International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

Adaptive Resources Management for Analyzing Video Streams from Globally Distributed Network Cameras

Dr.P.Sri Vani[1], M.Maanvi Reddy[2], M.Meghana[3], P.Sowmya[4]
[1]Associate Professor, Department of CSE, Malla Reddy Engineering College for Women, Hyderabad, Telangana, India.
[2,3,4]UG-Students, Department of CSE, Malla Reddy Engineering College for Women, Hyderabad, Telangana, India.

ABSTRACT

There has been tremendous growth in the amount of visual data available on the Internet in recent years. One type of visual data of particular interest is produced by network cameras providing real-time views. Millions of network cameras around the world continuously stream data to viewers connected to the Internet. This data may be used by a wide variety of applications such as enhancing public safety, urban planning, emergency response, and traffic management which are computationally intensive. Analyzing this data requires significant amounts of computational resources. Cloud computing can be a preferred solution for meeting the resource requirements for analyzing these data. There are many options when selecting cloud instances (amounts of memory, number of cores, locations, etc.). Inefficient provisioning of cloud resources may become costly in pay-per-use cloud computing. This paper presents a method to select cloud instances in order to meet the performance requirements for visual data analysis at a lower cost. We measure the frame rates when analyzing the data using different computer vision methods and model the relationships between frame rates and resource utilizations. We formulate the problem of managing cloud resources as a Variable Size Bin Packing Problem and use a heuristic solution. Experiments using Amazon EC2 validate the model and demonstrate that the proposed solution can reduce the cost up to 62% while meeting the performance requirements. Index Terms—cloud computing, video streaming, MJPEG, resource management, cost reduction **.**

INDEX TERMS: Deep neural network, electricity theft, machine learning, minimum redundancy maximum relevance, principal component analysis, smart grids.

INTRODUCTION

THE use of visual data such as images and videos for scientific analysis to solve real-world problems has been increasing significantly over the past decade. Network cameras are of particular interest as they generate continuous real-time video data with rich and versatile content [31]. Millions of network cameras are deployed every year [24]. The video analysis market is rapidly growing and is estimated to be worth more than $1.2 billion by the year 2017 [6]. A wide variety of applications such as improving public safety [21], aiding emergency response [14], and surveillance [28] may use the large volumes of visual data. The network cameras considered in this paper consist of both indoor and outdoor cameras including traffic cameras, cameras inside shopping malls, and other institutions. These are public cameras providing free access and owned by different organizations. Modifying any configuration settings of the network cameras is impossible.

These applications may require (1) high resolution video data, (2) analysis for long durations, (3) data from multiple cameras, and (4) high frame rates. These requirements represent “big data” problems that require analysis of large amounts of visual data which needs substantial amounts of computational resources. Cloud computing has the potential to meet these resource needs by selecting many cloud instances (i.e., virtual machines, VMs) containing more cores and large memory. Many applications require streaming data from network cameras around the world, for example, studying the traffic pattern of cities, analyzing global fashion trends, and monitoring weather conditions at different regions. The distance between the network camera and cloud instance can affect the performance of the analysis. Hence there is a need to efficiently stream the data from multiple sources at different geographical locations. Cloud vendors offer many types of VM instances: with different number of cores, memory capacities, and geographical locations. The “pay-per-use” pricing model encourages the use of only a few cloud instances with small number of cores and less memory. The computational requirements of the applications may vary depending on the time of the day and the content of the scene being analyzed. The different competing factors mentioned above make resource management a challenging problem. Few studies have been devoted to selecting the most efficient cloud instances to analyze many video streams at low monetary costs. These studies do not consider the effects of the different types and locations of the instances on the overall cost and performance of the analysis.

This paper presents a method called Adaptive Resource Management for Video Analysis in Cloud (**ARMVAC**). ARMVAC determines the configurations (types, locations, and numbers) of cloud instances needed to meet the performance requirements at low costs. We consider Motion JPEG (MJPEG) [4] as the format of video data because most network cameras support MJPEG streaming (some newer network cameras also support H.264). ARMVAC considers the network distances (measured by the round-trip time, RTT) between cameras and cloud instances. ARMVAC models the relationships of the frame rates and CPU utilization on different types of cloud instances.We develop our model through the analysis of three different computer vision methods provided by the OpenCV library [7]: People Detection, Edge Detection, and Color Histogram. We model the problem of selecting cloud instances at low costs as a Variable Size Bin Packing Problem (VSBPP) and use a heuristic algorithm [12] to find a solution. ARMVAC predicts the maximum number of streams from cameras to be analyzed on different cloud instances for the given analysis programs. Our solution can dynamically adapt to the varying resource requirements of analysis programs running for long durations. ARMVAC monitors the utilization of the cloud resources at regular intervals and automatically scales the number of resources based on the utilization and performance requirements. ARMVAC is evaluated using Amazon Elastic Compute Cloud (EC2) instances [2]. Three analysis programs: Motion Estimation, Face Detection, and SIFT feature extraction are used for evaluation. The method can achieve the required frame rates and save up to 62% cost compared with four other cloud resource selection strategies.

This paper makes the following contributions: (1) It is one of the first papers devoted to selecting the cloud

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362
IJARST

configurations for analyzing large (GB) amounts of data from multiple video streams. The sources of the streams are globally distributed. (2) Our method considers both performance requirements and costs, modelling this problem as a bin packing problem and using a heuristic solution. (3) The paper presents a prediction model based on CPU utilization for determining the number of streams that can be analyzed on different types of cloud instances for a given analysis program. (4) We evaluate the solution using Amazon EC2 and demonstrate up to 62% cost reduction compared with four other strategies for selecting cloud instances.

LITERATURE SURVEY:

Years	Title	Methodology	Research Proposal	Algorithm
2012	Resource Allocation for Service Composition in Cloud-based Video Surveillance Platform	Effective VM resource allocation model for composition of video surveillance services in cloud computing platform	video streams are delivered to the cloud from video capturing services.	VM resources,VSS services
2014	A System for Large-Scale Analysis of Distributed Cameras	For executing the same submitted analysis method for all the selected cameras and providing the aggregated results as well.	The system provides an API that makes it easy to migrate existing analysis	RSA
2015	Adaptive Cloud Resource Allocation for Analysing Many Video Streams	Resource management strategies in this case mainly focus on the threads management and distribution of threads.	The goal of resource management is to maintain the required performance while making the cost as low as possible	ECC
2016	Location Based Cloud Resource Management for Analysing Real-Time Videos from Globally Distributed Network Cameras	Dealing with the location selection of cloud instances and the placement of data and applications in distributed cloud networks.	Determine the instance locations with the least cost, we define the problem of selecting locations and identify the factors influencing the cost.	DES
2017	Network cameras video analytical market	Revolves around application,function,system architecture,vertical and geography	Centralized or server based analytics and distributed or edge based analytics	AES

Existing System:

There has been tremendous growth in the amount of visual data available on the Internet in recent years. One type of visual data of particular interest is produced by network cameras providing real-time views. Millions of network cameras around the world continuously stream data to viewers connected to the Internet. This data may be used by a wide variety of applications such as enhancing public safety, urban planning, emergency response, and traffic management which are computationally intensive. Analyzing this data requires significant amounts of computational resources. Cloud computing can be a preferred solution for meeting the resource requirements for analyzing these data. There are many options when selecting cloud instances (amounts of memory, number of cores, locations, etc.). Inefficient provisioning of cloud resources may become costly in pay-per-use cloud computing.

Proposed System:

This paper presents a method to select cloud instances in order to meet the performance requirements for visual data analysis at a lower cost. We measure the frame rates when analyzing the data using different computer vision methods and model the relationships between frame rates and resource utilizations. We formulate the problem of managing cloud resources as a Variable Size Bin Packing Problem and use a heuristic solution. Experiments using Amazon EC2 validate the model and demonstrate that the proposed solution can reduce the cost up to 62% while meeting the performance requirements.

INPUT DESIGN AND OUTPUT DESIGN

INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

OBJECTIVES

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

RELATED WORK:

Several papers discuss the issues related to streaming media content residing on the cloud for uses such as Video on Demand (VoD). For example, Cloudmedia [34], optimizes the cloud resources based on the server utilization and network bandwidth. Wang et al. [33] discuss data streaming with optimizing cloud resources based on cost and availability. They consider the locations of data sources. They focus only on media delivery and hence do not consider utilization and performance issues while streaming and analyzing video data flowing into the cloud. These studies focus on streaming data out from the cloud instances. This paper solves a different problem: streaming video data into cloud instances for analysis. The data come from globally distributed network cameras.

2.1 Prediction Based Resource Management Plenty of research work exists on developing prediction models for cloud resource provisioning. Islam et al. [17] develop statistical models to predict the surge in resource requirements. Khan et al. [20] use a multiple time series to predict workload variations in a cluster of virtual machines. Xiao et al. [35] present a system which dynamically allocates resources based on the past behavior of the virtual machines. Van et al. [27] propose an autonomic resource manager for service hosting platforms using a constraint based programming approach. Most of these studies use past information to predict future resource usage and perform resource management. Each analysis program has a unique resource usage pattern. Obtaining the past resource usage information for all analysis programs is not feasible. Our solution does not use past information and instead performs resource management based on the current resource usage. Our method allocates resources and adaptively adjusts to the change in resource consumption.

2.2 Dynamic Resource Provisioning Methods Many studies investigate

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

resource allocation on the cloud for a variety of applications. Vijaykumar et al. [32] consider dynamic resource provisioning for data streaming. They do not consider analyzing video data from network cameras. In contrast, this paper considers how to analyze video streams from network cameras.Sharma et al. [29] use an integer linear programming formulation to select cloud configurations. Their model handles tasks with small uniform data sizes and our solution can handle tasks of different sizes. Hossain et al. [16] propose a heuristic resource management for cloud based video surveillance systems. Song et al. [30] propose a queuing based resource management for multimedia applications. The main objective for both methods is to reduce the service wait time and long term service cost to satisfy the QoS requirements. They reduce the monetary cost by reducing the running time of the tasks. Our solution reduces the overall cost by selecting costefficient cloud instance types and locations. We model the performance in terms of frame rate and not the running time as the tasks we consider may run for any duration. Hossain et al. [15] discuss resource allocation for service composition in cloud based video surveillance applications. They minimize the number of cloud instances used for service composition by modelling the problem as a multidimensional bin packing problem. Their method does not reduce the overall cost. Our solution considers multiple analysis programs of different characteristics analyzing data at different frame rates. The solution by Hossain et al. [15] is evaluated by simulation, but our method is evaluated using a commercial cloud vendor—Amazon EC2.

2.3 Amazon Auto Scaling Amazon EC2

provides auto scaling [1] to add or remove cloud instances based on utilization. Auto scaling lets users select a group of instances and scale them up or down based on use. It does not consider the overall cost. Auto scaling is tied to a particular region and cannot handle the effect of network delay on the input data as explained in Section 3.3.

TABLE 1: COMPARISON OF ARMVAC WITH RELATED WORK

	Data Direction with the Cloud Instance	Selects Different Types of Instances	Selects Instances at Different Locations	Evaluation Method	Use Globally Distributed Data Sources	Reduces Overall Cost
Auto Scaling [1]	N/A	No	No	N/A	N/A	No
Cloudmedia [34]	OUT	No	No	Simulation	No	Yes
Wang et al. [33]	OUT	No	Yes	Simulation	Yes	Yes
Xiao et al. [35]	N/A	No	No	Simulation	N/A	No
Van et al. [27]	N/A	No	No	Simulation	N/A	No
Vijaykumar et al. [32]	IN	No	No	Xen Virtual Environment	Yes	No
Hossain et al. [15]	IN	No	No	Simulation	Yes	No
ARMVAC	**IN**	**Yes**	**Yes**	**Amazon EC2**	**Yes**	**Yes**

TABLE 2: COMPARISON OF ARMVAC WITH OUR PREVIOUS WORK

	Adaptive Resource Manager	Selects Different Types of Instances	Selects Instances at Different Locations	Cost Optimization Method	Speedup Resource Allocation by Predicting the Maximum Number of Streams that can be Analyzed on Instances
Kaseb et al. [19]	Yes	No	No	Heuristic	No
Chen et al. [9]	Yes	No	No	Heuristic	No
ARMVAC	**Yes**	**Yes**	**Yes**	**VSBPP [13]**	**Yes**

Our method determines the types and locations of instances best suited for the given application such that the overall cost is low.

2.4 Improvements

from Our Previous Work This paper extends our previous work [10, 18, 31] that builds a software infrastructure using cloud computing to analyze visual data from thousands of network cameras. The system is referred to as Continuous Analysis of Many CAMeras (CAM2). This work adds adaptive resource management of cloud instances for analyzing video streams. This paper extends our work [9] which gradually adds resources based on utilization and our work [19] on resource management by finding cost effective instances. This paper is an improvement over our work [25] which primarily focuses on selecting cloud instance locations. This paper implements a resource manager which considers the location of the cloud instances and cameras. The resource manager predicts the number of streams that can be analyzed using a given program on different types of cloud instances to perform resource allocation faster. The resource manager selects cloud instances with different types and locations at lower cost. Table

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
IJARST
ISSN: 2457-0362

1 compares ARMVAC with related work. Table 2 compares ARMVAC with our previous work. Readers are encouraged to visit the CAM2 website at https://cam2.ecn. purdue.edu/ and register to explore this system.

2.5 Variable Size Bin Packing Problem ARMVAC

models the problem of selecting cloud instances at low cost as a Variable Size Bin Packing Problem (VSBPP) [13]. It considers a collection of bins of finite sizes. The number of bins of each size is unlimited. Each bin has a cost associated proportional to its size. The objective of VSBPP is to pack a list of items into bins so as to minimize the total cost. This is an NP-Hard problem [13]. There are different heuristic algorithms to solve VSBPP that includes First Fit algorithm, Best Fit algorithm, and Best Fit Decreasing algorithm. [23]. This paper considers a variant of the Adapted-Best First Decreasing (A-BFD) algorithm [12] as it is reported to perform better than the other methods. A-BFD sorts the bins according to the ratio of their costs and sizes. The best bin is the one with the least cost-size ratio. We consider the cloud instances to be equivalent to the bins in a VSBPP and the number of streams that can be analyzed on an instance to be equivalent to the size of the bin.

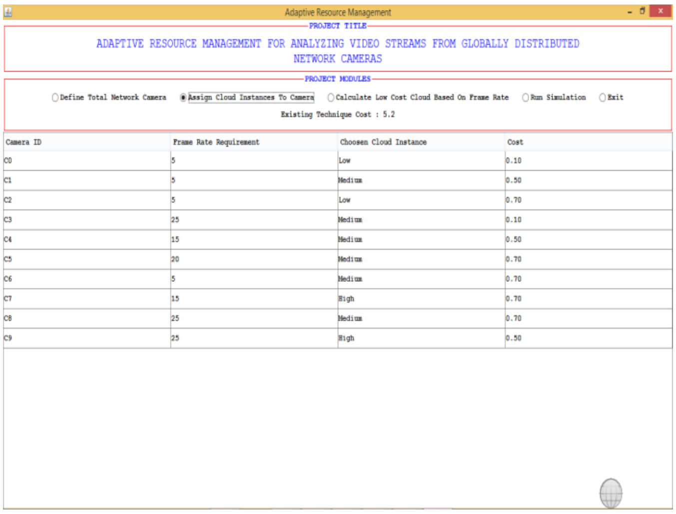
RESULTS:

SCREENSHOTS:

In above screen click on 'Define Total Network Camera' radio button to enter number of cameras for this project simulation

In above screen i am giving number of cameras as 10. Now click ok to get below screen

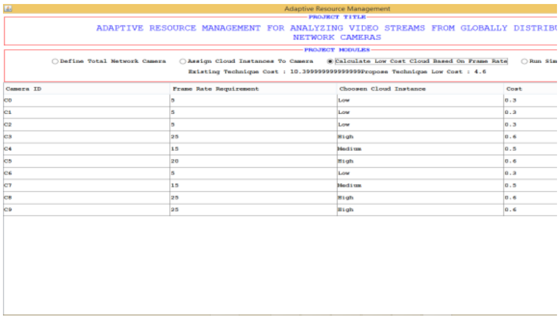
In above screen total 10 cameras are created from 0 to 9 but they don't have any assigned cloud and frame rate requirement or cloud cost. Click on 'Assign Cloud Instances To Camera' radio button to assign this cameras to cloud instances



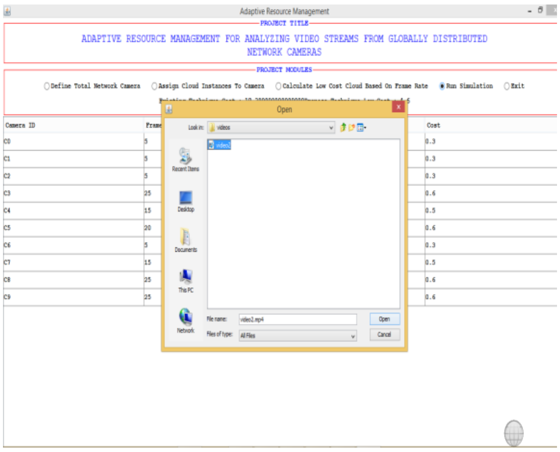
In above screen base on Frame requirements cloud instances and cost assign to each camera. Now to assign lowest cost cloud to each camera base on frame requirement and closeness of cloud click on 'Calculate Low Cost

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

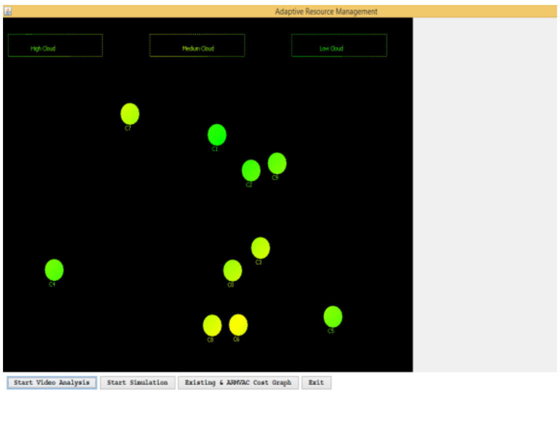
Cloud Base on Frame Rate' button to get below screen. In above screen we can see with existing technique for each hour cost is 5.2



In above screen we allocate low cost cloud to camera and with propose technique we can see per hour required cost is 4.6$. So propose ARMVAC technique charge less compare to all existing technique. Now click on 'Run Simulation' button to upload video and to see simulation



In above screen I uploading one video and after video upload will get below screen

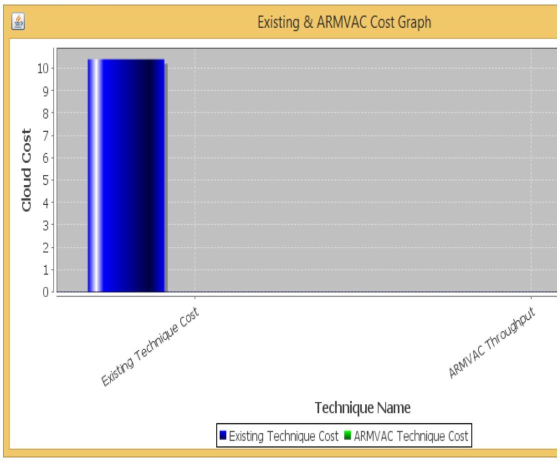


In above screen all circles represents cameras and click on 'Start Video Analysis' to play video and then click on 'Start Simulation' button to send that video frame to cloud instances. All rectangles represents as cloud instances.

In above screen we can see video is playing and in button we can see person detected at 46th frame. Now click on 'Start Simulation' button to start sending frames to cloud

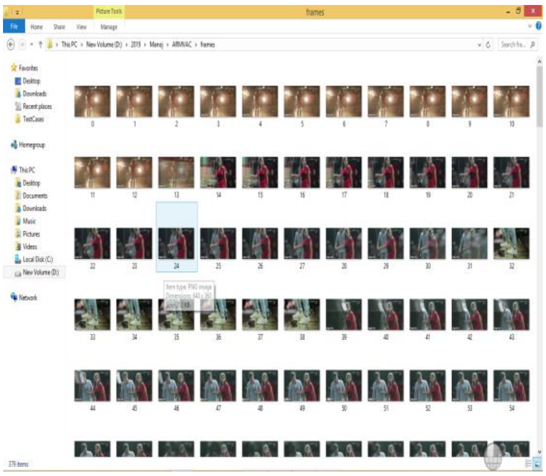
In above screen we can see camera is sending frame to nearest cloud instance. Now click on 'Existing & ARMVAC Cost Graph' button to view cost details in graph

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal www.ijarst.in
ISSN: 2457-0362



In above graph x-axis represents technique name and y-axis represents cost taken by that technique and we can see ARMVAC took less cost.

All generated frames from video can be seen inside frames folder. See below screen



So by choosing nearest cloud instances with less cost we can adaptively use and manage resources of cloud

CONCLUSION:

This paper presents ARMVAC, an adaptive resource manager to select low-cost cloud instances for analyzing MJPEG data from globally distributed network cameras. Inputs to ARMVAC are the analysis programs, the required number of cameras, the locations of the cameras, the target frame rates, and the durations of the analyses. The outputs are the types, locations, and number of cloud instances to be launched to achieve the target frame rate on all the cameras. ARMVAC includes a model to predict the maximum number of streams that can be analyzed on different types of instances. We evaluate ARMVAC using Amazon EC2 cloud instances and observe that the achieved frame rate on all cameras is equal to the target frame rate for different input scenarios thereby satisfying the performance requirements. We observe that ARMVAC lowers the overall cost up to 62% when compared with four other reasonable strategies (ST1 - ST4) for selecting cloud configurations. Our evaluation demonstrates that our method is not ad-hoc and can be applied to different analysis programs.

As part of our future work, we will extend the method to handle analysis programs which are memory intensive, bandwidth intensive, or I/O intensive. We would also like to improve our method of adaptively launching instances while adjusting to the run-time conditions. We also plan to study effect of adaptive nature of H.264 streams on resource selection.

FUTURE WORK:

As part of our future work, we will extend the method to handle analysis programs which are memory intensive, bandwidth intensive, or I/O intensive. We would also like to improve our method of adaptively launching instances while adjusting to the run-time conditions. We also plan to study effect of adaptive nature of H.264 streams on resource selection.

REFERENCES:

[1] Amazon Auto Scaling. https://aws.amazon.com/ autoscaling/.

International Journal For Advanced Research In Science & Technology
A peer reviewed international journal
www.ijarst.in
ISSN: 2457-0362

[2] Amazon EC2. http://aws.amazon.com/ec2/..
[3] Amazon EC2 Pricing. http://aws.amazon.com/ec2/ pricing/.
[4] Motion JPEG Video Codec. http://www. digitalpreservation.gov/formats/fdd/fdd000063.shtml.
[5] netem, Network Emulation Utility. http://www. linuxfoundation.org/collaborate/workgroups/ networking/netem.
[6] Network camera and video analytics market. http://www.marketsandmarkets.com/Market-Reports/ visual-communication-market-775.html, 2011.
[7] G. Bradski. The OpenCV library. Dr. Dobb's Journal of Software Tools, 25(11):120, 122–125, 2000.
[8] J. Canny. A computational approach to edge detection. Pattern Analysis and Machine Intelligence, IEEE Transactions on, (6):679–698, 1986.
[9] W. Chen et al. Adaptive cloud resource allocation for analysing many video streams. In IEEE International Conference on Cloud Computing Technology and Science, 2015.
[9] W. Chen et al. Analysis of large-scale distributed cameras using the cloud. Cloud Computing, IEEE, 2(5):54–62, 2015.
[11] I. I. CPLEX. V12. 1: Users manual for cplex. International Business Machines Corporation, 46(53):157, 2009.
[12] Crainic et al. Efficient lower bounds and heuristics for the variable cost and size bin packing problem. Computers & Operations Research, 38(11):1474–1482, 2011.