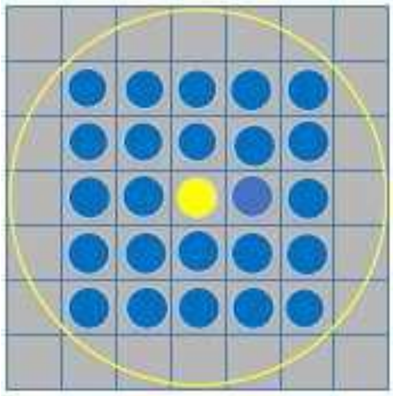
$H_{x,y}(i, j) = 1 \ \Sigma_{(p,q)\varepsilon B}$ if $o(p, q) = i \ o(p + x, q + y) = j$

$H_{x,y}(i, j) = 0$ otherwise (1)

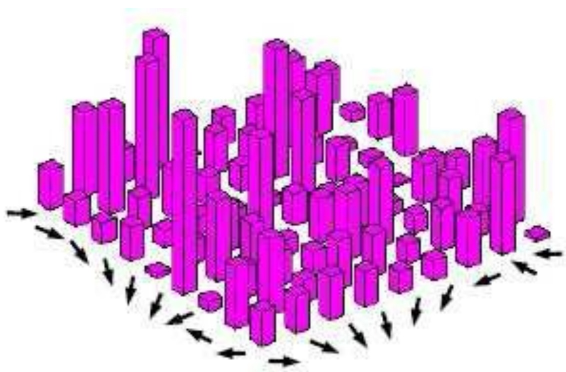
where $H_{x;y}$ is the co-occurrence matrix at offset (x; y), which is a square matrix and its dimension is decide by number of orientation bins. Therefore, we will have 24 co-occurrence matrix with offsets as illustrated in Fig. 6a. O is the gradient orientation of the input image I and B is a block in the image. Therefore, Equation 1 computes co-occurrence matrix in a block and Fig. 6b shows an example. The Co-HOG feature descriptor of an image can thus be constructed by vectorizing and concatenating the Co-HOG matrix of all blocks of the image under study.

The Co-HOG feature extraction process can be summarized in the following three steps.

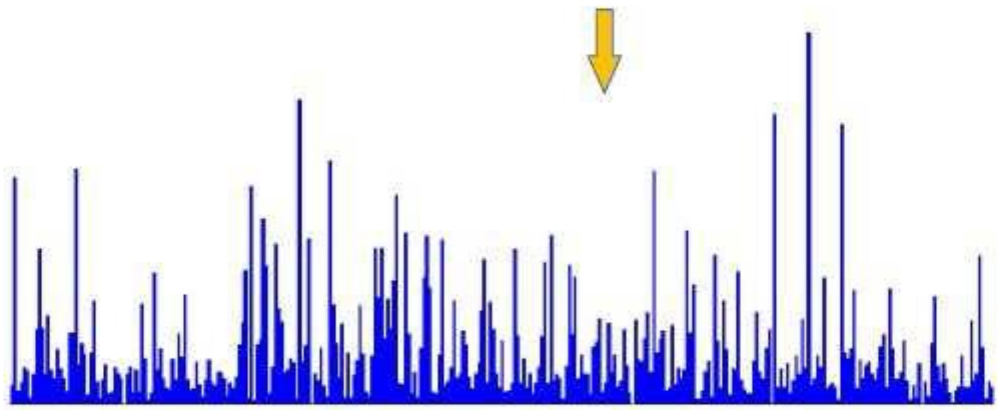
1) Gradient Magnitude and Orientation Computation:: Gradient magnitude is computed as an L2 norm of horizontal and vertical magnitude computed by Sobel filter. For color images, the gradient is computed separately for each color channel and the one with maximum magnitude is used. Gradient orientation ranges between 0^0-180^0 (unsigned gradient) and is quantized into 9 orientation bins.



(a) Offset in Co-HOG



(b) Co-occurrence matrix



(c) Vectorization

Fig. 6: Illustration of Co-HOG feature extraction: (a) illustrates the offset used in Co-HOG. (b) shows the co-occurrence of one block (c) shows the vectorization of co-occurrence matrix and concatenated one after another to form Co-HOG feature vector.

2) Weighted Voting : The original Co-HOG is computed without weighting as specified in Equation 1 [43], which by itself can not reflect the difference between strong gradient and weak gradient pixels. We propose to add in a weighting mechanism based on the gradient magnitude where bi-linear interpolation is employed to vote between two neighboring orientation bins. Equation 1 shows how the weighting of gradient magnitude and orientation bin is combined and Fig. 7 gives a simple illustration.

$$H(\Theta_1,\Theta_3) \leftarrow H(\Theta_1,\Theta_3) + M_1\,(1 - (\alpha - \Theta_1) / (\Theta_2 - \Theta_1)) + M_2\,(1 - (\beta - \Theta_3) / (\Theta_4 - \Theta_1))$$
$$H(\Theta_1,\Theta_4) \leftarrow H(\Theta_1,\Theta_4) + M_1\,(1 - (\alpha - \Theta_1) / (\Theta_2 - \Theta_1)) + M_2\,(1 - (\beta - \Theta_3) / (\Theta_4 - \Theta_1))$$

$$H(\Theta_2,\Theta_3) \leftarrow H(\Theta_2,\Theta_3) + M_1\,(1 - (\alpha - \Theta_1) / (\Theta_2 - \Theta_1)) + M_2\,(1 - (\beta - \Theta_3) / (\Theta_4 - \Theta_1)) \quad (2)$$

$$H(\Theta_2,\Theta_4) \leftarrow H(\Theta_2,\Theta_4) + M_1\,(1 - (\alpha - \Theta_1) / (\Theta_2 - \Theta_1)) + M_2\,(1 - (\beta - \Theta_3) / (\Theta_4 - \Theta_1))$$

where H is the co-occurrence matrix at a specific offset as defined in Equation 2. M_1 is the gradient magnitude at location (p ; q) and a is its corresponding gradient orientation. M_2 is the gradient magnitude at location (p + x ; q + y) with corresponding gradient orientation . Θ_1 and Θ_2 denote the neighboring orientation bin centers of a, similar to Θ_3 and Θ_4 .
In the proposed weighting scheme, a pixel with very small gradient value can have a fair large weight if its pair pixel has a large gradient value. To avoid such situations, we do not count pixel pairs when at least one of them has very small gradient value.

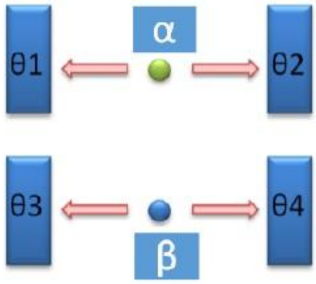


Fig. 7: Bi-linear interpolation of weighted magnitude.

3) Feature Vector Construction : The obtained block features are first normalized with L2 normalization method. The Co-HOG feature descriptor of whole image under study can then be constructed by concatenating all normalized block features.

E. Building a representation for face recognition using HOGs

For face recognition, in [10] it first normalizes faces and then extract HOG features from a regular grid. The grid is formed by placing equal side patches around a first cell centered in the image, until the whole image is covered.

On the other hand, the size of the patch used to extract the HOG features is important. In the best size for the patch was estimated via cross-validation in the Yale database, prior to using the FERET database for the final experiments. The locality of the extracted features is determined by the patch size.

Suppose R individual classifiers c_k (k = 1, . . . ,R) each one trained using HOG features are extracted from different patch sizes. Each classifier assigns one input sample (represented as x_k) to a label L_k ($L_k = w_1,. . . ,w_m$). Assume the classifier c_k gives every output a measurement which is represented as a posterior probability vector, $P_k = [p(w_1|x_k),...,p(w_m|x_k)]^t$, where $p(w_i|x_k)$ denotes the probability that the classifier considers that x was labeled with w_i. The product rule consists of fusing the final decision as:

$$j = \arg\max_i p(w_i) \Pi^R_{k=1} p(w_i \mid x_k) \quad (3)$$

Note that when several overlapping patches are used, the final feature representation will be highly redundant and if the classifier does not have any mechanism for feature selection it might severely suffer from over-fitting. Observe that the human face displays a structure common to all individuals. This implies that some gradient orientations would be very frequent in some specific zones of the face. Other orientations, on the contrary, would never or almost never appear in a given region. This reinforces the idea of the need for dimensionality reduction techniques.

F. Pyramid of Histogram of Oriented Gradients

To better present the spatial relationship of the oriented gradients, the Pyramid of Histogram of Oriented Gradients (PHOG) [44] was proposed for object categorization. Inspired by this work, we propose to adopt PHOG to extract more information by encoding HOG feature in a spatial pyramid. The main idea for PHOG is to represent the image shape and its spatial layout so that the correspondence between two shapes can be calculated by chi-square kernel as shown in Fig. 8.

For a given image, instead of having fixed cell size as in HOG, PHOG divides the image into different cells at different pyramid levels, with $2^l \times 2^l$ cells at the l^{th} level. Having a total dimension of $K.\ \Sigma_{l\,\varepsilon\,L} 4^l$, where K = 20 orientation bins, L is the total number of levels which is limited to no more than 3 to prevent over-fitting.

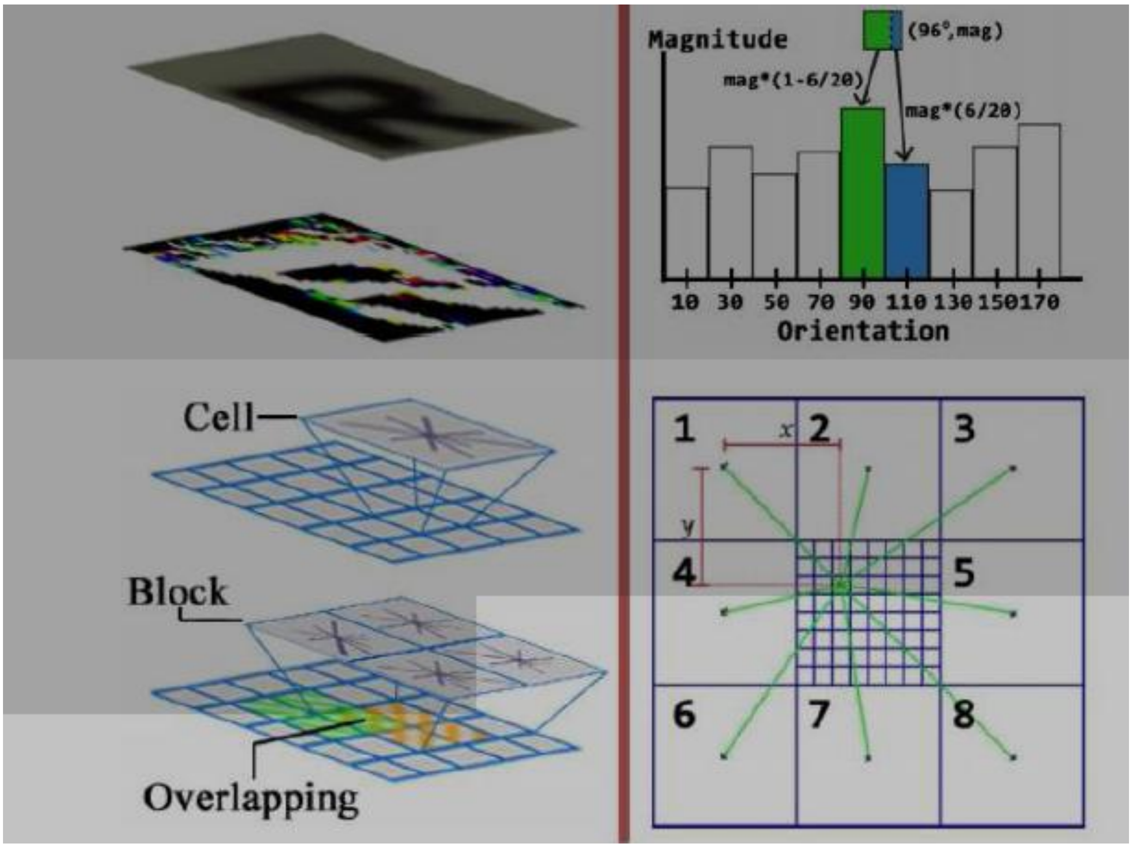


Fig. 8: Bi-linear interpolation of weighted magnitude.

It can be observed that PHOG features at each level for different characters are represented differently. The final feature is then normalized like in HOG to prevent unequal weighting for images with various illuminations and contrast. Since PHOG divides the cell into different resolution that gets increasingly smaller to form the pyramid structure, the pyramid structure at the lower resolutions allows for focus onto the region of interest. Specifically, on top of the orientation information captured by HOG, the spatial matching [45] of the pyramid structures allows for the geometric matching of the orientations at finer resolution. The variation made to the original PHOG method is to follow the original HOG method with the overlapping block normalization so that each cell can contribute to more than one component of the final feature where each cell is normalized to their respective different blocks [7]. Bilinear interpolation of the pixels is also implemented, where tri-linear interpolation kicks in when $l = 2$ as interpolation helps to decrease the artifacts and distortions. In addition, no weights specific to the levels were given, because although lower levels has less dissimilar features, but they already weigh less in terms of feature dimension, hence additional weights to penalize the lower levels will make these features meaningless; but the lower levels are useful as they serve to capture the more general outline of the image and is more robust to noises in the image compared to the higher levels at lower resolution.

IV. Experimental Results

The testing result in Table 1 shows that PHOG performed comparatively close to the result obtained from Co-HOG. One main advantage compared with Co-HOG is that PHOG has a much smaller feature dimension (4K vs 100K) but it extracts an almost equal amount of feature information. Additionally, the computation complexity is lower for PHOG over Co-HOG (O2 vs O3). This is crucial for realtime mobile applications which require less memory and processing time. Besides, the accuracy of all 2 datasets is also comparatively close to each other, especially for case insensitive testing, showing that the method is stable across different natural scene text images. One of the features that are hard to distinguish in natural scene text will be the upper and lower cases, especially for characters like c, p, z etc. where two distinctive light blue lines running parallel to the main line corresponds to the upper and lower casing classified wrongly into its counterpart. Hence, when tested without the case-sensitivity, the accuracy result shows marked improvement.Another feature that is hard to distinguish is for ambiguous characters like l (love), I (Indigo) and 1 (one) or O (orange) and 0 (zero) etc. It is made worst in natural scene text where characters come in various fonts which express these ambiguous characters differently. Such ambiguities can only be resolved if the context of the character is known, i.e. the whole word is needed.

The pre-processing segmentation is also very crucial to the recognition result. For the same image, if the segmentation is not appropriate, it can be labeled wrongly. if the orientation is very skewed when the original character already has a noisy background, or the original character has abnormal shape (e.g. extra-long tail of R) and the segmentation did not bound the character properly.

Table no 1 : Shows metabolic parameters of patients of the three groups before treatment.

Testing Method	Testing Dataset Accuracy (%)	
	ICDAR	SVT
ABBYY FineReader 10 [46]	26.6%	15.4%
GB + NN [47]	41.0%	–
HOG + NN [48]	51.5%	--
NATIVE + FERNS [49]	64.0%	--
MSER [50]	67.0%	--
HOG + SVM [51]	74.5%	61.9%
Co-HOG [42]	79.4%	75.4%
Co-HOG (case insensituve) [42]	83.6%	80.6%
PHOG (Linear Kernel) [52]	76.5%	76.4%

Still, the recognition for some disjoint characters or characters in noisy background performed well. These are hard to even visually recognize due to poor lighting or low contrast etc. Sometimes, the distortion could be due to the size of the original image, which might be very small. Hence, the resizing is constrained to 32×32 pixels, and by enlarging the images, it will inevitably lead to lose of image quality and hence affect the resultant feature.

V. Conclusion

The presented survey of HOG features are very useful for classifying the images and these are very easy to understand also.

These HOG features have several advantages -

• The HOG descriptor operates on localized cells, the method upholds invariance to geometric and photometric transformations, except for object orientation. Such changes would only appear in larger spatial regions.

• It captures edge or gradient structure that is very characteristic of local shape, and it does so in a local representation with an easily controllable degree of invariance to local geometric and photometric transformations: translations or rotations make little difference if they are much smaller that the local spatial or orientation bin size.

• For human detection, rather coarse spatial sampling, fine orientation sampling and strong local photometric normalization turns out to be the best strategy, presumably because it permits limbs and body segments to change appearance and move from side to side quite a lot provided that they maintain a roughly upright orientation.

• For INRIA human detection database, the C-HOG and R-HOG block descriptors perform comparatively, with the C-HOG descriptors maintaining a slight advantage in the detection miss rate at fixed false positive rates across both data sets.

References

[1]. X. Chen and A. L. Yuille, “Detecting and reading text in natural scenes,” in Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, ser. CVPR’04, 2004, pp. 366–373.

[2]. B. Bataineh, S. N. H. S. Abdullah, and K. Omar, “An adaptive local binarization method for document images based on a novel thresholding method and dynamic windows,” Pattern Recogn. Lett., vol. 32, no. 14, pp. 1805–1813, Oct. 2011.

[3]. T. Wakahara and K. Kita, “Binarization of color character strings in scene images using k-means clustering and support vector machines,” in Proceedings of the 2011 International Conference on Document Analysis and Recognition, ser. ICDAR ’11, 2011, pp. 274–278.

[4]. K. Crammer and Y. Singer, “On the algorithmic implementation of multiclass kernel-based vector machines,” J. Mach. Learn. Res., vol. 2, pp. 265–292, Mar. 2002.

[5]. C.-W. Hsu, C.-C. Chang, and C.-J. Lin, “A practical guide to support vector classification,” Tech. Rep., 2003.

[6]. C. J. C. Burges, “A tutorial on support vector machines for pattern recognition,” Data Min. Knowl. Discov., vol. 2, no. 2, pp. 121–167, Jun. 1998.

[7]. N. Dalal and B. 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 - Volume 01, ser. CVPR ’05, 2005, pp. 886–893.

[8]. Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan, “Fast human detection using a cascade of histograms of oriented gradients,” in Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2, ser. CVPR ’06, 2006, pp. 1491–1498.

[9]. S. Tong and D. Koller, “Support vector machine active learning with applications to text classification,” J. Mach. Learn. Res., vol. 2, pp. 45–66, Mar. 2002.

[10]. O. Deniz, G. Bueno, J. Salido, and F. De la Torre, “Face recognition using histograms of oriented gradients,” Pattern Recogn. Lett., vol. 32, no. 12, pp. 1598–1603, Sep. 2011.

[11]. R. Hu and J. Collomosse, "A performance evaluation of gradient field hog descriptor for sketch based image retrieval," Comput. Vis. Image Underst., vol. 117, no. 7, pp. 790–806, Jul. 2013.
[12]. D. G. Lowe, "Object recognition from local scale-invariant features," in Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2, ser. ICCV '99, 1999, pp. 1150–1157.
[13]. K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," Biological Cybernetics, vol. 36, pp. 193–202, 1980.
[14]. S. Thumfart, R. H. A. H. Jacobs, E. Lughofer, C. Eitzinger, F. W. Cornelissen, W. Groissboeck, and R. Richter, "Modeling human aesthetic perception of visual textures," ACM Trans. Appl. Percept., vol. 8, no. 4, pp. 27:1–27:29, Dec. 2008.
[15]. D. J. Heeger and J. R. Bergen, "Pyramid-based texture analysis/synthesis," in Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques, ser. SIGGRAPH '95, 1995, pp. 229–238.
[16]. J. Puzicha, T. Hofmann, and J. M. Buhmann, "Non-parametric similarity measures for unsupervised texture segmentation and image retrieval," in Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97), ser. CVPR '97, 1997, pp. 267–272.
[17]. P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," 2001, pp. 511–518.
[18]. Z. Tu, "Probabilistic boosting-tree: Learning discriminative models for classification, recognition, and clustering," in Proceedings of the Tenth IEEE International Conference on Computer Vision - Volume 2, ser. ICCV '05, 2005, pp. 1589–1596.
[19]. B. Yao, A. Khosla, and F.-F. Li, "Combining randomization and discrimination for fine-grained image categorization." IEEE, 2011, pp. 1577–1584.
[20]. H. Zhang and L. Zhao, "Integral channel features for particle filter based object tracking," in Proceedings of the 2013 5th International Conference on Intelligent Human-Machine Systems and Cybernetics - Volume 02, ser. IHMSC '13, 2013, pp. 190–193.
[21]. R. E. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated predictions," Mach. Learn., vol. 37, no. 3, pp. 297–336, Dec. 1999.
[22]. Y.-F. Pan, X. Hou, and C.-L. Liu, "A robust system to detect and localize texts in natural scene images," in Proceedings of the 2008 The Eighth IAPR International Workshop on Document Analysis Systems, ser. DAS '08, 2008, pp. 35–42.
[23]. J. Sochman and J. Matas, "Waldboost " learning for time constrained sequential detection," in Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 - Volume 02, ser. CVPR '05, 2005, pp. 150–156.
[24]. J. Zhang, K. Huang, Y. Yu, and T. Tan, "Boosted local structured hog-lbp for object localization," in Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, ser. CVPR '11, 2011, pp. 1393–1400.
[25]. X. Wang, T. X. Han, and S. Yan, "An hog-lbp human detector with partial occlusion handling," in ICCV'09, 2009, pp. 32–39.
[26]. M. Varma and B. R. Babu, "More generality in efficient multiple kernel learning," in Proceedings of the 26th Annual International Conference on Machine Learning, ser. ICML '09, 2009, pp. 1065–1072.
[27]. A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman, "Multiple kernels for object detection," in ICCV'09, 2009, pp. 606–613.
[28]. D. Mease and A. Wyner, "Evidence contrary to the statistical view of boosting," J. Mach. Learn. Res., vol. 9, pp. 131–156, Jun. 2008.
[29]. S.-u. Hussain and B. Triggs, "Feature sets and dimensionality reduction for visual object detection," in Proceedings of the British Machine Vision Conference, 2010, pp. 112.1–112.10.
[30]. I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," Mach. Learn., vol. 46, no. 1-3, pp. 389–422, Mar. 2002.
[31]. H.-T. Lin, C.-J. Lin, and R. C. Weng, "A note on platt's probabilistic outputs for support vector machines," Mach. Learn., vol. 68, no. 3, pp. 267–276, Oct. 2007.
[32]. C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," Trans. Neur. Netw., vol. 13, no. 2, pp. 415–425, Mar. 2002.
[33]. B. Scholkopf and A. J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. Cambridge, MA, USA: MIT Press, 2001.
[34]. T. Hofmann, B. Sch¨olkopf, and A. J. Smola, "Kernel methods in machine learning," Annals of Statistics, vol. 36, no. 3, pp. 1171–1220, 2008.
[35]. C.-C. Chang and C.-J. Lin, "Libsvm: A library for support vector machines," ACM Trans. Intell. Syst. Technol., vol. 2, no. 3, pp. 27:1–27:27, May 2011.
[36]. D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2, ser. CVPR '06, 2006, pp. 2161–2168.
[37]. M. Khayyat, L. Lam, and C. Y. Suen, "Learning-based word spotting system for arabic handwritten documents," Pattern Recogn., vol. 47, no. 3, pp. 1021–1030, Mar. 2014.
[38]. K. Terasawa and Y. Tanaka, "Slit style hog feature for document image word spotting," in Proceedings of the 2009 10th International Conference on Document Analysis and Recognition, ser. ICDAR '09, 2009, pp. 116–120.
[39]. V. Chandrasekhar, G. Takacs, D. M. Chen, S. S. Tsai, Y. Reznik, R. Grzeszczuk, and B. Girod, "Chog: Compressed histogram of gradients a low bit-rate feature descriptor," Int. J. Comput. Vision, pp. 2504–2511, 2009.
[40]. H. Chen, S. S. Tsai, G. Schroth, D. M. Chen, R. Grzeszczuk, and B. Girod, "Robust text detection in natural images with edge-enhanced maximally stable extremal regions," J, pp. 2609–2612, Sep 2011.
[41]. V. Chandrasekhar, G. Takacs, D. M. Chen, S. S. Tsai, Y. Reznik, R. Grzeszczuk, and B. G., "Compressed histogram of gradients: A low-bitrate descriptor," Int. J. Comput. Vision, vol. 96, no. 3, pp. 384–399, Feb. 2012.
[42]. S. Tian, S. Lu, B. Su, and C. L. Tan, "Scene text recognition using co-occurrence of histogram of oriented gradients," in Proceedings of the 2013 12th International Conference on Document Analysis and Recognition, ser. ICDAR '13, 2013, pp. 912–916.
[43]. T. Watanabe, S. Ito, and K. Yokoi, "Co-occurrence histograms of oriented gradients for pedestrian detection," in Proceedings of the 3rd Pacific Rim Symposium on Advances in Image and Video Technology, ser. PSIVT '09, 2008, pp. 37–47.
[44]. A. Bosch, A. Zisserman, and X. Munoz, "Representing shape with a spatial pyramid kernel," in Proceedings of the 6th ACM International Conference on Image and Video Retrieval, ser. CIVR '07, 2007, pp. 401–408.
[45]. S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2, ser. CVPR '06, 2006, pp. 2169–2178.
[46]. "Abbyy finereader 10." [Online]. Available: http://www.abbyy.com/

[47]. A. Coates, B. Carpenter, C. Case, S. Satheesh, B. Suresh, T. Wang, D. J. Wu, and A. Y. Ng, "Text detection and character recognition in scene images with unsupervised feature learning," in Proceedings of the 2011 International Conference on Document Analysis and Recognition, ser. ICDAR '11, 2011, pp. 440–445.

[48]. K. Wang and S. Belongie, "Word spotting in the wild," in Proceedings of the 11th European Conference on Computer Vision: Part I, ser. ECCV'10, 2010, pp. 591–604.

[49]. K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in Proceedings of the 2011 International Conference on Computer Vision, ser. ICCV '11, 2011, pp. 1457–1464.

[50]. L. Neumann and J. Matas, "A method for text localization and recognition in real-world images," in Proceedings of the 10th Asian Conference on Computer Vision - Volume Part III, ser. ACCV'10, 2011, pp. 770–783.

[51]. C. V. Jawahar, "Top-down and bottom-up cues for scene text recognition," in Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), ser. CVPR '12, 2012, pp. 2687–2694.

[52]. Y. Bai, L. Guo, L. Jin, and Q. Huang, "A novel feature extraction method using pyramid histogram of orientation gradients for smile recognition," in Proceedings of the 16th IEEE International Conference on Image Processing, ser. ICIP'09, 2009, pp. 3269–3272.

IOSR Journal of Computer Engineering (IOSR-JCE) is UGC approved Journal with Sl. No. 5019, Journal no. 49102.

Suman Mondal" Hog Feature - A Survey." IOSR Journal of Computer Engineering (IOSR-JCE) 20.4 (2018): 01-11.