Towards Data-efficient Continuous Learning for Edge Video Analytics via Smart Caching

Lei Zhang
lei.zhang@njust.edu.cn
Nanjing University of Science and Technology
China

Guanyu Gao*
gygao@njust.edu.cn
Nanjing University of Science and Technology
China

Huaizheng Zhang
huaizhen001@e.ntu.edu.sg
Nanyang Technological University
Singapore

ABSTRACT

Continuous learning (CL) has recently been adopted into edge video analytics, gaining huge success in maintaining high accuracy without constantly retraining DNN models by human intervention. Though existing solutions offer optimized processing pipelines, the cost brought by CL should not be neglected. This vision paper starts an investigation by exploring two kinds of cost, human labeling and edge storage. The former comes from the need for CL's automatically tuning, and the latter is due to an exemplar pool (including both drift and historical data) maintained to prevent catastrophic forgetting caused by naive retraining. To alleviate the costs, we propose a new CL-based edge video analytics system by incorporating an active learner mechanism. Specifically, we revisit the current CL video system design and develop an active CL pipeline atop them. The pipeline first accepts the drift data stored in drift pool and utilizes an active learner to sample a small partition of them for labeling. Then it mixes up both small labeled drifted data and some historical data to send them to an exemplar pool for CL. Our preliminary benchmark studies exhibit that the new system can achieve competitive accuracy by spending only 30% labeling and storage cost compared to other baselines, showing a promising research direction for future study.

CCS CONCEPTS

• **Information systems → Multimedia information systems**; • **Theory of computation → Active learning**; • **Computing methodologies → Lifelong machine learning**.

KEYWORDS

video analytics, active learning, continual learning

ACM Reference Format:
Lei Zhang, Guanyu Gao, and Huaizheng Zhang. 2022. Towards Data-efficient Continuous Learning for Edge Video Analytics via Smart Caching. In *The 20th ACM Conference on Embedded Networked Sensor Systems (SenSys '22), November 6–9, 2022, Boston, MA, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3560905.3568430

*Corresponding author.

SenSys '22, November 6–9, 2022, Boston, MA, USA

ACM ISBN 978-1-4503-9886-2/22/11…$15.00
https://doi.org/10.1145/3560905.3568430

1 INTRODUCTION

Video analytics systems aim to provide high-accuracy video content recognition in a resource-efficient manner (e.g., low cost and high throughput). These systems are pervasive in today's industry, supporting various applications ranging from video surveillance [10, 21, 34] and traffic monitor [2, 29], to smart retail [1, 25] and wildlife protection [20]. The successes of these video applications heavily rely on the recent advances in deep neural networks (DNNs) models trained on large-scale datasets to achieve even superhuman performance [5, 17].

However, DNN-based recognition models often contain billions of parameters, resulting in a high latency inference speed if not deployed into high-performing hardware (e.g., GPUs and TPUs). This new requirement brings new challenges to today's video analytics system design, especially systems deployed on edge devices. As edge devices always have no such powerful GPUs or TPUs same as clouds, both industry and academia are upgrading edge video analytics systems to meet the requirements of new scenarios.

Existing efforts can be generally grouped into three classes, 1) utilizing specialized binary classification or compressed models [6, 13], 2) automatically adjusting video parameters like resolution and frame rate [9, 28, 31, 32], and 3) designing edge-cloud continuum systems for cloud aid [27, 33, 35]. For example, NoScope [13] and FileterForward [6] trained a cascade of specialized models to reproduce the binarized outputs for inference. AWStream [31] and VideoStorm [32] adapted video frame rate, resolution, and encoding quality to reduce network cost and inference latency. VPaaS [33] and DICE-IoT [35] utilized edge-cloud synergy for joint latency- and accuracy-aware video analytics.

Though previous efforts have made significant progress, they still leave a large room to be improved. One of the challenging issues is the video data drift [23, 33], which means that after being deployed to real-world, the system accuracy will be dramatically degraded. The main reason is that it is hard to ensure that a training dataset has the same distribution as that of online data. Moreover, the system will soon become stale as the new data comes, whose class may not exist in training sets [26].

To eliminate the issue, some continuous learning based video analytic platforms have been proposed [3, 33]. For example, Ekya [3] utilized transfer learning to continuously distill a small and personalized model for drift scenarios. VPaaS [33] enabled the cloud- and fog- models to learn incremental knowledge as a human-in-the-loop schema. By incorporating continuous learning and designing new mechanisms, these systems can maintain a high and stable accuracy in real-world environments.

However, these systems did not consider the extremely high cost brought by continuous learning. The cost has two sides, namely, human labeling cost and edge storage cost. First, to adopt continuous learning for maintaining accuracy, they need to continuously send newly captured video frames for labeling. These video frames may contain many samples that can be recognized by current stale models and have no need to be labeled. Without careful selection, the human labeling cost will be very expensive. Second, the mainstream and best-performing continuous learning method is rehearsal-based, which maintains an exemplar pool for continuously retraining models on newly labeled and historical data. As video frames are very redundant, the exemplar pool will easily occupy a large edge storage if not being carefully improved.

To improve the current continuous learning-based edge video platforms, this paper presents ArcVideo, employing active learning to judiciously select representative drift video frames to be labeled and stored into the exemplar pool. Specifically, our system adopts a drift monitor to analyze the feature distribution of the video stream over time. When data drift occurs, the video data from the corresponding periods will be cached into a provisional drift pool. To reduce labeling cost, we employ the active learning strategy to sample informative drifted parts for labeling. Then, the labeled data along with some pre-stored historical data will be replayed into an exemplar pool for model retraining. During continuous learning, some representative data will be saved into the persistence storage to replace the historical data.

In this vision paper, we employ our ArcVideo to conduct a preliminary benchmark study. Experiments on both academic standard and real-world datasets show that our system can maintain a high accuracy, require fewer labeling efforts and occupy very small storage. Specifically, on the CORe50 benchmark, our system utilizes no more than 30% labeling efforts and storage to achieve competitive accuracy compared with naive continuous learning. Besides, on the real-world DashCam dataset, our system can achieve up to 47% AP accuracy in object detection while using only 30% labeled data for continuous learning. We will further investigate the problem and improve our system from both algorithm and system efficiency perspectives. The code will be released for research purposes.

2 RELATED WORK

The edge computing paradigm is widely adopted for video analytics applications due to the benefits of low data transmission delays and better data privacy protection [7, 16, 30]. However, when new data distribution significantly differs from pre-training data, the models will suffer from the data drift issue, and the accuracy will drop dramatically [33]. Moreover, the models deployed on the edge devices are always compressed with fewer parameters and layers [13], which makes the on-edge deployed models more prone to the data drift issue under the unseen scenarios [3].

2.1 Continuous Learning for Video Analytics

To address the data drift issues, continuous learning is a promising approach to enable the deployed models to continuously learn incremental knowledge from new scenarios to achieve lifelong intelligence. We illustrate the existing continuous learning systems for video analytics [3, 33] in Fig. 1. Ekya [3] periodically distills a shallower model from the golden models for every new scenario. It balances the trade-off between inference speed and model accuracy for ever-changing scenarios. However, Ekya's performance largely depends on the accuracy of the golden models and the model distillation approaches. If the pre-trained golden models fail to cover any unseen domains, the distilled model will be unable to handle the unseen drift scenarios. In contrast, VPaaS [33] enables the deployed models to continuously learn incremental knowledge from the drifted data. It proposed an interactive retraining platform for video analytics with the human-in-the-loop paradigm. However, this approach requires expensive human labor to obtain drifted data from video streams for labeling and retraining.

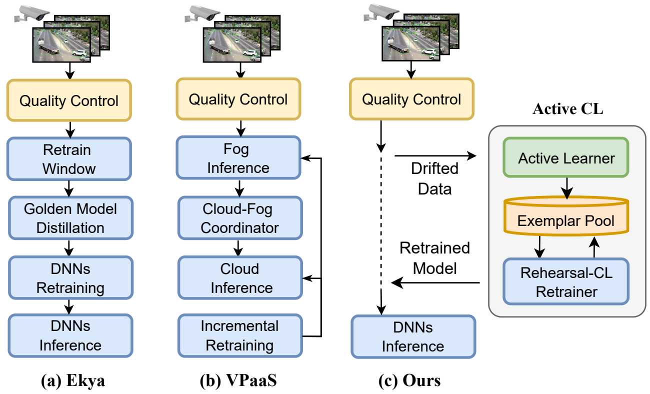

Figure 1: The comparison of different video analytic pipelines (VAP) with continuous learning. (a) Ekya [3] distills the personalized models for drift scenarios. (b) VPaaS [33] retrains models under a human-in-the-loop schema. (c) Our system designs an active learning based labeling- and storage-efficient continuous learning for edge video analytics.

2.2 Active Continuous Learning

Rehearsal-based strategy, such as iCaRL [22], can achieve the best performance for continuous learning in the on-edge deployment scenarios [14]. This approach enhances continuous learning at the expense of an exemplar pool for model retraining on newly labeled or historical data to prevent catastrophic forgetting. However, video frames are very redundant, the exemplar pool will easily occupy a large edge storage, which brings a mismatch between large storage requirements and limited edge storage resources. Recently, the rapid advance in active learning strategies (e.g., diversity-based [36], cluster-based [4, 24], uncertainty-based [8, 12]) and platforms (e.g., ALaaS [11]) brings a possible solution for large-scale rehearsal-based continuous learning. Active learning aims at retrieving informative or relevant instances from the data pools to reduce labeling costs. This strategy is naturally applicable for rehearsal-based continuous learning, where the redundant or similar data are filtered efficiently to reduce labeling cost and to preserve the representative data in the limited storage. To the best of our knowledge, we are one of the first works which studied active rehearsal-based continuous learning in the video analytics scenario.

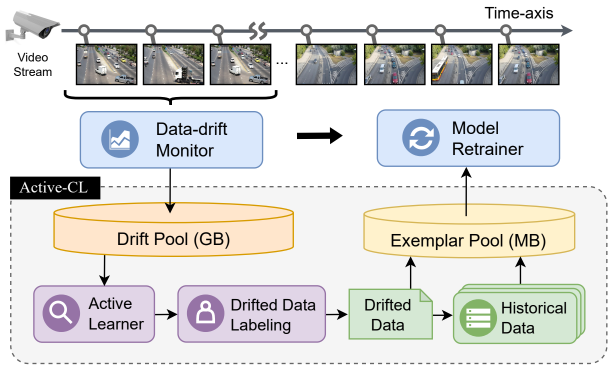

Figure 2: The workflows of our video analytic system. Our system can actively select representative drifted data for labeling and model retraining to reduce labeling and storage cost while improving accuracy.

3 SYSTEM DESIGN

In this section, we first present the design goals and the architecture overview. We then illustrate the core components and implementation details of our system.

3.1 Design Goal

To design a practical continuous learning based video analytics system for every-changing scenarios, we design the system with the following goals.

Continuous drift monitoring and retraining. The system can detect the data drift issue in time and provide continuous model retraining service for drifted scenarios.

Active sampling of retraining data. The system should actively select the representative retraining data from drift periods to reduce the labeling and storage costs.

Low labeling cost and high accuracy. Saving labeling costs for drifted data while maintaining a high retraining accuracy is an essential requirement in a practical video analytic system.

3.2 Architecture Overview

Fig. 2 depicts an overview of our video analytic system. Specifically, the video frames arrive as a stream, however, some drifted data are always mixed-in to affect the inference accuracy. To monitor the drift degree of the incoming video data, our system continuously analyzes the distribution of data features over time. When the distribution of data features shift, the video data of the corresponding periods will be cached in a large-size drift pool. However, using the full-size drift pool for model retraining is exhausting for both labeling and storage. Hence, we propose to utilize the active learning strategy to sample some representative drifted data from the drift pool. The sampled drifted data will be labeled interactively, and then cached in the exemplar pool along with some historical training data. The exemplar pool will supply retraining data periodically for continuous learning service to enhance the model performance for the drifted scenarios. Besides, during the continuous learning periods, some representative data will be saved in the persistence storage to replace the well-trained or outdated historical data to support future retraining.

3.3 Active Continuous Learning

Our system can perform as an active continuous learning (Active-CL) schema to provide a labeling- and storage- efficient retraining service for long-term video analytics. For the sake of simplicity, we divided our system into different decoupled modules, including drift monitor, active learner, exemplar manager, and model retrainer. We illustrate the procedure of these modules as follows.

3.3.1 Drift Monitor. The drift monitor can continuously monitor the changes in video streams, including feature distribution (e.g., new instances & classes), environment changes (e.g., from daytime to nighttime), and in-situ sensor shifts (e.g., RGB-camera v.s. IR-camera). Our drift monitor is built upon AzureML Data-drifter [23], which provides many easy-to-use approaches to simplify drift detection in real-world scenarios. The drift monitor can perform detection continuously during the life-cycle of video analytics. When the drift degree of input data exceeds a given threshold, the video data collected from the corresponding periods will be provisionally stored into a large-size drift pool.

3.3.2 Active Learner. Many video frames of the drifted videos can be recognized accurately and have no need to be labeled for retraining. To reduce the labeling and storage efforts, our system employs active learning to judiciously select informative drifted video frames for labeling. For example, one of the most popular active learning strategies, the least confidence [15], can select the video frames that have the lowest recognition confidences by the models from drift pool. The active learning strategies in our system are alternatively and configurable built upon ALaaS [11]. Specifically, when the drift pool is filled with video data, the active sampler will select a small partition from the drift pool based on the given active learning strategies. Then, the actively sampled partition will be interactively labeled under the human-in-the-loop paradigm. The human feedback on drifted data will be involved in the learning loop to enhance the model performance continuously. In general, the active learning strategy enables continuous learning on video analytics to be labeling- and storage- efficient.

3.3.3 Exemplar Manager. Rehearsal-based strategy can enhance continuous learning by retraining exemplars with drifted and historical data. Motivated by this, our system schedules a small-size exemplar pool by a balanced organization of drifted data and historical data. The exemplar pool will supply retraining data to enhance continuous learning. Besides, some representative drifted data will replace the well-trained and overdue historical data in the persistence storage for future retraining.

3.3.4 Model Retrainer. The deployed models on the edge side are retrained periodically to enhance the performance under ever-changing scenarios. When the edge resources are free-up, the model retrainer will perform continuous learning by fine-tuning or knowledge distilling manner. The fine-tune retraining will optimize the pre-deployed models to fit the data in the exemplar pool. The distill retraining will train a personalized model on the exemplar pool by

Table 1: The performance comparison of different continuous learning strategies under the CORe50 benchmark.

Model Retraining Configuration			Accuracy % (10 classes)		Accuracy % (50 classes)	
Strategy	**Label**	**CL-Pool**	**Top-1**	**Top-5**	**Top-1**	**Top-5**
Full-set	100%	4.82GB	86.71	99.81	83.11	98.29
Random	5%	253MB	40.08	84.48	30.07	60.48
Random	10%	433MB	48.01	91.01	44.91	80.79
Random	20%	792MB	66.41	96.16	62.36	90.82
Random	30%	1.12GB	77.66	98.76	69.71	93.73
Active-CL	5%	249MB	45.94	88.81	34.59	67.46
Active-CL	10%	451MB	52.38	92.70	46.56	80.40
Active-CL	20%	756MB	68.60	96.34	65.54	91.30
Active-CL	30%	1.15GB	80.80	99.22	78.85	97.16

distilling the knowledge from expert models. Both aforementioned approaches for model retraining are optional in our system.

4 EXPERIMENTAL RESULT

In this section, we first present the experimental settings, and then we evaluate the performance by comparing our method with other baselines using academic standard and real-world datasets.

4.1 Experimental Setting

Our continuous learning based video analytics system is developed with PyTorch. The drift monitor is implemented with AzureML Data-drift plugin [23]. We implement our active sampler with ALaaS [33] to support various active learning strategies. ALaaS contains a vast collection of alternative active learning methods for adoption. Our system is configurable with Python API for sampling, retraining, and backbone selection.

4.1.1 Dataset. We use a popular continuous learning benchmark, CORe50 [18], to simulate data drift. CORe50 contains abundant video data and classes to support the simulation of various drifted scenarios [26], such as new instances, new classes, and both of new instances and classes. Besides, we also use a real-world DashCam dataset, an object-tracking dataset collected from various illumination and camera views, to evaluate the continuous learning in real-life video scenarios, which is in line with the prior work [33].

4.1.2 Network Structure. We adopt ResNet18 as the backbone for the experiments on the CORe50 dataset due to the limited edge resources. The weights of the backbone are initialized randomly, and we use automatic mixed precision (AMP) for training to reduce on-edge memory usage [19]. For the DashCam dataset, we adopt MobileNet-V3 as the backbone of Faster RCNN for object detection. The backbone is initialized with the pre-trained weights, and other modules (e.g., RPN and ROI) are initialized randomly.

4.2 Evaluation on CORe50 Benchmark

We utilize a standard continuous learning benchmark, CORe50, to evaluate the performances of accuracy (Top-1, Top-5), labeling cost (Labeling), and exemplar pool usage (CL-Pool). The baseline approaches include 1) Full-set CL that utilizes full-scale drifted data for continuous learning, and 2) Random-sampling CL (Random) that randomly samples some parts of the drifted data for retraining. Our approach, Active-CL, can judiciously choose representative drifted data by active learning for model retraining. These approaches all adopt iCaRL [22] as the continuous learning strategy for retraining. We utilize the least confidence algorithm [15] as the active learner strategy for Active-CL in our experiments, which can achieve a tradeoff between labeling cost and continuous learning accuracy.

Table 1 compares the performances of different continuous learning strategies on the CORe50 benchmark. We can observe that our Active-CL can effectively reduce both the labeling cost and memory usage to achieve considerable accuracy. Specifically, our Active-CL outperforms random baseline by retraining no more than 30% drifted data to achieve competitive accuracy. In contrast, the Full-set CL achieves approximate accuracy at the expense of more than three times of labeling cost and exemplar pool usage compared with our Active-CL. This improvement is largely owing to the active learning strategy in our system, which can judiciously choose the representative drifted data for labeling and retraining.

Table 2: The performance comparison of different active learning strategies under real-world DashCam Dataset.

Strategy (label %)	AP	AP_{50}	AP_{75}	CL-Pool
Random (5%)	28.53	54.02	28.00	92MB
Random (10%)	37.47	65.81	36.99	175MB
Random (20%)	45.12	74.58	46.17	356MB
Random (30%)	45.91	76.41	46.91	510MB
K-means (5%)	38.55	68.02	37.68	87MB
K-means (10%)	44.23	75.35	44.83	191MB
K-means (20%)	45.44	75.75	46.98	366MB
K-means (30%)	47.91	77.60	49.82	537MB
Coreset (5%)	36.89	66.48	36.43	88MB
Coreset (10%)	42.16	73.42	41.42	203MB
Coreset (20%)	43.80	74.87	44.35	395MB
Coreset (30%)	47.49	77.49	48.90	524MB
Diversity (5%)	37.33	66.40	37.03	85MB
Diversity (10%)	44.67	74.49	45.67	176MB
Diversity (20%)	46.91	77.25	49.90	371MB
Diversity (30%)	47.84	79.09	48.76	498MB

4.3 Evaluation on DashCam Dataset

To further investigate the effectiveness of our system in real-world scenarios, we evaluate the performances of our system under different active learning strategies on the real-world dataset of DashCam. The Dashcam dataset consists of sufficient videos collected from various illumination and camera views by real-world vehicle cameras. We compare the average precision (AP) accuracy under three popular active learning strategies, namely, K-means sampling [4], diverse mini-batch sampling [36], and core-set selection approach [24]. As illustrated in Table 2, all of the active learning strategies can effectively facilitate continuous learning for video analytics. By utilizing no more than 30% drifted data for labeling and continuous learning, our system can achieve up to 47.84% AP accuracy on the real-world DashCam dataset.

5 CONCLUSION AND FUTURE WORK

This vision paper revisited continuous learning based edge video analytics systems and pointed out that the cost for these systems can be reduced. The paper designed a new CL video system by plugging an active learner. Specifically, the system first abstracts a general CL video pipeline. Then, it designs a plugin with an active learner to sample a small partition of informative drift data for labeling. Next, the labeled drift data will be mixed up with some historical data into an exemplar pool for CL. The preliminary benchmark studies show that our lightweight system can save 70% labeling and storage costs while maintaining a competitive accuracy. In the future, we will continue improving the system efficiency by 1) scheduling the active learner in more fine-grained configurations (e.g., sampling ratio or frequency) under the limited edge resources; 2) organizing the cross-device active learners judiciously to reduce the label and storage costs for multi-edge continuous learning in video analytics.

6 ACKNOWLEDGMENTS

This research was supported by the Natural Science Foundation of Jiangsu Province under Grant BK20200486 and NSFC No. 61972086.

REFERENCES

[1] Amazon. 2018. Amazon Go, automatic retail. https://www.amazon.com/b?node=16008589011. Accessed: 2018-01-22.

[2] Erhan Bas, A Murat Tekalp, and F Sibel Salman. 2007. Automatic vehicle counting from video for traffic flow analysis. In *2007 IEEE intelligent vehicles symposium*. Ieee, 392–397.

[3] Romil Bhardwaj, Zhengxu Xia, Ganesh Ananthanarayanan, Junchen Jiang, Yuanchao Shu, Nikolaos Karianakis, Kevin Hsieh, Paramvir Bahl, and Ion Stoica. 2022. Ekya: Continuous learning of video analytics models on edge compute servers. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22)*. 119–135.

[4] Zalán Bodó, Zsolt Minier, and Lehel Csató. 2011. Active learning with clustering. In *Active Learning and Experimental Design workshop In conjunction with AISTATS 2010*. JMLR Workshop and Conference Proceedings, 127–139.

[5] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258* (2021).

[6] Christopher Canel, Thomas Kim, Giulio Zhou, Conglong Li, Hyeontaek Lim, David G Andersen, Michael Kaminsky, and Subramanya Dulloor. 2019. Scaling video analytics on constrained edge nodes. *Proceedings of Machine Learning and Systems* 1 (2019), 406–417.

[7] Jiasi Chen and Xukan Ran. 2019. Deep learning with edge computing: A review. *Proc. IEEE* 107, 8 (2019), 1655–1674.

[8] Yarin Gal, Riashat Islam, and Zoubin Ghahramani. 2017. Deep bayesian active learning with image data. In *International Conference on Machine Learning*. PMLR, 1183–1192.

[9] Guanyu Gao, Linsen Dong, Huaizheng Zhang, Yonggang Wen, and Wenjun Zeng. 2019. Content-aware personalised rate adaptation for adaptive streaming via deep video analysis. In *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 1–8.

[10] Fadhil Hidayat et al. 2020. Intelligent video analytic for suspicious object detection: a systematic review. In *2020 International Conference on ICT for Smart Society (ICISS)*. IEEE, 1–8.

[11] Yizheng Huang, Huaizheng Zhang, Yuanming Li, Chiew Tong Lau, and Yang You. 2022. Active-Learning-as-a-Service: An Efficient MLOps System for Data-Centric AI. *arXiv preprint arXiv:2207.09109* (2022).

[12] Michael Kampffmeyer, Arnt-Borre Salberg, and Robert Jenssen. 2016. Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 1–9.

[13] Daniel Kang, John Emmons, Firas Abuzaid, Peter Bailis, and Matei Zaharia. 2017. Noscope: optimizing neural network queries over video at scale. *arXiv preprint arXiv:1703.02529* (2017).

[14] Young D Kwon, Jagmohan Chauhan, Abhishek Kumar, Pan Hui HKUST, and Cecilia Mascolo. 2021. Exploring system performance of continual learning for mobile and embedded sensing applications. In *2021 IEEE/ACM Symposium on Edge Computing (SEC)*. IEEE, 319–332.

[15] David D Lewis. 1995. A sequential algorithm for training text classifiers: Corrigendum and additional data. In *Acm Sigir Forum*, Vol. 29. ACM New York, NY, USA, 13–19.

[16] Yuanqi Li, Arthi Padmanabhan, Pengzhan Zhao, Yufei Wang, Guoqing Harry Xu, and Ravi Netravali. 2020. Reducto: On-camera filtering for resource-efficient real-time video analytics. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*. 359–376.

[17] Honghai Liu, Shengyong Chen, and Naoyuki Kubota. 2013. Intelligent video systems and analytics: A survey. *IEEE Transactions on Industrial Informatics* 9, 3 (2013), 1222–1233.

[18] V Lomanco and Davide Maltoni. 2017. Core50: a new dataset and benchmark for continual object recognition. In *Proceedings of the 1st Annual Conference on Robot Learning*. 17–26.

[19] Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. 2017. Mixed precision training. *arXiv preprint arXiv:1710.03740* (2017).

[20] Mohammad Sadegh Norouzzadeh, Anh Nguyen, Margaret Kosmala, Alexandra Swanson, Meredith S Palmer, Craig Packer, and Jeff Clune. 2018. Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning. *Proceedings of the National Academy of Sciences* 115, 25 (2018), E5716–E5725.

[21] Su-wan Park, Kyung-Soo Lim, and Jong Wook Han. 2012. Videos analytic retrieval system for CCTV surveillance. In *Future Information Technology, Application, and Service*. Springer, 239–247.

[22] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. 2017. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2001–2010.

[23] Microsoft Azure Research. 2022. AzureML, Data Drift Detector. https://aka.ms/datadrift. Accessed: 2022-09-11.

[24] Ozan Sener and Silvio Savarese. 2017. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489* (2017).

[25] Harkirat Singh. 2018. Applications of intelligent video analytics in the field of retail management: A study. In *Supply Chain Management Strategies and Risk Assessment in Retail Environments*. IGI Global, 42–59.

[26] Gido M Van de Ven and Andreas S Tolias. 2019. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734* (2019).

[27] Xuezhi Wang and Guanyu Gao. 2021. SmartEye: An Open Source Framework for Real-Time Video Analytics with Edge-Cloud Collaboration. In *Proceedings of the 29th ACM International Conference on Multimedia*. 3767–3770.

[28] Xuezhi Wang, Guanyu Gao, Xiaohu Wu, Yan Lyu, and Weiwei Wu. 2022. Dynamic DNN model selection and inference off loading for video analytics with edge-cloud collaboration. In *Proceedings of the 32nd Workshop on Network and Operating Systems Support for Digital Audio and Video*. 64–70.

[29] Yuankai Wu, Huachun Tan, Lingqiao Qin, Bin Ran, and Zhuxi Jiang. 2018. A hybrid deep learning based traffic flow prediction method and its understanding. *Transportation Research Part C: Emerging Technologies* 90 (2018), 166–180.

[30] Shanhe Yi, Zijiang Hao, Qingyang Zhang, Quan Zhang, Weisong Shi, and Qun Li. 2017. Lavea: Latency-aware video analytics on edge computing platform. In *Proceedings of the Second ACM/IEEE Symposium on Edge Computing*. 1–13.

[31] Ben Zhang, Xin Jin, Sylvia Ratnasamy, John Wawrzynek, and Edward A Lee. 2018. Awstream: Adaptive wide-area streaming analytics. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*. 236–252.

[32] Haoyu Zhang, Ganesh Ananthanarayanan, Peter Bodik, Matthai Philipose, Paramvir Bahl, and Michael J Freedman. 2017. Live Video Analytics at Scale with Approximation and {Delay-Tolerance}. In *14th USENIX Symposium on Networked Systems Design and Implementation (NSDI 17)*. 377–392.

[33] Huaizheng Zhang, Meng Shen, Yizheng Huang, Yonggang Wen, Yong Luo, Guanyu Gao, and Kyle Guan. 2021. A serverless cloud-fog platform for dnn-based video analytics with incremental learning. *arXiv preprint arXiv:2102.03012* (2021).

[34] Lei Zhang, Guanyu Gao, and Huaizheng Zhang. 2022. Spatial-Temporal Federated Learning for Lifelong Person Re-identification on Distributed Edges. *arXiv preprint arXiv:2207.11759* (2022).

[35] Yi Zhang, Jiun-Hao Liu, Chih-Yu Wang, and Hung-Yu Wei. 2020. Decomposable intelligence on cloud-edge IoT framework for live video analytics. *IEEE Internet of Things Journal* 7, 9 (2020), 8860–8873.

[36] Fedor Zhdanov. 2019. Diverse mini-batch active learning. *arXiv preprint arXiv:1901.05954* (2019).