International Journal of Trend in Scientific Research and Development (IJTSRD)
International Open Access Journal | www.ijtsrd.com

ISSN No: 2456 - 6470 | Volume - 3 | Issue – 1 | Nov – Dec 2018

Framework of Object Detection and Classification High Performance Using Video Analytics in Cloud

R. Gnana bharathy
Lecturer (Sr. Gr), Department of Computer Engineering,
Ayya Nadar Janaki Ammal Polytechnic College, Tamil Nadu, India

ABSTRACT

Video analytics framework detection performance is worked at cloud. Object detection and classification are the basic tasks in video analytics and become the initial point for other complex submissions. Old-fashioned video analytics approaches are manual and time consuming. These are particular due to the very participation of human factor. This paper present a cloud based video analytics framework for accessible and robust analysis of video streams. The framework enables an operative by programing the object detection and classification process from recorded video streams. An operative only specifies an analysis criteria and period of video streams to analyze. The streams are then realized from cloud storage, cracked and analyzed on the cloud. The framework performs compute severe parts of the analysis to CPU powered servers in the cloud. Vehicle and face finding are accessible as two case studies for assessing the framework, with one month of data and a 15 node cloud. The framework consistently performed object detection and classification on the data, comprising of 21,600 video streams and 175 GB in size, in 6.52 hours. The GPU enabled placement of the framework took 3 hours to perform analysis on the same number of video streams, thus making it at least double as fast than the cloud deployment Without GPUs. The analysis framework is high.

KEY WORDS: *Cloud Computing, Video Stream Analytics, Object Detection, Object Classification, High Performance, and GPU*

I. INTRODUCTION

Recent past has observed a rapid increase in the availability of inexpensive video cameras producing good quality videos. This led to a widespread use of these video cameras for security and monitoring purposes. The video streams coming from these cameras need to be analyzed for extracting useful information such as object detection and object classification. Object detection from these video streams is one of the important applications of video analysis and becomes a starting point for other complex video analytics applications. Video analysis is a resource intensive process and needs massive compute, network and data resources to deal with the computational, transmission and storage challenges of video streams coming from thousands of cameras deployed to protect utilities and assist law enforcement agencies. There are approximately 6 million cameras in the UK alone [1]. Camera based traffic monitoring and enforcement of speed restrictions have increased from just over 300,000 in 1996 to over 2 million in 2004 [2]. In a traditional video analysis approach, a video stream coming from a monitoring camera is either viewed live or is recorded on a bank of DVRs or computer HDD for later processing. Depending upon the needs, the recorded video stream is retrospectively analyzed by the operators. Manual analysis of the recorded video streams is an expensive undertaking. It is not only time consuming, but also requires a large number of staff, office work place and resources. A human operator loses concentration from video monitors only after 20 minutes [3]; making it impractical to go through the recorded videos in a time constrained scenario. In real life, an operator may have to juggle between viewing live and recorded video contents while searching for an object of interest, making the situation a lot worse especially when resources are scarce and decisions need to be made relatively quicker. Traditional video analysis approaches for object detection and classification such as color based [4], statistical background suppression [5], adaptive background [6], template matching [7] and Gaussian [8] are subjective, inaccurate and at times may provide incomplete monitoring results. There is also a lack of object classification in these approaches [4], [5], [8]. These approaches do not automatically produce color, size and object type information [5], [6]. Moreover, these approaches are costly and time consuming to such an extent that their usefulness is sometimes questionable [7], [9]. To overcome these challenges, we present a cloud based video stream analysis framework for object detection and classification.

The framework focuses on building a scalable and robust cloud computing platform for performing automated analysis of thousands of recorded video streams with high detection and classification accuracy. An operator using this framework will only specify the analysis criteria and the duration of video streams to analyze. The analysis criteria define parameters for detecting objects of interests (face, car, van or truck) and size/color based classification of the detected objects. The recorded video streams are then automatically fetched from the cloud storage, decoded and analyzed on cloud resources. The operator is notified after completion of the video analysis and the analysis results can be accessed from the cloud storage. The framework reduces latencies in the video analysis process by using GPUs mounted on computer servers in the cloud.

This cloud based solution offers the capability to analyze video streams for on-demand and on-the-fly monitoring and analysis of the events. The framework is evaluated with two case studies. The first case study is for vehicle detection and classification from the recorded video streams and the second one is for face detection from the video streams. We have

Selected these case studies for their wide spread applicability in the video analysis domain.

The following are the main contributions of this paper:

Firstly, to build a scalable and robust cloud solution that can perform quick analysis on thousands of stored/recorded video streams. Secondly, to automate the video analysis process so that no or minimal manual intervention is needed. Thirdly, achieve high accuracy in object detection and classification during the video analysis process. This work is an extended version of our previous work [10].

The rest of the paper is organized as followed: The related work and state of the art are described in Section II. Our proposed video analysis framework is explained in Section III. This section also explains different components of our framework and their interaction with each other. Porting the framework to a public cloud is also discussed in this section. The video analysis approach used for detecting objects of interest from the recorded video streams is explained in Section IV. Section V explains the experimental setup and Section VI describes the evaluation of the framework in great detail. The paper is concluded in Section VII with some future research directions.

II. RELATED WORK

Quite a large number of works have already been completed in this field. In this section, we will be discussing some of the recent studies defining the approaches for video analysis as well as available algorithms and tools for cloud based video analytics. An overview of the supported video recording formats is also provided in this section. We will conclude this section with salient features of the framework that is likely to bridge the gaps in existing research.

Object Detection Approaches

Automatic detection of objects in images/video streams has been performed in many different ways. Most commonly used algorithms include template matching [7], background separation using Gaussian Mixture Models (GMM) [11], [12], [13] and cascade classifiers [14]. Template matching techniques find a small part of an image that matches with a template image. A template image is a small image that may match to a part of a large image by correlating it to the large image. Template matching is not suitable in our case as object detection is done only for pre-defined object features or templates.

Video Analytics in the Clouds

Large systems usually consist of hundreds or even thousands number of cameras covering over wide areas. Video streams are captured and processed at the local processing server and are later transferred to a cloud based storage infrastructure for a wide scale analysis. Since, enormous amount of computation is required to process and analyze the video streams, high performance and scalable computational approaches can be a good choice for obtaining high throughputs in a short span of time. Hence, video stream processing in the clouds is likely to become an active area of research to provide high speed computation at scale, precision and efficiency for real world implementation of video analysis systems. However, so far major research focus has been on efficient video content retrieval using Hadoop [18], encoding/decoding [19], distribution of video streams [20] and on load balancing of computing resources for on-demand video streaming systems using cloud computing platforms [20], [21].

Supported Video Formats

CIF, QCIF, 4CIF and Full HD video formats are supported for video stream recording in the presented framework. The resolution (number of pixels present in one frame) of a video stream in CIF format is 352x288 and each video frame has 99k pixels. QCIF (Quarter CIF) is a low resolution video format and is used in setups with limited network bandwidth. Video stream resolution in QCIF format is 176x144 and each video frame has 24.8k pixels. 4CIF video format has 4 times higher resolution (704x576) than that of the CIF format and captures more details in each video frame. CIF and 4CIF formats have been used for acquiring video streams from the camera sources for traffic/object monitoring in our framework. Full HD (Full High Definition) video format captures video streams with 1920x1080 resolutions and contains 24 times more details in a video stream than CIF format. It is used for high resolution

An “Analysis Request” comprises of the defined region of Interest, an analysis criteria and the analysis time interval. The operator defines a region of interest in a video stream for an analysis. The analysis criteria define parameters for detecting objects of interests (face, car, van or truck) and size/color based classification of the detected objects. The time interval represents the duration of analysis from the recorded video streams as the analysis of all the recorded video streams might not be required.

Existing cloud based video analytics approaches do not support recorded video streams [22] and lack scalability [23], [24]. GPU based approaches are still experimental [28]. IVA 5.60 [25] supports only embedded video analytics and Intelligent Vision [27] is not scalable, otherwise their approaches are close to the approach presented in this research. The framework being reported in this paper uses GPU mounted servers in the cloud to capture and record video streams and to analyze the recorded video streams using a cascade of classifiers for object detection.

A cascade of classifiers (termed as HaarCascade Classifier)[14] is an object detection approach and uses real Adobos[17] Algorithm to create a strong classifier from a collection of weak classifiers. Building a cascade of classifiers is a time and resource consuming process. However, it increases detection performance and reduces the computation power needed during the object detection process. We used a cascade of classifiers for detecting faces/vehicles in video streams for the results reported in this paper.

A higher resolution video stream presents a clearer image of the scene and captures more details. However, it also requires more network bandwidth to transmit the video stream and occupies more disk storage. Other factors that may affect the video stream quality are video bit rate and frames per second. Video bit rate represents the number of bits transmitted from a video stream source to the destination over a set period of time and is a combination of the video stream itself and mate-data about the video stream. Frames per second (fps) represent the number of video frames stuffed in a video stream in one second and determines the smoothness of a video stream. The video streams have been captured with a constant bit rate of 200kbps and at 25 fps in the results reported in this paper.

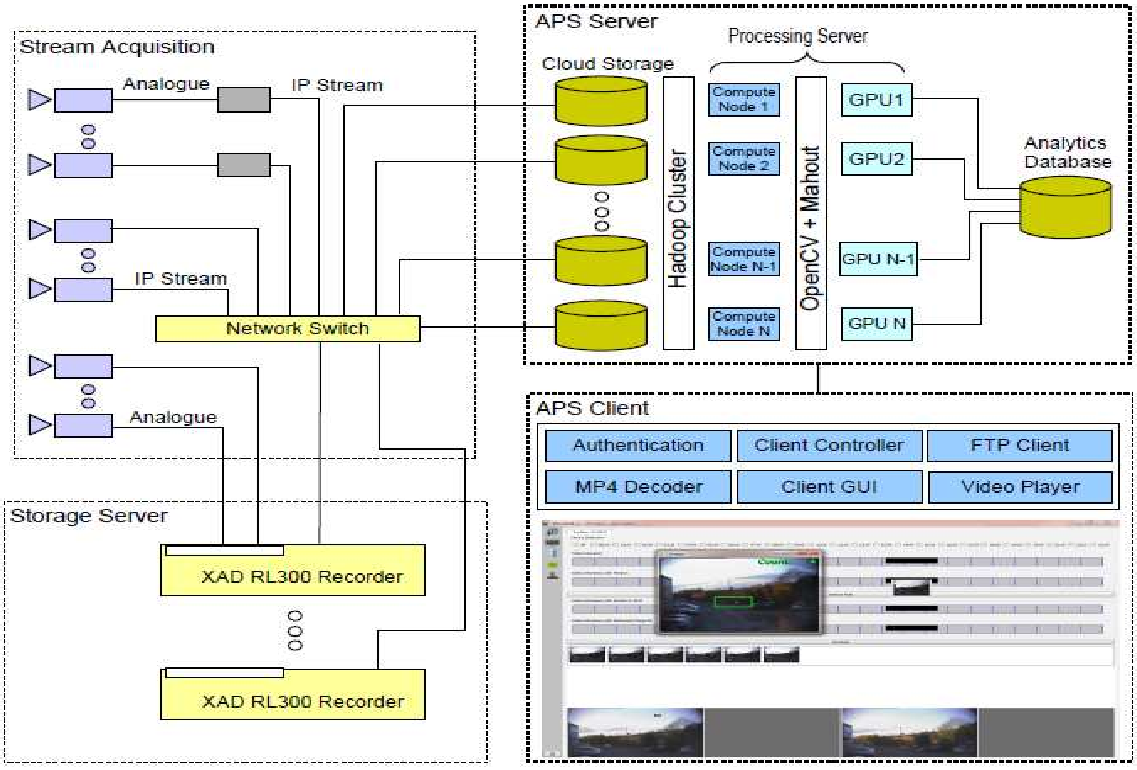


Figure 1: System Architecture of the Video Analysis Framework

III. VIDEO ANALYSIS FRAMEWORK

This section outlines the proposed framework, its different components and the interaction between them (Figure 1) the proposed framework provides a scalable and automated solution for video stream analysis with minimum latencies and user intervention. It also provides capability for video stream capture, storage and retrieval. This framework makes the video stream analysis process efficient and reduces the processing latencies by using GPU mounted servers in the cloud. Item powers a user by automating the process of identifying and finding objects and events of interest. Video streams are captured and stored in a local storage from a cluster of cameras that have been installed on roads/buildings for the experiments being reported in this paper. The video streams are then transferred to cloud storage for further analysis and processing. The system architecture of the video analysis framework is depicted in Figure 1 and the video streams analysis process on an individual compute node is depicted in Figure 2a. We explain the framework components and the video stream analysis process in the remainder of this section.

Automated Video Analysis: The framework automates the video stream analysis by reducing the user interaction during this process. An operator/user initiates the video stream analysis by defining an "Analysis Request" from the APS Client component (Figure 1) of the framework. The analysis request is sent to the cloud data center for analysis and no more operator interaction is required during the video stream analysis. The video streams, specified in the analysis request, are fetched from the cloud storage. These video streams are analyzed according to the analysis criteria and the analysis results are stored in the analytics database.

Framework Components

Our framework employs a modular approach in its design. At the top level, it is divided into client and server components (Figure 1). The server component runs as a daemon on the cloud machines and performs the main task of video stream analysis. Whereas, the client component supports multi-user environment and runs on the client machines (operators in our case). The control/data flow in the framework is divided into the following three stages:

- ➢ Video stream acquisition and storage
- ➢ Video stream analysis
- ➢ Storing analysis results and informing operators

The deployment of the client and server components is as follows: The Video Stream Acquisition is deployed at the video stream sources and is connected to the Storage Server through 1/10Gbps LAN connection. The cloud based storage and processing servers are deployed collectively in the cloud based data center. The APS Client is deployed at the end-user sites. We explain the details of the framework components in the remainder of this section.

Storage Server

The scale and management of the data coming from hundreds or thousands of cameras will be in extra bytes, let alone all of the more than 4 million cameras in UK. Therefore, storage of these video streams is a real challenge. To address this issue, H.264 encoded video streams received from the video sources, via video stream acquisition, are recorded as MP4files on storage servers in the cloud. The storage server has RL300 recorders for real time recording of video streams.

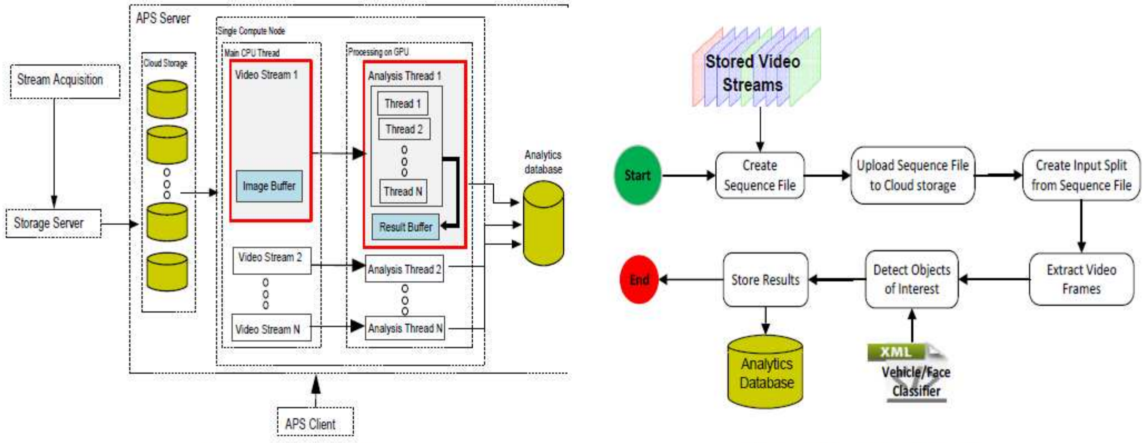
It stores video streams on disk drives and meta-data about the video streams is recorded in a database Analytics Processing Server (APS)

The APS server sits at the core of our framework and performs the video stream analysis. It uses the cloud storage for retrieving the recorded video streams and implements a processing server as compute nodes in a Hadoop cluster in the cloud data center (as shown in Figure 1). The analysis of the recorded video streams is performed on the compute nodes by applying the selected video analysis approach. The selection of a video analysis approach varies according to the intended video analysis purpose. The analytics results and meta-data about the video streams is stored in the Analytics Database.

APS Client

The APS Client is responsible for the end-user/operator interaction with the APS Server. The APS Client is deployed at the client sites such as police traffic control rooms or city council monitoring centers. It supports multi-user interaction and different users may initiate the analysis process for their specific requirements, such as object identification, object classification, or the region of interest analysis. These operator scan select the duration of recorded video streams for analysis and can specify the analysis parameters. The analysis results are presented to the end-users after an analysis is completed.

The analyzed video streams along with the analysis results are accessible to the operator over 1/10Gbps LAN connection from the cloud storage. The APS Client is deployed at the client sites such as police traffic control rooms or city council monitoring centers.



(a) *Schematic Diagram* (b) *Flow/Procedural Diagram*

Figure 2: Video Stream Analysis on a Compute Node

EXPERIMENTAL SETUP

This section explains the implementation and experimental details for evaluating the video analysis framework. The results focus on the accuracy, performance and scalability of the presented framework. The experiments are executed in two configurations; cloud deployment and cloud deployment with Nvidia GPUs.

The cloud deployment evaluates the scalability and robustness of the framework by analyzing different aspects of the framework including (i) video stream decoding time, (ii) vide data transfer time to the cloud, (iii) video data analysis time on the cloud nodes and (iv) collecting the results after completion of the analysis. The experiments on the cloud nodes with GPUs evaluate the accuracy and performance of the video analysis approach on state of the art compute nodes with two GPUs each. These experiments also evaluate the video stream decoding and video stream data transfer between CPU and GPU during the video stream analysis. The energy implications of the framework at different stages of the video analytics life cycle are also discussed towards the end of this section.

EXPERIMENTAL RESULTS

We present and discuss the results obtained from the two configurations detailed in Section V. These results focus on evaluating the framework for object detection accuracy, performance and scalability of the framework. The execution of the framework on the cloud nodes with GPUs evaluates the performance and detection accuracy of the video analysis approach for object detection and classification. It also evaluates the performance of the framework for video stream decoding, video stream data transfer between CPU and GPU and the performance gains by porting the compute intensive parts of the algorithm to the GPUs.

The cloud deployment without GPUs evaluates the scalability and robustness of the framework by analyzing different components of the framework such as video stream decoding, video data transfer from local storage to the cloud nodes, video data analysis on the cloud nodes, fault-tolerance and collecting the results after completion of the analysis. The object detection and classification results for vehicle/face detection and vehicle classification case studies are summarized towards the end of this section.

Performance of the Trained Cascade Classifiers

The performance of the trained cascade classifiers is evaluated for the two case studies presented in this paper i.e. vehicle and face detection from the recorded video streams. It is evaluated by the detection accuracy of the trained cascade classifiers and the time taken to detect the objects of interest from the recorded video streams. The training part of the real AdaBoost algorithm is not executed on the cloud resources. The cascade classifiers for vehicle and face detection are trained once on a single compute node and are used for detecting objects from the recorded video streams on the cloud resources.

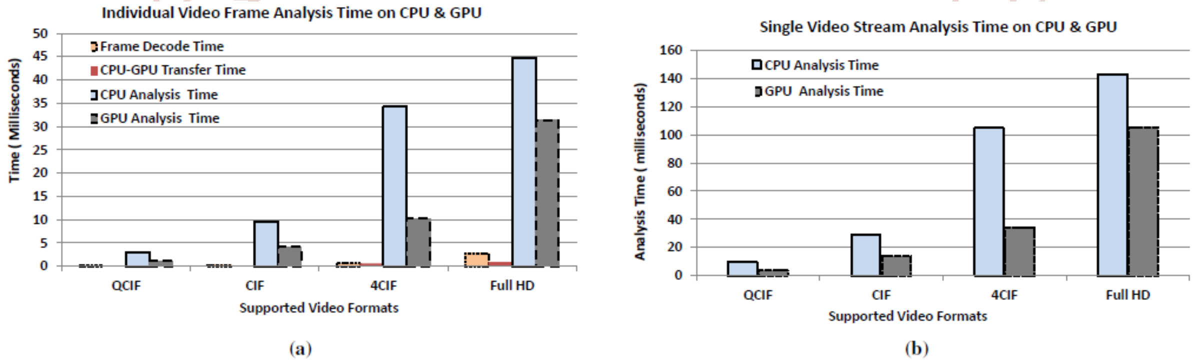


Figure 5: (a) Frame Decode, Transfer and Analysis Times for the Supported Video Formats, (b) Total Analysis Time of One Video Stream for the Supported Video Formats on CPU & GPU

CONCLUSIONS & FUTURE RESEARCH DIRECTIONS

The cloud based video analytics framework for automated object detection and classification is presented and evaluated in this paper. The framework automated the video stream analysis process by using a cascade classifier and laid the foundation for the experimentation of a wide variety of video analytics algorithms.

The video analytics framework is robust and can cope with varying number of nodes or increased volumes of data. The time to analyze one month of video data depicted a decreasing trend with the increasing number of nodes in the cloud, as summarized in Figure 9. The analysis time of the recorded video streams decreased from 27.80 hours to 5.83 hours, when the number of nodes in the cloud varied from 3-15. The analysis time would further decrease when more nodes are added to the cloud. The larger volumes of video streams required more time to perform object detection and classification. The analysis time varied from 6.38 minutes to 5.83 hours, with the video stream data increasing from 5GB to 175GB.

ACKNOWLEDGMENTS

This research is jointly supported by Technology Support Board, UK and XAD Communications, Bristol under Knowledge Transfer Partnership grant number KTP008832. The authors would like to say thanks to Tulsa Vamshi Mohan and Go khan Koch from the video group of XAD Communications for their support in developing the software platform used in this research. We would also like to thank Erol Kantardzhiev and AhsanIkram from the data group of XAD Communications for their support in developing the file client library. We are especially thankful to Erol for his expert opinion and continued support in resolving all the network related issues.

REFERENCES

1. “The picture in not clear: How many surveillance cameras are there in the UK?” Research Report, July 2013.
2. K. Ball, D. Lyon, D. M. Wood, C. Norris, and C. Raab, “A report on the surveillance society,” Report, September 2006.
3. M. Gill and A. Spriggs, “Assessing the impact of CCTV,” London Home Office Research, Development and Statistics Directorate, February 2005.
4. S. J. McKenna and S. Gong, “Tracking colour objects using adaptive mixture models,” Image Vision Computing, vol. 17, pp. 225–231, 1999.
5. N. Ohta, “A statistical approach to background supression for surveillance systems,” in International Conference on Computer Vision, 2001,pp. 481–486.
6. D. Koller, J. W. W. Haung, J. Malik, G. Ogasawara, B. Rao, and S. Russel, “Towards robust automatic traffic scene analysis in real-time,” in International conference on Pattern recognition, 1994, pp. 126–131.
7. J. S. Bae and T. L. Song, “Image tracking algorithm using template matching and PSNF-m,” International Journal of Control, Automation, and Systems, vol. 6, no. 3, pp. 413–423, June 2008.
8. J. Hsieh, W. Hu, C. Chang, and Y. Chen, “Shadow elimination for effective moving object detection by Gaussian shadow modeling,” Image and Vision Computing, vol. 21, no. 3, pp. 505–516, 2003.
9. S. Mantri and D. Bullock, “Analysis of feed forward-back propagation neural networks used in vehicle detection,” Transportation Research Part C– Emerging Technologies, vol. 3, no. 3, pp. 161–174, June 1995.
10. T. Abdullah, A. Anjum, M. Tariq, Y. Baltaci, and N. Antonopoulos,” Traffic monitoring using video analytics in clouds,” in 7th IEEE/ACM International Conference on Utility and Cloud Computing (UCC), 2014,pp. 39–48.
11. K. F. Mac Dorman, H. Nobuta, S. Koizumi, and H. Ishiguro, “Memory based attention control for activity recognition at a subway station,” IEEE Multimedia, vol. 14, no. 2, pp. 38–49, April 2007.
12. C. Stauffer and W. E. L. Grim son, “Learning patterns of activity using real-time tracking,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 8, pp. 747–757, August 2000.
13. C. Stauffer and W. Grim son, “Adaptive background mixture models for real-time tracking,” in IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1999, pp. 246–252.
14. P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in IEEE Conference on Computer Vision and Pattern Recognition, 2001, pp. 511–518.
15. [15] Y. Lin, F. Lv, S. Zhu, M. Yang, T. Cour, K. Yu, L. Cao, and T. Huang,“ Large-scale image classification: Fast feature extraction and svm training,” in IEEE Conference on Computer Vision and Pattern Recognition(CVPR), 2011.
16. V. Nikam and B. B. Meshram, “Parallel and scalable rules based classifier using map-reduce paradigm on hadoop cloud,” International Journal of Advanced Technology in Engineering and Science, vol. 02, no. 08, pp.558–568, 2014.
17. R. E. Schapire and Y. Singer, “Improved boosting algorithms using confidence-rated predictions,” Machine Learning, vol. 37, no. 3, pp. 297– 336, December 1999.
18. K. Shvachko, H. Kuang, S. Radia, and R. Chansler, “The hadoop distributed file system,” in 26th IEEE Symposium on Mass Storage Systems and Technologies (MSST), 2010.
19. A. Ishii and T. Suzumura, “Elastic stream computing with clouds,” in4th IEEE Intl. Conference on Cloud Computing, 2011, pp. 195–202.

20. Y. Wu, C. Wu, B. Li, X. Qiu, and F. Lau, "Cloud Media: When cloud on demand meets video on demand," in 31st International Conference on Distributed Computing Systems, 2011, pp. 268–277.

21. J. Feng, P. Wen, J. Liu, and H. Li, "Elastic Stream Cloud (ESC): A stream-oriented cloud computing platform for rich internet application," in Intl. Conf. on High Performance Computing and Simulation, 2010.

22. "Vi-system," http://www.agentvi.com/.

23. "Smart CCTV," http://www.smartcctvltd.com/.

24. "Project BESAFE," http://imagelab.ing.unimore.it/besafe/.

25. B. S. System, "IVA 5.60 intelligent video analysis," Bosh Security System, Tech. Rep., 2014.

26. "EPTA Cloud," http://www.eptascape.com/products/eptaCloud.html

27. "Intelligent vision," http://www.intelli-vision.com/products/intelligentvideo-analytics.

28. K.-Y. Liu, T. Zhang, and L. Wang, "A new parallel video understanding and retrieval system," in IEEE International Conference on Multimedia and Expo (ICME), July 2010, pp. 679–684.

29. J. Dean and S. Ghemawat, "Map reduce: Simplified data processing on large clusters," Communications of the ACM, vol. 51, no. 1, pp. 107–113, January 2008.

30. T. Wigand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," IEEE Trans. on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 560–576, July 2003.

31. H. Schulzrinne, A. Rao, and R. Lanphier, "Real time streaming protocol (RTSP)," Internet RFC 2326, April 1996.

32. H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A transport protocol for real-time applications," Internet RFC 3550, 2203.

33. "Open CV," http://opencv.org/.

34. "Open Stack icehouse," http://www.openstack.org/software/icehouse/.

35. J. Nickolls, I. Buck, M. Garland, and K. Skadron, "Scalable parallel programming with CUDA," Queue-GPU Computing, vol. 16, no. 2, pp.40–53, April 2008.

36. J. Sanders and E. Kandrot, CUDA by Example: An Introduction to General-Purpose GPU Programming, 1st ed. Addison-Wesley Professional, 2010.

37. http://cogcomp.cs.illinois.edu/Data/Car/.

38. http://www.itl.nist.gov/iad/humanid/feret/.

39. http://www.nvidia.com/object/gcr-energy-efficiency.html.

40. M. Zaharia, M. Chowdhury, M. J. Franklin, S. Schenker, and I. Stoica, "Spark: Cluster computing with working sets," in 2nd USENIX conference on Hot topics in cloud computing, 2010.