An Efficient Method for Generating Training Data for AI Retraining in Edge Video Surveillance Systems

Chang Mo Yang
Information Media Research Center
Korea Electronics Technology Institute
Seoul, Korea
cmyang@keti.re.kr

Dongchil Kim
Information Media Research Center
Korea Electronics Technology Institute
Seoul, Korea
dckim@keti.re.kr

Kyeongeun Seo
Information Media Research Center
Korea Electronics Technology Institute
Seoul, Korea
ke.seo@keti.re.kr

***Abstract*—In this paper, we propose an efficient training data generation method for AI retraining in edge video surveillance systems. The proposed method performs clustering on multiple edge devices, then generates training data and retrains AI at the cluster level. The training data for AI retraining is generated using video analysis results from edge devices within the cluster. Unlike existing methods that manually generate fragmented ground truth information, the method proposed in this paper utilizes various forms of ground truth information to comprehensively describe the behavior of video objects. The ground truth information generated through this process is converted into XML metadata and stored in a database. Implementation results demonstrate that the proposed method can rapidly generate ground truth information in various formats.**

Keywords—training data, AI retraining, edge video surveillance systems, edge devices, ground truth information

I. Introduction

Recently, active research has been conducted on video surveillance systems that analyze CCTV video to automatically identify dangerous situations [1,2]. Typical video surveillance systems primarily use a method whereby video captured by CCTV is transmitted to a video management system (VMS), which then analyzes and stores the received video in real time [3-6]. However, this method has the disadvantage of increasing the complexity of the VMS as the number of CCTV installations increases. To address these issues, edge video surveillance systems have recently gained attention [7,8]. Instead of traditional CCTV, edge video surveillance systems utilize edge devices, which combine camera modules with edge modules. The edge device analyzes video captured by the camera module in real time and transmits the captured video and analysis results to the VMS. The VMS then uses the transmitted video and analysis results to store, search, and display them. However, because this method relies on lightweight AI embedded in the edge device, AI performance can degrade significantly if issues such as data drift occur. If this performance degradation occurs, AI retraining is necessary, and training data for this retraining is required.

AI retraining requires training videos along with annotation information that describes the details of the video. This annotation information is called ground truth (GT) information for the video, and the process of generating GT is called data labeling. Typically, existing data labeling methods manually generate video GT information [9-12]. However, these data labeling methods require significant manpower and time to manually generate GT information for a vast amount of training videos.

In this paper, we propose an efficient method for generating training data for AI retraining in edge video surveillance systems. Unlike existing methods that manually generate GT information for training data, the method proposed in this paper generates training data using existing video analysis results stored in the VMS. Since the video analysis results from individual edge devices are too small to be used as training data, edge devices installed in similar locations are grouped into clusters. Then, the video and analysis results from edge devices in the same cluster are used to generate training data for AI retraining. The AI generated through this process is then installed and applied to edge devices included in the same cluster.

II. Proposed Training Data Generation Method

A. Edge Video Surveillance System Architecture Using AI Retraining

Fig. 1 illustrates the technical architecture of the edge video surveillance system used in this paper. As shown in Fig. 1, the edge video surveillance system consists of edge devices, edge AI technology, edge AI monitoring technology, edge AI retraining technology, edge AI deployment technology, and video surveillance technology. Edge devices consist of edge-integrated CCTVs and multi-channel edge AI boxes, which allow for replaceable or upgradable AI. Edge AI technology, embedded in the edge module of the edge device, analyzes video captured through the camera module in real time and transmits the analyzed results to external systems. Edge AI monitoring technology detects performance degradation of the AI embedded in the edge device, and edge AI retraining technology generates new AI when performance degradation occurs. Edge AI deployment technology installs and operates the newly generated AI on the edge device, while video surveillance technology stores, searches, and displays the video transmitted from the edge device and the video analysis results.

B. Method for Generating Training Data for AI Retraining

The edge video surveillance system in Fig. 1 monitors performance degradation of AI embedded in edge devices and performs AI retraining and deployment based on the monitoring results. However, performing AI retraining and deployment simply because performance degradation occurs on individual edge devices is inefficient. To address this issue, the edge video surveillance system proposed in this paper clusters multiple edge devices installed in similar locations, and then performs AI performance degradation detection, AI retraining, and AI deployment for each cluster.

When AI performance degradation is detected in a specific cluster using the edge AI monitoring technology in Fig. 1, AI retraining is performed. AI retraining requires training data, and the method proposed in this paper generates training data using existing video analysis results stored in the VMS. To perform this process, GT is automatically generated using the video analysis results analyzed by edge devices in the cluster and stored in the VMS. The user then inspects and modifies

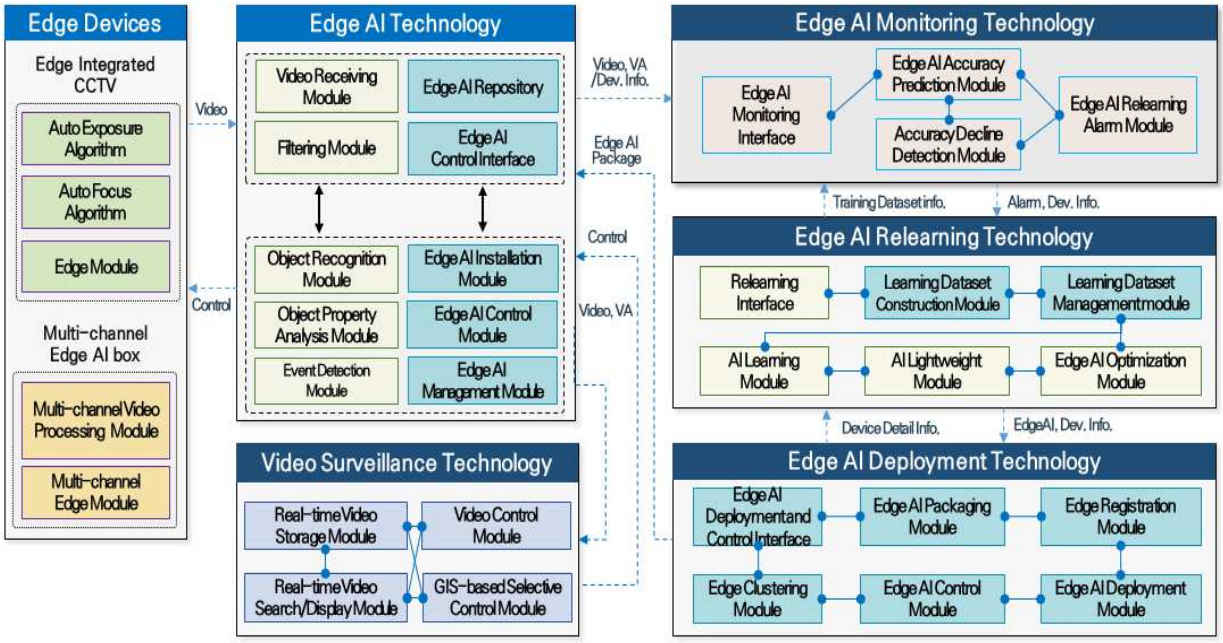


Fig. 1. Technical architecture of edge video surveillance system

the automatically generated GT, completing the training data generation for AI retraining.

C. Training Data Structure for AI Retraining

Fig. 2 illustrates the training data structure for AI retraining proposed in this paper. As shown in the Fig. 2, the training data is converted to XML and stored as elements such as <source>, <object>, <face>, and <event>.

Fig. 3 and Fig. 4 show the data structure of <source> and <object>. <source> describes information about the input video and consists of the input type, folder, file name, file path, and size information. Size information describes the width, height, and bytes per pixel of the input video. <object> describes information about a person object present in each frame image of the video input. <object> consists of a list of <sobject>, which describe individual object information. Individual object information includes the frame number, frame image information, object bounding box information, object attribute information, object pose information, object clipping information, and object detection difficulty information. Frame image information includes image location and object-level clipping information. The bounding box information includes the object's upper-left and lower-right x and y coordinates. Object attribute information includes gender, age, striped top, sleeve length, top style, bottom style, and possession information.

Fig. 5 and Fig. 6 shows the data structure of <face> and <event>. <face> describes the face information of a human object contained in the input video. <face> consists of a list of <sface>, each of which describes an individual face object.

```
<annotation>
  <source> </source>
  <object> </object>
  <face> </face>
  <event> </event>
</annotation>
```

Fig. 2. Training data structure

```
<source>
  <type> </type>
  <folder> </folder>
  <filename> </filename>
  <path> </path>
  <size>
    <width> </width>
    <height> </height>
    <depth> </depth>
  </size>
</source>
```

Fig. 3. Data structure of <source>

```
<object>
  <sobject>
    <nframe> </nframe>
    <image>
      <imagepath> </imagepath>
      <segmented> </segmented>
    </image>
    <bndbox>
      <xmin> </xmin>
      <ymin> </ymin>
      <xmax> </xmax>
      <ymax> </ymax>
    </bndbox>
    <attribute>
      <gender> </gender>
      <agegroup> </agegroup>
      <toppattern> </toppattern>
      <topsleevelength> </topsleevelength>
      <topstyle> </topstyle>
      <pantsstyle> </pantsstyle>
      <belongings> </belongings>
    </attribute>
    <pose> </pose>
    <truncated> </truncated>
    <difficult> </difficult>
  </sobject>
</object>
```

Fig. 4. Data structure of <object>

```
<face>
  <sface>
    <nframe> </nframe>
    <image>
      <imagepath> </imagepath>
      <segmented> </segmented>
    </image>
    <bndbox>
      <xmin> </xmin>
      <ymin> </ymin>
      <xmax> </xmax>
      <ymax> </ymax>
    </bndbox>
    <points>
      <p0> </p0>
      <p1> </p1>
      <p2> </p2>
      <p3> </p3>
      <p4> </p4>
    </points>
    <mask> <mask>
  </sface>
</face>
```

Fig. 5. Data structure of <face>

```
<event>
  <sevent>
    <startframe> </startframe>
    <endframe> </endframe>
    <type> </type>
  </sevent>
</event>
```

Fig. 6. Data structure of <event>

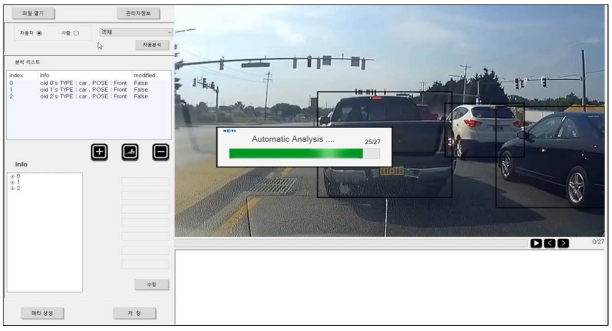
Each face object is composed of a frame number, frame image information, face bounding box information, face point information, and mask-wearing information. Frame image information consists of image position information and information on whether the image is cropped at the object level. Face bounding box information consists of the upper-left x and y coordinates and the lower-right x and y coordinates. Face point information consists of the left eye position, right eye position, upper nose position, left corner of the mouth position, and right corner of the mouth position. <event> describes event occurrence time information based on the input video. Events are categorized as weapon possession, assault, crowding, loitering, fire, flooding, collapse, etc. <event> consists of a list of <sevent>, which describe information about multiple individual events. Each event consists of a start frame number, an end frame number, and event type information.

III. Implementation Results

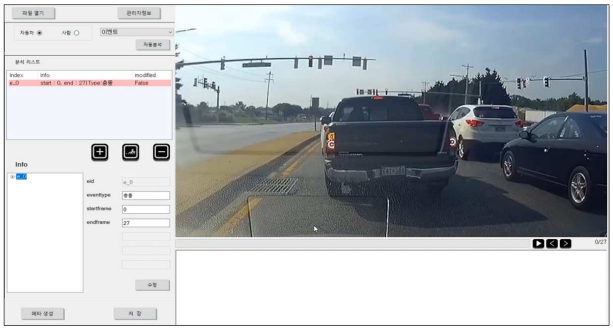
Fig. 7 shows the implementation results of the training data generation method proposed in this paper. Fig. 7(a)-(d) show automatic analysis, GT modification, GT metadata generation, and GT metadata generation results using video and video analysis results stored in the VMS, respectively. As shown in Fig. 7(a), the proposed method provides the ability to automatically generate GT using video analysis results analyzed on the edge device and stored in the VMS. Then, as shown in Fig. 7(b), the user can modify the automatically generated GT information or create new GT information. Upon completion of this process, the GT metadata generation process is performed, as shown in Fig. 7(c). The generated GT metadata is stored in the database in XML format, as shown in Fig. 7(d).

IV. Conclusions

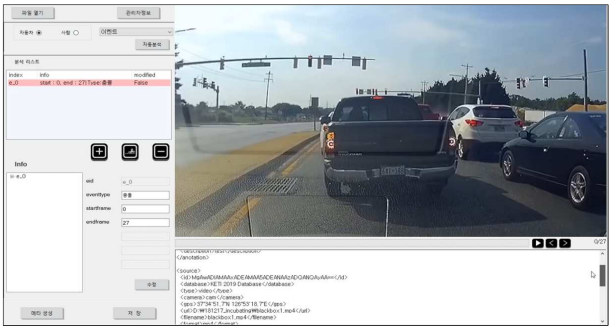
In this paper, we proposed an efficient training data generation method for AI retraining in edge video surveillance systems. The proposed method uses a semi-automatic labeling



(a) Automatic analysis



(b) GT modification



(c) GT metadata generation

(d) GT Metadata generation results

Fig. 7. Implementation results of the proposed training data generation method

method that automatically generates GT information using video and analysis results captured and analyzed at an edge terminal and stored in a VMS, and then allows the user to modify the automatically generated GT information if necessary. Since the number of videos and analysis results from individual edge devices is too small to be used as training data, edge devices installed in similar locations are grouped into clusters, and the training data is generated using the videos and analysis results from edge devices in the same cluster. The implementation results demonstrate that the proposed method can rapidly generate various types of GT information, such as source information, object information, face information, and event information. In addition, the implementation results show that the proposed method is suitable for application to edge video surveillance systems.

ACKNOWLEDGMENT

This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea Government (MIST) (No. RS-2024-00437576, Development of autonomous performance improvement technology for video surveillance system based on edge-analysis server connection).

REFERENCES

[1] N. Dilshad, J. Hwang, J. Song, and N. Sung, "Applications and Challenges in Video Surveillance via Drone: A Brief Survey", Int'l Conf. on Info. and Comm. Tech. Convergence, pp. 728-732, Oct. 2020.

[2] H. Kim, Y. Cha, T. Kim, and P. Kim, "A Study on the Security Threats and Privacy Policy of Intelligent Video Surveillance System Considering 5G Network Architecture", Int'l Conf. on Electronics, Info., and Comm., pp. 1-4, April 2020.

[3] D. S. Goud, Prasannakiruba G S, N. G. Vidhya, Mohanraj. M, Rajalakshmi S, and M. Venkatasudhahar, "AI in Real-Time Video Surveillance Systems," Global Conference in Emerging Tech., pp. 1-4, May 2025.

[4] H. Sun, W. Shi, X. Liang, and Y. Yu, "VU: Edge Computing-Enabled Video Usefulness Detection and its Application in Large-Scale Video Surveillance Systems," IEEE Internet of Things Journal, vol. 7, no. 2, pp. 800 - 817, Feb. 2020.

[5] J. Park, and S. Yi, "Development of Video Database and a Video Annotation Tool for Evaluation of Smart CCTV System", Journal of Korea Institute of Electronic Comm. Science, vol. 9, no. 7, pp. 739-745, July 2014.

[6] L. Li, W. Huang, I. Gu, R. Luo and Q. Tian, "An Efficient Sequential Approach to Tracking Multiple Objects Through Crowds for Real-Time Intelligent CCTV Systems", IEEE Trans. on Systems, Man, and Cybernetics, Part B (Cybernetics), vol. 38, no. 5, pp. 1254 - 1269, Oct. 2008.

[7] R. P. Singh, H. Srivastava, H. Gautam, R. Shukla, and R. K. Dwivedi. "An Intelligent Video Surveillance System using Edge Computing based Deep Learning Model," Int'l Conference on Intelligent Data Communication Technologies and Internet of Things, pp. 439-444, January 2023.

[8] M. G. Ismail, F. H. Tarabay, R. E. Masry, M. A. E. Ghany, and M. A. M. Salem, "Smart Cloud-Edge Video Surveillance System," Int'l Conference on Modern Circuits and Systems Tech., pp. 1-4, June 2022.

[9] M. Liu, Y. Bian, Q. Liu, X. Wang, and Y. Wang, "Weakly Supervised Tracklet Association Learning With Video Labels for Person Re-Identification," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 46, no. 5, pp. 3595 – 3607, May 2024.

[10] M. Mobaraki, S. Ahani, K. M. Yi, M. Asadi, K. V. Heusden, and G. A. Dumont, "Efficient Multi-purpose Video Annotation for Fast Labeling," IEEE Int' l Conference on Visual Communications and Image Processing, pp. 1-5, January 2024.

[11] J. Park, and S. Yi, "Development of video database and a video annotation tool for evaluation of smart CCTV system," Journal of Korea Institute of Electronic Communication Science, vol. 9, no. 7, pp. 739-745, July 2014.

[12] J. S. L. Villa, H. D. I. Ceballos, S. M. Giraldo, A. A. Meza, G. C. Dominguez, "A novel tool for ground truth data generation for video-based object classification," Symposium on Signal Processing, Images and Computer Vision, pp. 1-6, September 2015.