Human Detection and Tracking on Surveillance Video Footage Using Convolutional Neural Networks

Dima Maharika Dinama*, Qurrota A'yun†, Achmad Dahlan Syahroni‡,
Indra Adji Sulistijono§, Anhar Risnumawan¶
Mechatronics Engineering Division*‡§¶, Multimedia Broadcasting Technology Division†
Politeknik Elektronika Negeri Surabaya
Surabaya, Indonesia
{*dmdinama,‡achmaddahlansyahroni}@me.student.pens.ac.id,†qurrota@mb.student.pens.ac.id,
{§indra, ¶anhar}@pens.ac.id

***Abstract*—Safety is one of basic human needs so we need a security system that able to prevent crime happens. Commonly, we use surveillance video to watch environment and human behaviour in a location. However, the surveillance video can only used to record images or videos with no additional information. Therefore we need more advanced camera to get another additional information such as human position and movement. This research were able to extract those information from surveillance video footage by using human detection and tracking algorithm. The human detection framework is based on Deep Learning Convolutional Neural Networks which is a very popular branch of artificial intelligence. For tracking algorithms, channel and spatial correlation filter is used to track detected human. This system will generate and export tracked movement on footage as an additional information. This tracked movement can be analysed furthermore for another research on surveillance video problems.**

***Index Terms*—Surveillance Video, Human Detection, Human Tracking, Deep Learning, Convolutional Neural Networks**

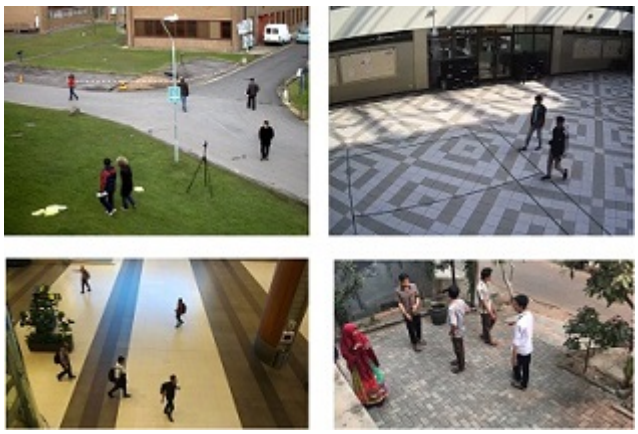

Fig. 1. Snapshot of surveillance video footage that contains human. From these image we can see that this footage contains no additional information.

I. Introduction

Safety is one of basic human need that have to be fulfilled. Public safety is one of major task problem faced in the world. As crime rate rising, needs of safety on public place is also becoming a big demand. Most common used solution for this problem is surveillance video. Surveillance video allows us to record images or videos on certain location. With this application of technology, we feel that we being watched and then give us sense of security.

However, surveillance video that are widely used today only able to capture image or record video. There is no additional information except that pixel combination provided by surveillance video device as in Fig. 1. Surveillance video device only send images or video to monitor in security room. This condition led to need of human resource to monitor the image or video footage recorded by surveillance video device. While the device is recording non-stop it also means that surveillance video operator needs to watch the monitor continuously. By watching the monitor continuously, the operator can suffer fatigue that can reduced effectiveness of surveillance video. Therefore, there is a high demand to automatically process footage from surveillance video device and extract additional information that can be useful for security officer. One of the information that we need from surveillance video footage is existence of human. As human is the target of surveillance we need to monitor human activity within the footage.

In this paper, to solve the problem on human detection we use artificial intelligence based on Deep Convolutional Neural Networks (CNN) to detect and localize human position inside surveillance video footage. This framework has been trained from arranged dataset consist of thousand of images to increase performance on detecting human in various condition. It is trained using deep residual neural network to detect human and added with regional proportion layer to localize the human condition. After we acquire the human location inside the footage then we're using tracking algorithm to track the human and record its movement. The experimental results of this method shows excellent results both on detecting and tracking human on surveillance video footage.

This paper is organized as follows. Related work is described in Section II. Section III explains the methodology and

978-1-7281-4449-8/19/$31.00 ©2019 IEEE

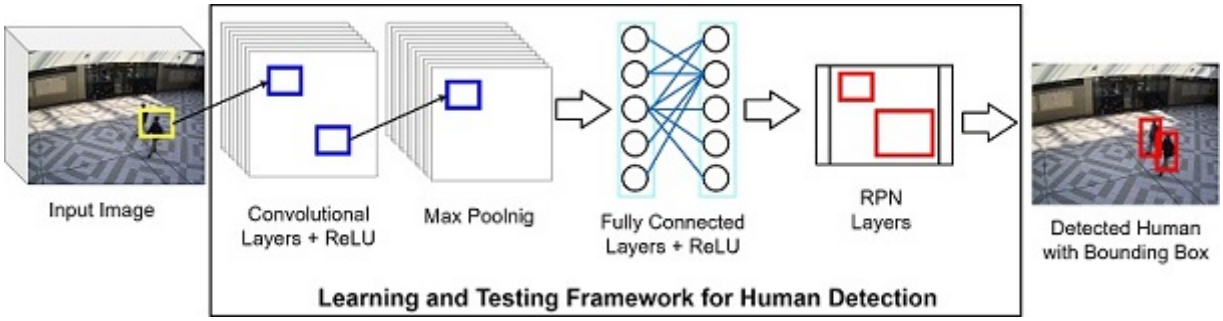

Fig. 2. Overview of human detection framework used in this research. The framework consist of convolutional layer, ReLU, max pooling, and fully connected layer. This framework given output of human location with bounding boxes.

it consists of III-A talks about Surveillance Video Footage, III-B describes Human Detection Framework, and III-C explains the Tracking Algorithm. IV and V show experiments results and conclusion of our works.

II. Related Work

Research on computer vision especially in surveillance video is growing fast in recent years. Human detection, human action recognition, motion tracking, scene modelling, and behaviour understanding have been growing popularity in computer vision and machine learning researcher and communities. This led to discussion about how to maximize performance on advanced surveillance video.

Deep learning method [1] have been successfully improved various visual detection and recognition tasks. Example of this application used for image classification [2], [3], image segmentation [4], and object detection [5], [6]. Deeper network has a main advantage of the ability to learn effective feature representation automatically, which make appealing for practitioners. All the network parameters are solely learned from the training data.

As for surveillance video, many researcher were trying to extract information from video footage. For surveillance video topics, its objective is to detect, recognize, or learn interesting events. This leads to research on action recognition, suspicious event detection [13], irregular behaviour [14], unusual activity [15], anomaly [16], and abnormal behaviour [17].

As before we able to do that works. We have to detect and locating human in our video footage. In this work, by utilizing deep technique on human detection combined with tracking algorithm we show it is possible to detect an track human movement from video footage.

III. Human Detection and Tracking Algorithm

As main subject of this paper is human detection and tracking. This section will explain both about detection frameworks and tracking algorithm used in this research.

A. Surveillance Video Footage

Footage used in this research consist of two dataset. First is PETS2009 [7] dataset especially on S2L1 scheme which consist of people walking inside footage. We also provide our own dataset which we record manually and saved as PENS Surveillance Video Research dataset or PSVR2019.

B. Human Detection Framework

Overview of proposed detection system is shown in Fig 2. Video footage from surveillance video is extracted to images frame-by-frame to be processed as an input layer using this method. This network given result of bounding boxes which contains information indicating human position within footage. Main core of this human detection framework is Convolutional Neural Networks (CNN). CNN as a deep learning method has shown significantly great performance for detection and classification. Therefore, we use CNN as base of our neural networks structure.

CNN is basically several layer staged together just like another neural network structure. A layer in CNN commonly consists of convolutional, max pooling, and fully connected layers which have different roles for each layer. During training, forward and backward stages are performed to increase accuracy. For an input section, forward stage is performed on each layer. During training, once the forward stage is performed then the output will be compared with the ground truth and calculate loss to perform backward stage by updating the weight and bias parameters. For achieving desired accuracy, many iteration of this process will be performed. Training data simultaneously updateng all layers parameters on this network.

A convolutional layer consist of linear filters which is followed by a non-linear activation function. This work used an activation layer such as the Rectified Linear Unit (ReLU). In this convolutional layer, a CNN utilizes various kernels to convolve the whole image as well as the intermediate feature maps, generating various feature maps. The feature map contains A width, B height, C channels to indicate size of feature map. To reduce dimensions of feature maps used, we can use pooling layer then followed by convolutional layer. Pooling layers are invariant to translation, it takes the neighbouring pixels of feature maps. Max pooling is simply taking the maximum value from predetermined window.

Fully-connected layers performs similar as feed forward neural network. It convert previous multi-dimensional feature maps into a pre-defined length. These layers acts as classi-

fication layer and could be used as feature vector for next processing.

- Neural Networks Structure
  For deep learning convolutional neural networks there are many kind of network architecture. In this research we use ResNet101 [8], which is a 101 layers network consist of combination convolutional layers, ReLU layers, max pooling layer, and fully-connected layers. What makes ResNet is special is its feature of Residual Network from which this network got its name. This residual network boots its performance to get much faster processing time with higher accuracy. ResNet has become the winner of ImageNet Large Scale Visual Recognition Competition (ILSVRC) 2015 with performance of its top-5 error only 5.71%.
  We combine ResNet101 with RPN as in [6] to localize detected human in footage. By using RPN we can get bounding boxes that surround detected human. With this combination of network we are not only can detect if there are human in out footage but we also can have information about its position from generated bounding boxes.
- Learning Process
  This research trained the proposed method using modification of PASCAL VOC [9] dataset. The dataset have been modified to be one-class classification which is person class. The ResNet101 architecture trained with 70 epochs and 10000 iteration. It means this learning proces doing 70,000 times forward and backward pass. We use Keras and TensorFLow [10] library for code implementation for this method. We achieve high accuracy level at 76.4% with very fast image processing at 0.145s.
- Indicating Human Position
  As we use RPN layer for localize human in our image, we can achieve bounding boxes for each detected human in order *(xmin,xmax,ymin,ymax)*. With simple calculation, we can have *(x,y)* position of detected human from tested image. These two information will be delivered to tracking algorithm on the next section.

C. Tracking Algorithm

For tracking object within video footage we are using OpenCV library implemented in python. OpenCV is an open-source library for computer vision. With OpenCV we can do many things about computer vision and one of them is tracking object. OpenCV has some library on object tracking algorithm such as MIL, MOSSE, TLD, KCF, MedianFlow, GOTURN, Boosting, and CSRT. From all tracking algorithm provided, in this research we use CSRT Tracker as in [11] its given very high performance comparing with other tracker available in OpenCV.

CSRT Tracker in OpenCV is an implementation of Discriminative Correlation Filter tracker with Channel and Spatial Reliability or CSR-DCF by [12]. This tracker use famously discriminative correlation filter with addition of spatial reliability channel to boost performance on tracking object. This tracker works by processing initialized bounding boxes from first frame with next frame. It produces new bounding boxes for next frame. This tracker is very easy and reliable to use for various environment with high speed processing for each frame.

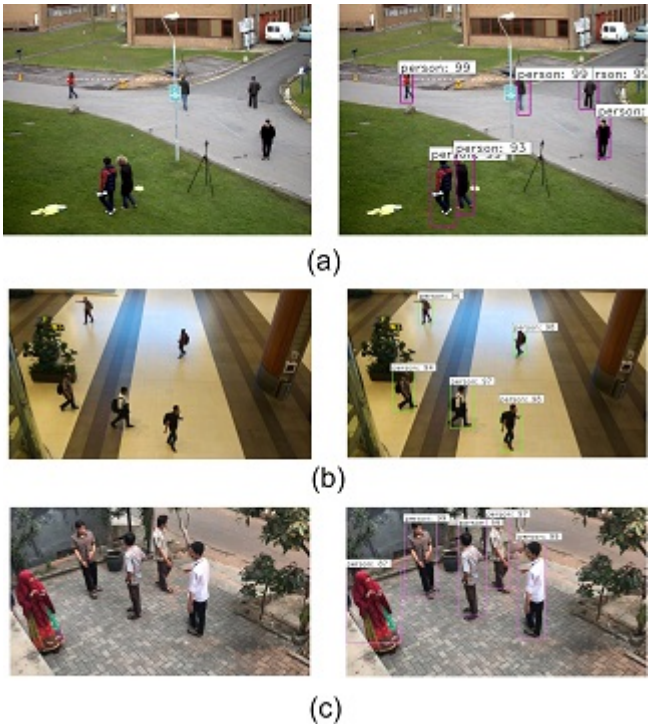

Fig. 3. Detection result on (a)PETS2009, (b) Airport scheme, and (c) Backyard scheme.

D. Detection and Tracking Combination

If we use only detection framework to detect human in each frames, it will high processing resource and time. Human detection took 0.145s to process one image. It means for 15fps video will need at least 2 second processing time to detect human on one second video footage. We provide solution for this problem by not using human detection on each frame in video footage but only several time for each second. By using this solution we can decrease the processing time on processed video footage.

Images processed with detection system will provide bounding boxes for each human detected. These bounding boxes contain four pixel coordinate of image, where $bboxes = (x_{start}, y_{start}, x_{end}, y_{end})$. We feed these bounding boxes from human detection framework to our tracker. By its process, this tracker correlate the detected human feature from previous frame with next frame to get new bounding boxes for the next frame. From this process, we have information about human location from its bounding boxes for each frame.

The next step is assign an ID for each detected or tracked human. We do this so we can identify tracked movement for each human. First we need to know the human exact location within images. For this problem, we can use information from

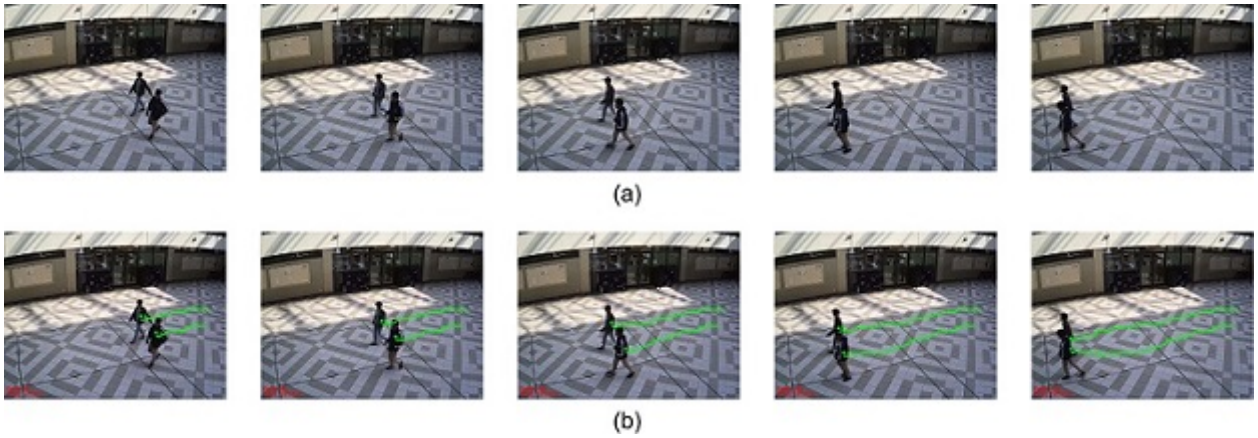

Fig. 4. Here we can see the result of human tracking algorithm. Each human detected assigned by different ID as its identity. Detected human provided with its position in footage. We can plot the tracking result of human movement by using its position from previous frame.

bounding boxes for each human detected. Bounding boxes provide *(x_{start},y_{start}, x_{end},y_{end})* and we can search its center by:

$$center(x, y) = ((x_{\text{start}} + x_{\text{end}})/2, (y_{\text{start}} + y_{\text{end}})/2) \quad (1)$$

From (1) we can use *(x,y)* information as location for each ID. We do this for each frame to get ID location. Next we need to assign each ID within frames to make sure it's not confused. For each ID in a frame we calculate its euclidean distance with each ID in previous frame. Then we assign lowest euclidean distance value as the same ID within the frames. As a result, we get information about ID and its location as *(ID,x,y)*.

IV. Experimental Result

In this experiments, we are using high performance personal computer with specification of Intel Core i7-6700K processor with 24GB of Random Access Memory and 8GB of GPU Memory with 2560 CUDA Cores based on Ubuntu 16.04LTS operating system. Learning and predicting experiments run on TensorFlow and Keras machine learning framework as described before. We are testing our proposed method on PETS2009 and our own dataset.

We test our human detection framework on several image as shown in Fig. 3. Acquired image then processed with our method giving result of detected human complete with its bounding boxes. We also add additional information which is confidence level of detected human. As we can see our human detection framework capable to detect human on different environment.

We test our human detection and tracking framework on our own dataset, PSVR2019, as shown in Fig. 4. Input video first separated into frame-by-frame images. These image first processed by our human detection framework to get its detected human location. Next process is we use tracking algorithm for the next frame so we can store the human location from each frame. We can see that our research were able to track human within the footage and plot its movement from previous frame. Stored data can also be extracted to be used for another use as shown in Fig. 5. We can plot the result of our tracked persons

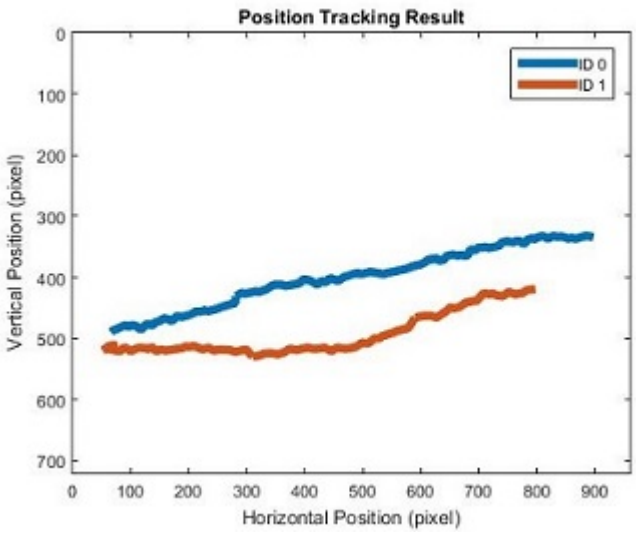

Fig. 5. Tracked plot from previous result can be stored and displayed independently. We can use this information for another research such as action classification.

independently. This result can lead to another research such as action classification or action recognition.

We also try to compare processing time needed for different method. First, we try to use full detection for each video footage tested. Second, we try to use detection for each 3 frames. It means we do 5 detections per one second footage. And last, we only use detection once per one second video footage. We can see the result at Table I shows combination of detection and tracking given faster processing time than using detection for all frames.

V. Conclusion

We have provided our research on human detection and tracking for surveillance video footage. We use deep convolutional neural networks for human detection framework and use tracking algorithm provided by OpenCV to track the detected

TABLE I
PROCESSING TIME COMPARISON

Video Duration (second)	Processing Time (second)		
	Detection Only	5 Detections per Second	1 Detection per Second
8	19.08	11.693	8.738
11	26.235	16.078	12.015
14	23.391	20.463	15.293
16	38.162	23.386	17.477

human. From this result we can see that our proposed method giving an excellence result both for detecting and tracking human on surveillance video footage with faster processing time. This result is very important for further research on surveillance video footage.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in neural information processing systems, 2012, pp. 1097–1105

[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014

[4] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," IEEE transactions on pattern analysis and machine intelligence, vol. 35, no. 8, pp. 1915–1929, 2013.

[5] R. Girshick, "Fast r-cnn," in Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1440–1448.

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580–587.

[7] Ferryman, James, and Ali Shahrokni. "Pets2009: Dataset and challenge." 2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance. IEEE, 2009.

[8] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[9] Everingham, Mark, et al. "The pascal visual object classes (voc) challenge." International journal of computer vision 88.2 (2010): 303-338.

[10] Abadi, M et. al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[11] Kristan, Matej, et al. "The visual object tracking vot2017 challenge results." Proceedings of the IEEE International Conference on Computer Vision. 2017.

[12] Alan Lukezic, Tom'as Voj'ir, Luka Cehovin Zajc, Jir'i Matas, and Matej Kristan. Discriminative correlation filter tracker with channel and spatial reliability. International Journal of Computer Vision, 2018.

[13] G. Lavee, L. Khan, and B. Thuraisingham, "A framework for a video analysis tool for suspicious event detection," Multimedia Tools Appl., vol. 35, pp. 109–123, 2007.

[14] Y. Zhang and Z. Liu, "Irregular behavior recognition based on treading track," in Proc. Int. Conf. Wavelet Anal. Pattern Recog., 2007, pp. 1322–1326.

[15] H. Zhong, J. Shi, and M. Visontai, "Detecting unusual activity in video," in Proc. 2004 IEEE Comput. Vis. Pattern Recog., 2004, vol. 2, pp. II-819–II-826.

[16] P. Feng and W. Weinong, "Anomaly detection based-on the regularity of normal behaviors," in Proc. 1st Int. Symp. Syst. Control Aerosp. Astronautics, Jan.19–21, 2006, pp. 1041–1046.

[17] Y. Benezeth, P. Jodoin, V. Saligrama, and C. Rosenberger, "Abnormal events detection based on spatio-temporal co-occurrences," in Proc. IEEE Conf. Comput. Vision Pattern Recog. Workshops, Jun.20–25, 2009, pp. 2458–2465.