International Journal of New Technology and Research (IJNTR)
ISSN:2454-4116, Volume-2, Issue-11, November 2016 Pages 22-25

Hidden and Unknown Object Detection in Video

R. A. Simonyan

***Abstract*— Object detection is applied to find such actual objects as faces, bicycles and buildings in images and videos. The algorithms executed in object detection normally use extracted features and learning algorithms to distinguish object category. It is often implemented in such processes as image retrieval, security, surveillance and automated vehicle parking system.Objects can be detected through a range of models, including Feature-based object detection, Viola-Jones object detection, SVM classification with histograms of oriented gradients (HOG) features, Image segmentation and blob analysis.For detection of hidden objects in the video the Object-class detection method is used, in which case the object or objects are defined in the video in advance [1][2].The proposed method is based on bitwise XOR comparison [3]. The method (system) detects moving as well as static hidden objects.The developed method detects objects with great accuracy; it detects also those hidden objects which have great color resemblance to the background images, which are undetectable for a human eye. There is no need to define or describe the searched object before the detection. Thus, the algorithm does not limit the search of the object depending on its type. The algorithm is developed to detect objects of any type and size. It is calculated so to work in case of weather change as well as at any time during a day irrespective of the brightness of the sun (which leads to the increase or the decrease of the intensity of the brightness of an image); in this way the method works dynamically. A system has been developed to execute the method.**

***Index Terms*— Image Processing, Hidden and Unknown Object Detection, Video Surveillance.**

I. Introduction

For monitoring purposes the camera is installed in such a place, where the penetration of the object is strictly banned and dangerous, for instance in railway junction.

Right after installing and turning on the camera the algorithm starts functioning directly parallel with recording. The algorithm saves the initial (first) image. The image is saved only once at the beginning of the execution of the algorithm. The camera records an image of n x m size (Fig. 1).

Fig. 1: The initial image.

Fig. 2: The appearance of an object in a banned place.

During the recording a new image is taken every second and is compared with the initial image. Suppose 30 seconds after the shooting a new object has appeared in the railway junction (Fig. 2).

As the algorithm works directly in parallel with the recording it will get the new image where the new object is and will execute bitwise XOR comparison of the initial image (Fig. 1). Consequently, we will get the result in Fig. 3, which will show the presence of the object.

As we know a bitwise XOR perform the comparison of two pixels (bits), being 1 if the two bits are different, and 0 if they are the same. In Fig. 3 as a result of XOR bitwise comparison the returned 1 is presented in white color, and 0- in black color.

Let's consider the case, when an object appears in the image not at the beginning of the recording but after a long time (e.g. 4 hours later), when the color values of pixels are changed because of the change of brightness of the sun. In this case as a result of XOR bitwise comparison 0 is returned for all pixels and the result will be a totally black image, the object will not be detected. Thus, in any weather change, the system, not detecting the object, will not serve its purpose.

That is why the following method has been developed which will work irrespective of light or weather change based on XOR bitwise comparison.

Fig. 3: The result of XOR bitwise comparison.

II. THE DESCRIPTION OF THE METHOD

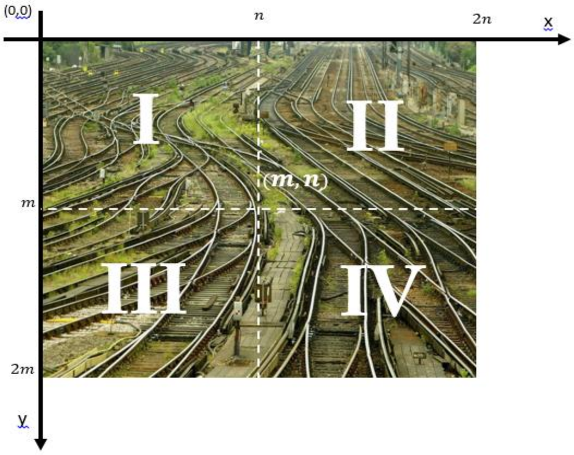

Fig. 4: The initial image divided in 4 parts.

To detect the object with great accuracy the image is divided into 4 parts, as it is presented in Fig. 4.

The image consists of pixels, which can be presented in the form of a matrix [3-5]. Initial image will be defined with a Latin letter A. Hence, A is the digital matrix of the initial image, each element of which presents one pixel, three-dimensional vector (1).

$$a_{i,j} = (r_{i,j}; g_{i,j}; b_{i,j}), \quad i = \overline{1,n}, \quad j = \overline{1,m} \qquad (1)$$

r,g,b ϵ (0, 255) and r,g,b represent the intensity of red, green and blue accordingly.

As the image has been divided into 4 parts, for each part there will be separate matrixes accordingly $A^{(1)}, A^{(2)}, A^{(3)}, A^{(4)}$:

For each part of the Fig. 4 different values are calculated: the values of red, green and blue colors (e.g. $r^{(1)}_{a\,mid}$, $g^{(1)}_{a\,mid}$, $b^{(1)}_{a\,mid}$). Each value is decided according to the color of all the pixels of that part (e.g. according to red) with the arithmetical average of numeric values (2-5). To put in other words, the value shows the average intensity of the color in a part.

The values of the intensity of all three colors are counted for the 1st,2nd,3rd and 4thparts(2-5).

$$a^{(1)}_{mid} = \frac{1}{m \times n} \sum_{i=1}^{m} \sum_{j=1}^{n} a_{i,j} \qquad (2)$$

$$a^{(2)}_{mid} = \frac{1}{m \times n} \sum_{i=1}^{m} \sum_{j=n}^{2n} a_{i,j} \qquad (3)$$

$$a^{(3)}_{mid} = \frac{1}{m \times n} \sum_{i=m}^{2m} \sum_{j=1}^{n} a_{i,j} \qquad (4)$$

$$a^{(4)}_{mid} = \frac{1}{m \times n} \sum_{i=m}^{2m} \sum_{j=n}^{2n} a_{i,j} \qquad (5)$$

Depending on the place and probable weather changes, σ is determined for the images as a total threshold ratio while installing the camera for the first time.

During the video recording the current image is taken per second. The current image will be marked with C letter. In image C the color values of all pixels can be changed because of the change of the brightness of the sun, for that reason we try to bring the color intensity as closer to the color intensity of the image A as possible, which will give the opportunity to execute XOR bitwise comparison of image A and C and to decide whether a new object has appeared in the image or not (the method is described below).

The above mentioned algorithm is executed for the image C and $c^{(1)}_{mid}$, $c^{(2)}_{mid}$, $c^{(3)}_{mid}$, $c^{(4)}_{mid}$ values are calculated (6). Then the differences of color values of the initial and the current images of each part are calculated (7).

$$c^{(k)}_{mid} = (r^{(k)}_{c\,mid}, g^{(k)}_{c\,mid}, b^{(k)}_{c\,mid}), \quad k = \overline{1,4} \qquad (6)$$

$$d^{(k)} = c^{(k)}_{mid} - a^{(k)}_{mid} \qquad (7)$$

$d^{(k)}$-is a three-dimensional vector (8):

$$d^{(k)} = (r^{(k)}_{diff}, g^{(k)}_{diff}, b^{(k)}_{diff}) \qquad (8)$$

$r^{(k)}_{diff}$ – is the numeric difference of the intensity of red color between image C and A. Thus $r^{(k)}_{diff}$ vectors show to what extent the color has changed in the current image in comparison with the initial one[6-7].

3rd image is taken, which is white (marked with D letter). The new object will be presented in black color in the 3rd image.

By subtracting the value of $d^{(k)}$ coefficient of difference from the value of each pixel of the current image C, $(c^{(k)}_{i,j} - d^{(k)})$, we will have the image C' , which is very close to the image A by its color intensity. In the image the value of each pixel is very close to the respective pixel in the image A.

In image C' the presence of an object will lead to a greater change of pixels in that particular part in comparison with the image A, much greater than the other pixels of image C', even in case when the image has a great color similarity to the background image C and is unnoticeable for the eyes, that is to say the object is hidden. To detect the changes of pixels in image C' the inequality in formula 9 is calculated (9).

$$|(c^{(k)}_{i,j} - d^{(k)}) - a^{(k)}_{i,j}| > \sigma \qquad (9)$$

If the inequality meets the condition, that is to say if the left part of the inequality is bigger than the index of σ, then i,j pixels of the image C is much different from i,j pixels of the image A, which means that this is the pixel of the detected object. The relevant pixel of i,j in the image D gets the value 0,0,0 ($d^{(k)}_{i,j}$ = (0, 0, 0)), i.e. it gets black. The collection of black pixels will show the presence of the object, the result will be presented in the image D (Fig. 8).Thus, if there is an object in an image the appropriate pixels will be blacken according to the algorithm, the object is detected. In Fig.5 the block scheme of object detection is presented.

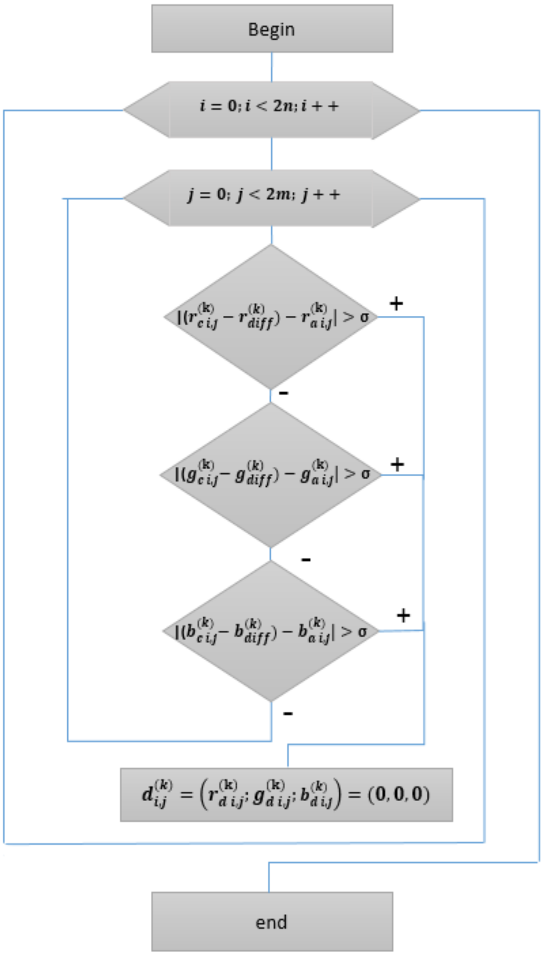

Fig. 5: The block scheme of object detection.

Fig. 6: There are three objects in the image, however only two objects can be detected by human eye (cars), the third one is hidden (which is a cat).

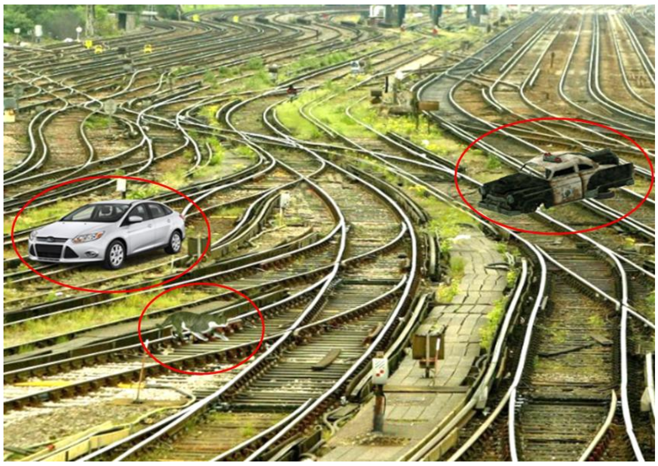

Fig. 7: The image C', where the presence of three objects are presented in with red circles.

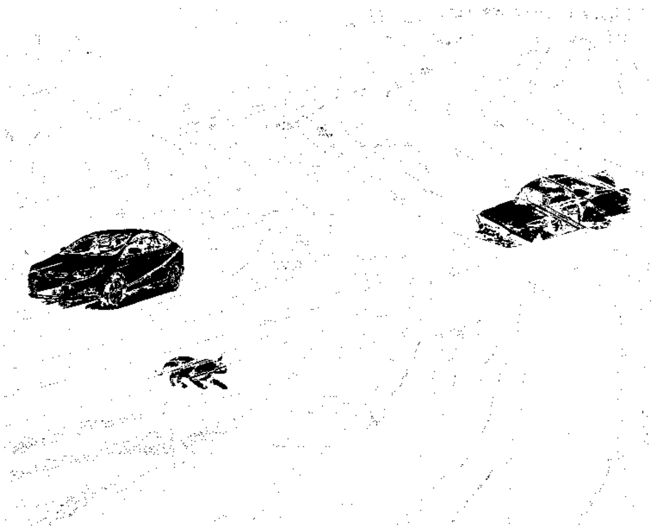

Fig. 8: The image D, where the detected objects are presented in black color.

III. CONCLUSION

Thus, a method has been developed, which detects hidden and unknown (previously undefined) objects in video (which are unnoticeable for human eye and have huge color similarity to the background image). Moreover, the object can be moving as well as static.

The developed method detects objects with great accuracy considering the possible changes of light and weather during recording. The algorithm does not limit the searched object according to its type.

REFERENCES

[1] R. G. Cinbis, S. Sclaroff, "Contextual Object Detection using Set-based Classification" 2012, pp. 15–64.

[2] F. Schwenker, A. Sachs, G. Palm, H.A. Kestler,"Orientation histograms for face recognition", In Schwenker F. and Marinai S., editors, Artificial neural Networks in Pattern Recognition, volume 4087 of LNAI, pages 253–259, Berlin Heidelberg New York, 2006.

[3] D. Phillips, "Image Processing in C" 1st ed. Lawrence, Kansas, 1994

[4] M. Nixon, "Feature Extraction and Image Processing for Computer Vision, Third Edition" 3rd ed. UK London, 2008, pp. 137-395

[5] S.Lazebnik, C.Schmid, J. Ponce,"Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories", In

Proceedings of 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Vol. 2, pp. 2169– 2178. IEEE Computer Society Press, June 2006.

[6] R.Poli.,"Genetic programming for feature detection and image segmentation", In T. C. Fogarty, editor, Evolutionary Computing, number 1143 in Lecture Notes in Computer Science, pp. 110–125. Springer-Verlag, University of Sussex, UK, 1-2 April 1996.

[7] M. Zhang, V. Ciesielski. "Using back propagation algorithm and genetic algorithm to train and refine neural networks for object detection". Florence, Italy, August 1999. pp 626–635, (LNCS Vol. 1677).