A faster Object Detection Framework using CENTRIST Visual Descriptor

Sreelakshmi. R[1],
[1]M Tech student,
Gov. College of Engineering,
Cherthala, Alapuzha, Kerala

Sreekumar. K[2]
[2]Asst. Prof, Dep. of Electronics,
Gov. College of Engineering, Cherthala,
Alapuzha, Kerala

Abstract— **This paper describes a faster object detection framework that has the ability of achieving high detection rates. There are two main contributions in this paper which accounts to the faster detection speed. The first is that contour is the main information root for object detection and contours are encoded by comparison with the neighboring pixels. Second contribution is the CENTRIST visual descriptor which is well suited for contour based object detection. CENTRIST visual descriptor encodes sign information and can precisely indicate the global contour. Feature computation is the main speed constraint in existing methods. Most of the time is exhausted for feature construction, feature vector normalization, image preprocessing, etc. Time consuming preprocessing and feature vector normalization are not needed in this framework. When CENTRIST is used with linear classifier, a computational method can be achieved in which feature vector need not be generated. Further hardware acceleration is also possible through this method. This method is applied to detect objects like faces, pedestrians and cars. It has equivalent detection accuracy with the existing methods and has a good improvement in detection speed.**

Keywords— Object detection, CENTRIST Introduction

Our daily lives are filled with thousands of objects ranging from manmade classes like buildings, cars, industries to natural ones like trees, rocks, leaves, mountains, animals and humans. Huge variations can be seen in any given class. For e.g. car can be used to denote many four wheeled vehicles like an Audi, Benz, SUV or Lamborghini. The color and exact type of a car is irrelevant in taking the decision that an object is a car. Similarly humans detect people without taking into account the appearance, color or kind of clothing, pose, illumination, background clutter or partial occlusions. Computers are far behind humans in such conclusion. Thus intention of researchers working in object detection has been to grant computers the ability to see visual analysis.

The main objective of this paper is to detect objects in real time with very high detection accuracy. In certain applications, the efficiency of detector is of supreme importance. Thus faster object detection has high impact in various applications.

Contour which defines outline of an object is necessary in many object detection tasks. CENTRIST feature is suitable for contour based object detection. This framework is a method using CENTRIST and a linear classifier for object detection that does not involve image preprocessing or feature vector normalization. CENTRIST feature vector need not be generated as it is seamlessly embedded into the classifier evaluation, achieving faster detection speed. A cascade classifier is used here. After fast rejections by the linear classifier, quality detection is guaranteed by a non linear classifier. This framework achieves comparable detection quality as state-of the- art detectors, and has a very good advantage in the detection speed.

I. CONTOUR BASED OBJECT DETECTION

Contour is an important cue for object detection. While many recent object detection researches have been based on feature based texture, some other powerful cues have not been sufficiently explored yet, and contour cue is one of them. Contours consist of edge fragments or curves, which present some meaningful concepts on geometry.

.

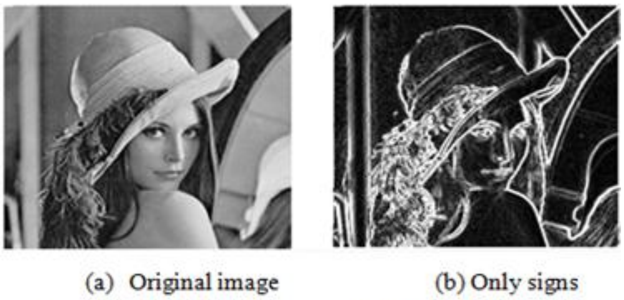

(a) Original image
(b) Only signs

Fig.1 Detection of humans from their contours

Contour features can effectively represent objects that can be clearly defined by shape .Humans recognize a wide range of objects based on their 2D outlines alone. Thus, contour features play an important role in object recognition.

The Sobel image creates an image which emphasizes edges and transitions. Each pixel is replaced with its gradient value normalized to [0 255]. Image gradients are used to detect contours, which are computed by comparing with neighboring pixels. Magnitude of comparison is not important whereas signs of comparisons are key to encode contours. Fig .1b shows such a sign comparison image.

II. CENTRIST VISUAL DESCRIPTOR

Census Transform (CT) emphasizes local image structure to map the intensity values of the pixels within a square window to a bit string, by which we can capture the image structure. CT relies on relative ordering of local intensity values, not on the intensity values themselves and hence is a form of non-parametric local transform[7].

$$\begin{bmatrix}149 & 160 & 230\\ 153 & 154 & 156\\ 156 & 157 & 152\end{bmatrix} \rightarrow \begin{bmatrix}0 & 1 & 1\\ 0 & \times & 1\\ 1 & 1 & 1\end{bmatrix} \longrightarrow (01101111)_2 \longrightarrow 111$$

Fig 2. Evaluating Census Transform value

CENTRIST [8] visual descriptor concisely gives the "signs of neighboring comparison". Census transform compares the pixel values with eight neighboring pixels. If the center pixel intensity value is larger than any of its neighbors, the corresponding position value is replaced with a bit 1.Otherwise its value is replaced with a bit 0. The bit is left shifted in each comparison, thus forming 8 bit string for a window of 3×3 size and a 32 bit string for a census window of 5×5 size. Census Transform value is obtained by finding the binary representation of these bits to a base 10 number.

CT image of an image is obtained by replacing each pixel with its CT value. Histogram of these CT values forms the CENTRIST descriptor. Fig 2. shows the evaluation of CT value of a given pixel.

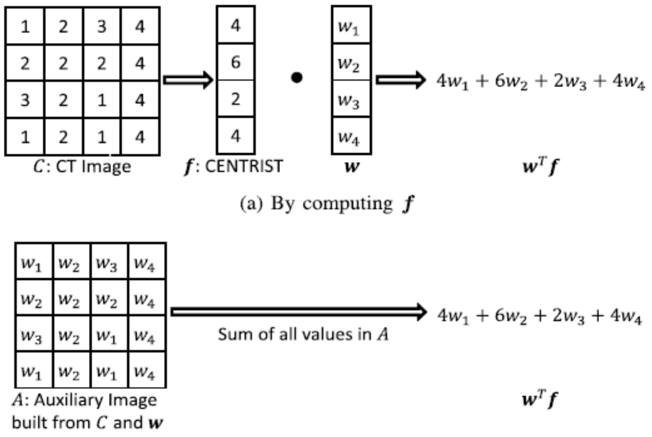
III. CLASSIFICATION

An important part of object detection is to train accurate classifiers.

A. *Linear classifier for fast rejection of non-object*

Linear classifiers are used along with CENTRIST to determine whether an image patch is object of interest or not. Computational cost can be efficiently minimized by this method. If we have already trained a linear classifier w € R^{256}, an image is classified as object only if $\mathbf{w}^{\mathbf{T}}\mathbf{f} > \theta$ where **f** is the centrist feature vector extracted from the patch. CENSUS histogram gives the pixel distribution of CT image's detection window. By directly combining the CT image C and classifier $\mathbf{w}^{\mathbf{T}}\mathbf{f}$, we can build an auxiliary image.

$$A(x, y) = w_{C(x,y)} \text{ ....(1)}$$



(a) By computing f

Fig 3. Illustration of ways to compute $\boldsymbol{w}^T\boldsymbol{f}$. For simplicity, we assume only 4 different CT values. \

From the above, it can be seen that the summation of pixels in the auxiliary image gives the value $\boldsymbol{w}^T\boldsymbol{f}$ which is the term that we need to calculate. Thus with the help of auxiliary image we need not even compute the centrist feature f. Since computation of $\boldsymbol{w}^T\boldsymbol{f}$ is equivalent to the summation of values in the image patch, incorporation of integral image can further reduce the computation time. By the concept of integral image it only requires 4 memory access and 3 summations to find the sums of all pixels in any rectangular region. So applying linear classifier with centrist visual descriptor for object detection requires only few machine instructions per detection window.

B. *Cascade classifier*

A cascade structure [2] is used in this framework. The linear classifier H_{lin} is very fast, but cannot guarantee accurate object detection. H_{lin} threshold is adjusted in such a way that about 99% of objects will be correctly classified as objects. Backgrounds are immediately rejected in this stage. Second classifier used is a more powerful classifier which is non-linear histogram intersection kernel. This is chosen because since the CENTRIST representation is a natural histogram, high accuracy can be achieved by histogram intersection kernel (HIK). Hence those detection windows that are classified as objects by H_{lin} are further considered by HIK SVM classifier H_{hik}. Only those pass the test of both H_{lin} and H_{hik} is considered as an object of interest. CENTRIST vector is not generated in this stage.

The fact that we need not explicitly construct feature vectors for H_{lin} is not the only factor that makes our detection extremely fast. H_{lin} is also a powerful classifier. It filters away about 99% of the candidate detection windows, only less than 1% patches require attentions of the expensive H_{hik}.

C. *Classifier training*

We have a set of h × w positive training image patches P and a set of larger negative images N which do not contain any object of interest. We first randomly form a negative training set N1 by choosing a small set of patches from the images in N. Using P *U* N1, we train a linear SVM classifier H1.

H1 is applied to all patches in the images in N, and false positives are added to N2.H2 is then trained using P and N2. This bootstrap process is repeated to obtain H1, H2,H3 . . .Linear SVM classifier H_{lin} is trained using P and the combined negative set $U_i\ N_i$, which is H_{lin}. The threshold of H_{lin} is adjusted in such a way that it classify a large portion of the objects as positive. A new negative training set N_{final} is generated by applying H_{lin} on N. Then we train an SVM classifier using the libHIK which is H_{hik}.

IV. CENTRIST DETECTION FRAMEWORK

In centrist based object detection framework, multiple rigid parts are used to detect objects. A detection window of fixed size h x w is first divided into n_x x n_y equal sized non-overlapping blocks. Hence size of each block is (h_s,w_s) = (h/n_x,w/n_y). Adjacent 2 x2 blocks are then combined to form a superblock. Thus, size of each superblock is (2h/n_x , 2w/n_y).

Adjacent 2 x2 blocks are then combined to form a superblock. Thus, size of each superblock is (2h/n_x , 2w/n_y). Since there are total of (n_x − 1) × (n_y − 1) superblocks, we create auxiliary images $A^{i,j}$ for $1 \le i \le n_x - 1, 1 \le j \le n_y - 1$ having same size as input image. The (x, y) pixel of $A^{i,j}$ is set to $A_{x,y}^{i,j} = w_{C(x,y).}^{i,j}$.

The decision value of the detection window with top left corner (t, l) has the value

$$\sum_{x=2}^{2h_s-1} \sum_{y=2}^{2w_s-1} A(t+x, l+y) \dots\dots\dots\dots\dots (2)$$

In CENTRIST based object detection brute-force search strategy is used. Every possible detection windows in the input are checked for whether object of interest or not. Top left corners of all possible detection windows form a grid with step size g. For faster object detection g is set to higher value. The input image is scaled down consecutively by a ratio 0.8 for detecting objects bigger than $h \times w$, until the image is smaller than h x w. For every resized version, brute force search is performed.

As mentioned before, two classifiers are used for detection. Those that pass the test of H_{lin} and H_{hik} are considered to be object of interest.

We get multiple overlapping detections for each instance of an object. For eliminating repeated detections, non-maximum suppression is used. For a particular object category in an image, we have a set of detections D. Every detection[6] is defined by a bounding box and a score. We sort the detections in D by score and highest scoring ones are selected. Detections with bounding boxes that are at least 50 percent covered by a bounding box of a previously selected detection are skipped.

TABLE I. CAR DETECTION ACCURACY (AT EER) ON THE UIUC DATASET)

Processing Module	*UIUC-Single*	*UIUC-Multi*
Centrist based object detector	97.5	97.8
Leibe et al.[11]	97.5	95
Mutch and Lowe [12]	99.9	90.6
Lampart et al.[10]	98.5	98.6
Gall et al.[13]	98.5	98.6

Table I shows accuracy of CENTRIST based object detector and existing methods. It can be seen that this method has a very god detection accuracy and clear advantage in detection speed.

V. CONCLUSION

Real-time processing is a must-have property in most of the object detection applications. A fast and efficient method of detecting humans can be achieved emphasizing on the human contour using a cascade classifier and the CENTRIST visual descriptor. One advantage of using this CENTRIST based detection is that it strives a balance between detection accuracy and speed. This framework achieves state-of-the-art accuracy at a reasonable computational cost. CENTRIST is a computationally efficient technique to capture contour cues as compared to other methods. CENTRIST is particularly suitable for human detection, as it precisely encodes the sign information, and can capture large scale structures or contours. Also it detects humans at 20 fps speed on 640 x 480 resolution using only one processing thread and achieves accuracies comparable to the state-of-the-art.

REFERENCES

[1] JianxinWu, Member, IEEE, Nini Liu, Christopher Geyer, and James M. Rehg, Member, IEEE,"C^4:A Real-Time object detection framework",IEEE trans. Image Processing, vol. 22, no. 10, Oct. 2013

[2] P. Viola and M. Jones, "Robust real-time face detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, 2004.

[3] Edgar Osuna,Robert Freund,Federico Girosi,"Training Support Vector Machines: an Application to Face Detection"

[4] J. Wu, W.-C. Tan, and J. M. Rehg, "Efficient and effective visual codebook generation using additive kernels," J. Mach. Learn. Res.,vol. 12, pp. 3097–3118, Nov 2011.

[5] J. Wu, C. Geyer, and J. M. Rehg, "Real-time human detection using contour cues," in Proc. IEEE Int. Conf. Robot. Autom., May 2011,pp. 860–867.

[6] J. Wu and J. M. Rehg, "CENTRIST: A visual descriptor for scene categorization," IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 8,pp. 1489–1501, Aug. 2011.M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989

[7] C. H. Lampert, M. B. Blaschko, and T. Hofmann, "Efficient subwindow search: A branch and bound framework for object localization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2129–2142, Dec. 2009.

[8] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," in Proc. 3rd Eur. Conf. Comput. Vis.,vol.2.1994,pp.151–158.

[9] J. Wu and J. M. Rehg, "CENTRIST: A visual descriptor for scene categorization," IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 8,pp. 1489–1501, Aug. 2011.M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[10] C. H. Lampert, M. B. Blaschko, and T. Hofmann, "Efficient subwindowsearch: A branch and bound framework for object localization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 12, pp. 2129–2142,Dec. 2009.

[11] J. Wu, "Efficient HIK SVM learning for image classification," *IEEETrans. Image Process.*, vol. 21, no. 10, pp. 4442–4453, Oct. 2012.

[12] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detectionwith interleaved categorization and segmentation," *Int. J. Comput. Vis.*,vol. 77, nos. 1–3, pp. 259–289, 2008

[13] J. Mutch and D. G. Lowe, "Multiclass object recognition with sparse,localized features," in Proc. IEEE Comput. Soc. Conf. Comput. Vis.*Pattern Recognit.*, Jun. 2006, pp. 11–18.

[14] J. Gall, A. Yao, N. Razavi, L. V. Gool, and V. Lempitsky, "Hough forests for object detection, tracking, and action recognition," *IEEE* Trans. Pattern Anal. Mach. Intell., vol. 33, no. 11, pp. 2188–2202,Nov. 2011.