TEST Engineering & Management

November-December 2019
ISSN: 0193-4120 Page No. 01 – 08

Implementation of Multi-object Recognition Algorithm Using Enhanced R-CNN

HyochangAhn[1], June-Hwan Lee[2], Han-Jin Cho[*3]

[1,2,3]Professor, Department of Energy IT,Far East University, Eumseong-gun, Chungcheongbuk-do, 27601, Republic of Korea

*youcu92@kdu.ac.kr [1], rainbow@kdu.ac.kr[2], hanjincho@kdu.ac.kr[*3]*

Article Info
Volume 81
Page Number: 01 - 08
Publication Issue:
November-December 2019

Abstract

Background/Objectives: Multi-object recognition is emerging as a technology that can be applied to various real worlds such as image security, gesture recognition, robot vision, and human robot interaction, and it is difficult to recognize public objects in a complex background.

Methods/Statistical analysis: Most multi-object tracking methods suffer from performance degradation due to the number of objects changing each frame, and this phenomenon is more pronounced in complex backgrounds. Therefore, in this paper, we propose an improved R-CNN-based multi-object recognition method that can detect multiple objects quickly while being robust to geometric distortions, lighting changes, and noise of environmental elements and objects in the image.

Findings: Experiments were compared the detection rate and detection rate of the CNN method, R-CNN method and the proposed method.

The proposed method takes more time to recognize multiple objects, but the object recognition rate shows a high result.

Improvements/Applications: Through the proposed deep learning based multi-object recognition, it can contribute to the research that monitors and tracks several objects at the same time in the surveillance system.

Keywords: Multi-Object, Object Recognition, Machine Learning, R-CNN, Surveillance.

Article History
Article Received: 3 January 2019
Revised: 25 March 2019
Accepted: 28 July 2019
Publication: 22 November 2019

I. Introduction

Computer vision is a technology that gives a machine visual capability in the field of artificial intelligence[1,2]. This technology is actively applied in various industries such as copyright protection of video contents, robotics, medicine, military, security and surveillance, automotive industry and quality inspection[3,4]. Computer vision is generally aimed at programming a computer to understand scenes or features in a image, and based on this, object detection, segmentation, recognition, and object tracking in continuous images are performed. Unlike humans who are basically given visual capabilities, artificially imparting visual capabilities to machines is a long-standing challenge and major issue in the field of artificial intelligence. This is because there are many environmental factors that are

November-December 2019
ISSN: 0193-4120 Page No. 01 – 08

difficult to recognize in the machine, such as uncontrolled and variable lighting, shadows, complex landscapes, and objects that contain other objects.

Multi-object tracking is emerging as a technology that can be applied to various real worlds such as image security, gesture recognition, robot vision, and human robot interaction[5,6]. The challenging problem of multi-object tracking is the occurrence of drift in tracking trajectories due to changes in appearance caused by noise, light changes, poses, complex backgrounds, interactions, obstructions by objects, camera movements, and so on. Most multi-object tracking methods suffer from performance degradation due to the number of objects changing each frame, and this phenomenon is more pronounced in complex backgrounds.

Object detection includes detection using feature points, background detection using a background modeling method such as Gaussian mixture model (GMM), DPM, and a method using a neural network such as YOLO[2,7]. Object detection has significantly improved its performance in the era of deep learning. In places such as Kaggle and ILSVRC, the object detection method using deep learning occupies the top rank. Unfortunately, there is a lack of research using deep learning for object tracking. In the case of MOT, which is a competition for tracking multiple objects, the method using the hand-crafted feature is still the majority, and the performance is also advanced. In this paper, we propose an improved R-CNN-based multi-object recognition method that can detect multiple objects quickly while being robust to geometrical distortions, illumination changes, and noise of environmental elements and objects in an image.

The rest of this paper is organized as follows. Chapter 2 introduces various methods for detecting objects, and Chapter 3 introduces the proposed method based on R-CNN for detecting multiple objects. Section 4 shows the performance evaluation and the results of the proposed method. Section 5 discusses the proposed method and discusses future research tasks.

II. Related Works

KLT(Kandade-Lucas-Tomasi)

In order to track multiple objects, proper object detection is required for image information collected at the previous stage[8,9]. A widely used feature detection method is an object detection method that finds and uses representative texture information in image information. The Harris edge detector uses the $M \in R \wedge (2 \times 2)$ matrix of first-order derivatives in the horizontal and vertical directions to represent the intensity of edges, boundaries, and flat areas. Harris edge detector shows strong feature detection performance based on the interrelationship of local signals, but it has a disadvantage that various post-processing is required to use it as feature points because the detected edges are output as area instead of independent points. Later, Lucas and Kanade proposed the feature tracking theory using the gradient descent method. It is characterized by repeatedly arranging image patches in successive image frames. Tomasi and Shi also proposed an estimation operator based on the Affine transformation, which is a transformation method that defines the transformation relationship between two images to be transformed.

The Kanade-Lucas-Tomasi (KLT) detector has been used continuously for feature point detection and tracking in continuous image

information. A feature matrix is extracted by calculating a local matrix of random points (x, y) for each image in the image frame. According to the calculated values, feature points present in the image are classified into various types such as corner points, one-way boundary lines, and noise. Such texture information-based object detection techniques show good overall performance but have common problems sensitive to image noise. Recently, as the necessity of the intelligent video system in the real-time environment that changes greatly, the object detection method through the adaptive background generation has been actively studied.

CNN (Convolutional Neutral Network)

Many algorithms, such as HOG and SIFT, were used as algorithms for detecting and classifying objects in images, but there were limitations in performance[9,10,11]. Most of the researches on the method of bringing the performance improvement through merging with other existing algorithms have shown a slight improvement in the performance. In 2012, CNN's alexNetappeared at the International Image Recognition Technology Competition called ImageNet, which showed about a 10% performance improvement over the previous best performing algorithms [12,13]. Currently, studies related to image recognition are mainly based on CNN. In 1989, CNN was first applied to a study that recognized handwritten postal codes, but it was used in a limited range due to excessive learning time and problems such as overfitting. However, due to the development of hardware, the computation time is reduced exponentially through parallel operation using GPU, and improved algorithms such as dropout and ReLu have been studied. In addition, with the advent of big data, it is easier to secure a large amount of data for learning and verification than before, thus creating an environment in which CNN can be used. Unlike conventional techniques, CNN has a structure that extracts and optimizes features for input images and repeats the classification process several times. The Convolution Layer is responsible for extracting meaningful features from the image. A kind of filtering is performed through the convolution operation on a certain mask. Perform convolution operation by applying filter to get desired feature, and finally create feature map. After that, the image feature values in the feature map generated by the convolution layer are extracted from the pooling layer. Feature extraction methods include Max Pooling, which selects the maximum value, and Average Polling, which selects the average value. By selecting and extracting the required features, the number of dimensions is reduced to reduce the amount of computation. Finally, the objects are classified and predicted using the features extracted as a result of the repetition of the convolution layer and the pooling layer as the fully connected layer. The application of CNN to image classification has made a big leap in image classification, but only the presence of objects in the input image can be known. However, since the location of the object is unknown, CNN alone is not suitable for solving the problem of detecting an object.

III. Proposed Method

R-CNN uses the technique used in the existing CNN, but because it learns based on image region information, it can detect the position of an object and goes through the process as shown in Figure 1[14,15].

November-December 2019
ISSN: 0193-4120 Page No. 01 – 08

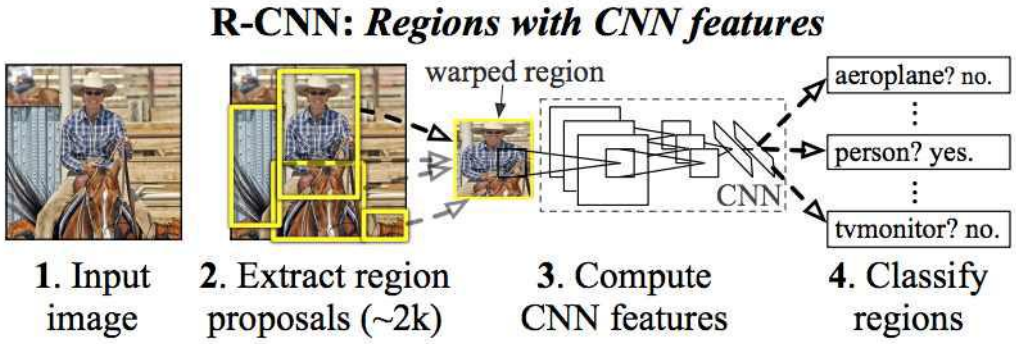


Figure 1. R-CNN basic structure

Once the image has been entered, the second step is to select the region proposal where the object may exist using the Selective Search method. Selective Search method performs Sub-Segmentation in the bottom-up manner[16,17]. After that, it divides the image into smaller regions, groups similar regions together, and repeats the process of merging them into larger regions to create an integrated candidate region. After selecting 2,000 candidate regions through Selective Search, the third step is to extract feature vectors through CNN. CNN uses alexNet to go through a series of Warp or Crop inputs to receive a fixed size image of 224 x 224 pixels. The final step is to classify the area using Linear SVM. R-CNN improves the detection performance of objects compared to previous methods by using region-based CNN feature, but there are some disadvantages. First, three steps of learning about Fine-Tuning, SVM Fitting, and Bounding Box Regressor for 2,000 candidate areas are required and it takes excessive time. Secondly, the extracted feature should be saved to disk. Finally, image warping or loss occurs due to Warp or Crop, which adjusts the size of the input image to 224 × 224, which can reduce recognition performance.

The RPN is a network for making region proposals. If the existing Fast-RCNN's selective search is done outside the Neural Net, the Faster-RCNN can improve the speed by allowing the RPN to be used internally. RPN computes an image using CNN and computes the result as a composite product of NxN. The result is compared with the bounding box data holding the Loss Function. After that, RPN (Region Proposal Network) is learned by doing back propagation with Loss Function result. At this time, in the process of finding a box, various types of boxes are created in consideration of various shapes of objects, which are called anchor boxes. RPN runs several anchor boxes around the output feature map.In this paper, we propose a method of recognizing multiple objects using the enhanced R-CNN method by combining the sliding window method to solve the problems of the existing R-CNN.

A sliding window method is used to extract candidate regions in which an object can exist in an entire image. After scaling the image to be detected, the candidate region is extracted from the left to the right and the top to the bottom of the image across the sliding window which is a different scale or fixed scale. The sliding window evaluates a quality function (f) (quality function / classifier score) as shown in Equation 1 on the candidate regions R extracted from the image(I) using Brute Force, and takes a maximum value of the object. Recommended location.

$$R_{obj} = argmax_{R \in I} f(R) \qquad (1)$$

November-December 2019
ISSN: 0193-4120 Page No. 01 – 08

However, when searching for objects of various sizes, if the size or aspect ratio of the candidate area is not constant, the image is repeatedly searched. For this reason, there are inefficient limitations in terms of computation time. Alternatively, increasing the moving step size of the sliding window may reduce the candidate areas slightly. However, still very many candidate regions are extracted. Set of m candidate regions in one image $M: \{f_1, \dots \dots, f_m\}$ is extracted and it is evaluated whether or not there is an object in the candidate region with the highest probability $Q[f]$ among the candidate region $\widehat{f_i}$. However, because the size and type of various objects are included in one image, not only one candidate region $\widetilde{f_i}$ can maximize $Q[f]$ but also other candidate regions. Therefore, many candidate regions must be extracted and calculated to calculate all $Q[f_i]$. By choosing a sufficiently large arbitrary subnet $\widetilde{M} \in M$, it is assumed that the maximum value taken at $\widetilde{M}$ and the maximum value taken at M are nearly similar. Calculate the number of sign areas $\widetilde{m}$ required for subnet $\widetilde{M}$ as shown in the following equation 2.

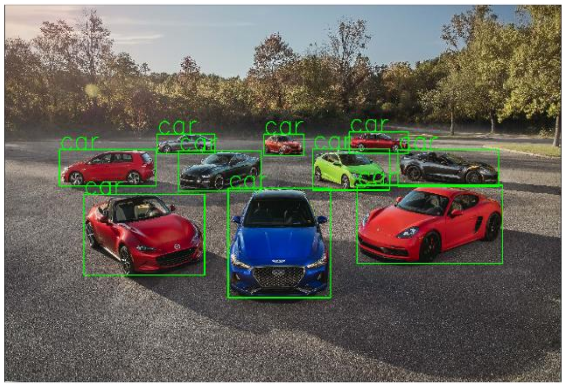
$$\widetilde{m} = \left|\widetilde{M}\right|\left(\widetilde{M} \in M\right) \quad (2)$$

Where μ denotes the desired confidence value and n denotes the number of elements in M with $Q[f_i]$ less than the maximum value of $Q[f_i]$ of the elements in $\widetilde{M}$. As shown in Equation 3, it can be confirmed that the larger the number of n is, the smaller the number of $\widetilde{m}$ is. Therefore, when the random subnet is applied to the sliding window, even if the number of very small candidate areas is randomly selected, almost all objects of the image can be covered.

$$\widetilde{m} = \frac{log * 1-\mu}{ln\left(\frac{n}{m}\right)} \quad (3)$$

IV. Results and Discussion

The environment used for the performance evaluation of the method proposed in this paper was constructed, learned and evaluated using Python on Windows 10. In addition, we implemented a multi-object recognition algorithm using the improved R-CNN proposed by using OpenCV library. Figure 2 shows the proposed method running in the experimental environment.



November-December 2019
ISSN: 0193-4120 Page No. 01 – 08

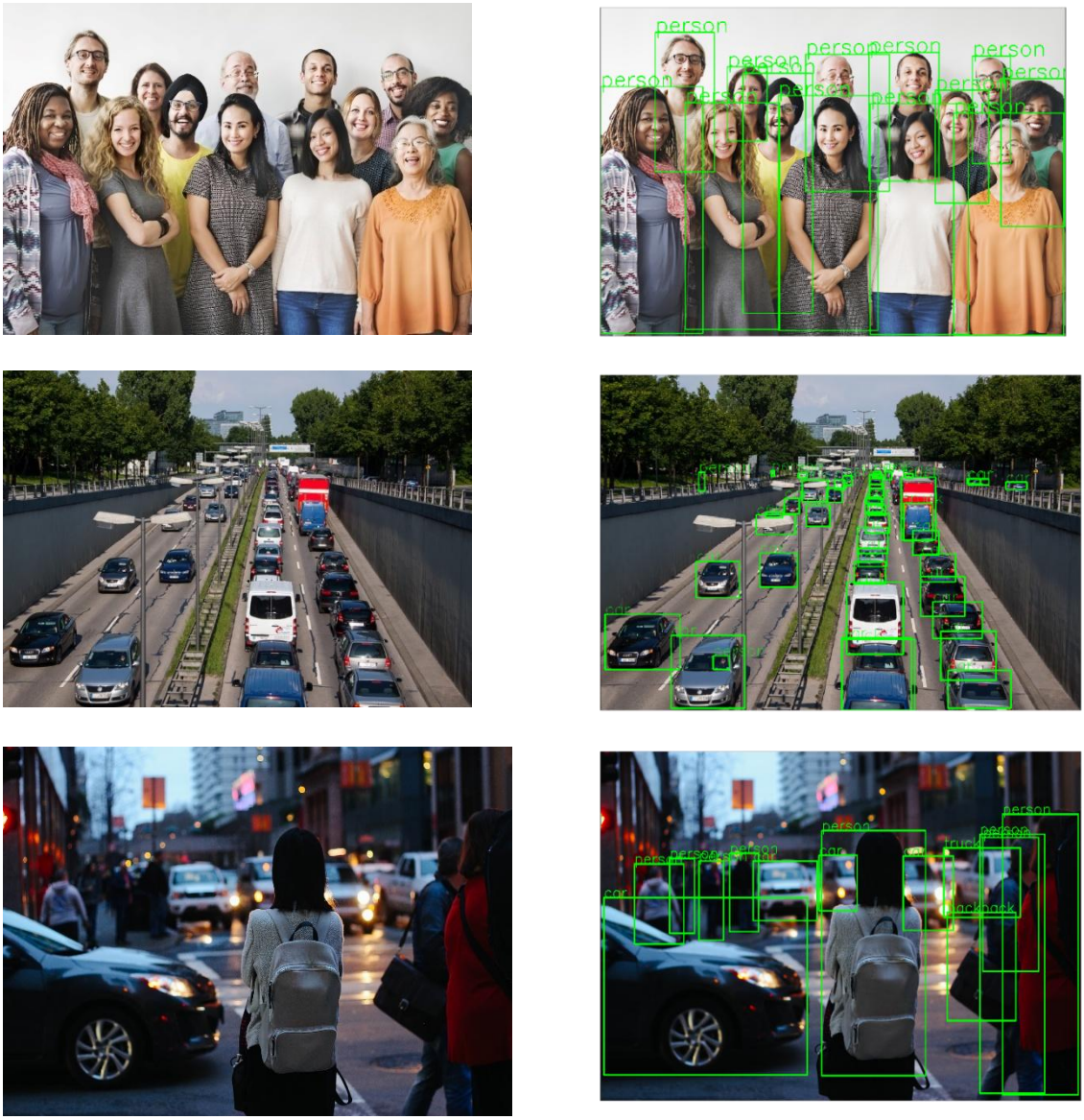


Figure 2. Results Images of the Proposed Method

To measure the performance improvement of the proposed R-CNN method, we compared three algorithms: recognition by CNN, recognition by R-CNN, and recognition by proposed algorithm.

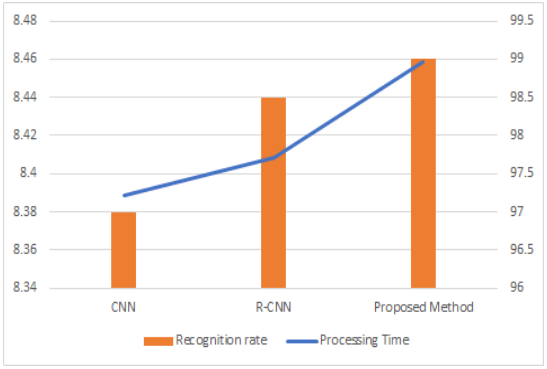


Figure 3. Results Processing Time and Recognition Rate

Experiments were compared the detection rate and detection rate of the CNN method, R-CNN

November-December 2019
ISSN: 0193-4120 Page No. 01 – 08

method and the proposed method. Figure 3 shows that the proposed method takes more time to recognize multiple objects, but the object recognition rate is higher.

V. CONCLUSION

Recently, video surveillance and security monitoring system technology has been rapidly developed to monitor various situations and respond quickly, and related researches are being actively conducted. We propose an improved R-CNN based method for multi-object recognition. Experimental results show that the proposed method takes more time to recognize multiple objects, but the object recognition rate is higher. Through the proposed deep learning based multi-object recognition, it can contribute to the research that monitors and tracks several objects at the same time in the surveillance system.

REFERENCES

[1]. Valera, Maria, and Sergio A. Velastin. "Intelligent distributed surveillance systems: a review." IEE Proceedings-Vision, Image and Signal Processing 152.2 (2005): 192-204.

[2] Ahn, Hyochang, and Yong-Hwan Lee. "Performance analysis of object recognition and tracking for the use of surveillance system." Journal of Ambient Intelligence and Humanized Computing 7.5 (2016): 673-679.

[3] Collins, Robert T., et al. "A system for video surveillance and monitoring." VSAM final report (2000): 1-68.

[4] Grabner, Helmut, et al. "Tracking the invisible: Learning where the object might be." 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE, 2010: 1285-1292

[5] Babenko, Boris, Ming-Hsuan Yang, and Serge Belongie. "Robust object tracking with online multiple instance learning." IEEE transactions on pattern analysis and machine intelligence 33.8 (2010): 1619-1632.

[6] Babenko, Boris, Ming-Hsuan Yang, and Serge Belongie. "Visual tracking with online multiple instance learning." 2009 IEEE Conference on computer vision and Pattern Recognition. IEEE, 2009: 983-990

[7] Redmon, Joseph, et al. "You only look once: Unified, real-time object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.

[8] Comaniciu, Dorin, Visvanathan Ramesh, and Peter Meer. "Kernel-based object tracking." IEEE Transactions on Pattern Analysis & Machine Intelligence 5 (2003): 564-575.

[9] Dalal, Navneet, and Bill Triggs. "Histograms of oriented gradients for human detection." 2005. CVPR 2005. IEEE Computer Society Conference on. Vol.1. IEEE:886-893.

[10] Lowe, David G. "Distinctive image features from scale-invariant keypoints." International journal of computer vision 60.2 (2004): 91-110.

[11] Zhou, Huiyu, Yuan Yuan, and Chunmei Shi. "Object tracking using SIFT features and mean shift." Computer vision and image understanding 113.3 (2009): 345-352.

[12] Chen, Yan, et al. "CNNTracker: Online discriminative object tracking via deep convolutional neural network." Applied Soft Computing 38 (2016): 1088-1098.

[13] Xu, Zhongwen, Yi Yang, and Alex G. Hauptmann. "A discriminative CNN video representation for event detection."

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015:1798-1807

[14] Nguyen, Anh, Jason Yosinski, and Jeff Clune. "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 427-436.

[15] He, Kaiming, et al. "Spatial pyramid pooling in deep convolutional networks for visual recognition." IEEE transactions on pattern analysis and machine intelligence 37.9 (2015): 1904-1916.

[16] Girshick, Ross. "Fast r-cnn." Proceedings of the IEEE international conference on computer vision. 2015:1440-1448.

[17] Ren, Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." Advances in neural information processing systems. 2015: 91-99.