

# **A Framework for Cloud Assisted Adaptive Video Streaming to Enhance User QoE**

**G. Rajasekaran**

Assistant Professor,

Department of Computer Science and Engineering,  
Vivekanandha College of Engineering for Women  
Namakkal, Tamilnadu, India

**K. Keerthiga**

Assistant Professor,

Department of Information Technology,  
Vivekanandha College of Technology for Women,  
Namakkal, Tamilnadu, India

## **ABSTRACT**

The drastic increase in the usage of handheld devices like smartphones, tablets, etc., produce huge traffic on the Internet which are mainly due the video streaming services applications. Due to the limitations of resources (Power, Memory, Processing, etc.), numerous design patterns of the mobile devices and heterogeneous users' expectations, it is difficult to handle those video streaming services requests. In order to provide an expected experience to the end user and to reduce the burden of the service providers, cloud computing provides the most easy and viable solution. A novel framework in cloud environment has been proposed in this paper to provide a good trade-off between the service handling as well as the user experience. The framework covers various aspects related to streaming services such as user experience, content distribution and retrieval, power usage, network analysis, video storage, device analysis on static and runtime conditions, parallel service provision and the experience perceived by user.

## **General Terms**

Cloud Based Video Streaming Services

## **Keywords**

Cloud Computing, Video Streaming, Quality of Experience (QoE), Transcoding, Cloudlet.

## **1. INTRODUCTION**

Cloud Computing is a finite, scalable and resource rich environment that makes software, platform and infrastructure services available to the end users virtually using the Internet. The Cloud Computing environment consists of a huge set of resources like storage, processors, networking components, databases, platforms and the software both on application/system levels. Due to this capability of cloud environment, it can handle any type of service request that requires resources for any applications.

Multimedia Streaming Services (MSS), particularly video streaming is one of the services on the Internet that consumes huge amount of resources to produce the desired results. As these applications are delay sensitive, it accepts certain level of compression for storage and transmission over the Internet. It requires massive amounts of storage space for video contents, huge number of processor cores with threads to transcode and compress the video files, and a high bandwidth for transmission over the Internet.

Video Streaming Services will normally produce expected results, under the controlled environments with stable network connectivity, but fails to deliver the exact results for the end

users under uncontrolled environments with dynamic connectivity. The Quality of Experience (QoE) perceived by the user is also analyzed in a controlled environment with the people expertise in the quality analysis. Now-a-days handheld devices, particularly smart mobile devices have come under the above category to deliver video streaming services.

Presently smartphones have become the mandatory gadgets of all peoples, despite age, profession and usage. According to the survey, more than 60 percentages of the people are using smartphones worldwide with different platforms which are responsible for most of the Internet traffic. A survey of CISCO states that the video applications usage using smartphones is responsible for more than 66% of the total Internet traffic [1]. Due to the easy and cost effective availability of smart devices, its ease of use, requests of people for various types of services through these mobile devices, huge amount of resources are demanded. Video Streaming is one of the services that incur heavy bandwidth, storage, process and a higher level of compatibility. But mobile devices are always limited in resources due to its compactness, power source (limited battery capacity), low processing speed, limited memory and network fluctuations due to the mobility of the user.

In order to compensate for the above limitations in mobile devices, researchers suggested the idea called computation offloading. In these, requests received from the users were submitted to the cloud environment and the processed results were given to the mobile devices of the end user. The actual computation, storage and other resources were all handled by the cloud environment and only the final results were given to the end user. This is one of the finest methods to preserve energy and other resources on user side.

Though the offloading method is efficient and the limitations of mobile devices were compensated by cloudlets, the Multimedia Service Providers (MSP) face challenges in delivering good Quality of Service (QoS) to the end users. If the QoS suffers, then the Quality of Experience felt by the end user will also deteriorate. Hence the service provider must consider multiple aspects like bit rate, throughput, transmission delay, availability, jitter, etc. as the parameters of performance to provide a good Quality of Service. QoE is the experience perceived by the end user which depends on the technical parameters of QoS and other non-technical parameters pertaining to the end user.

Hence a framework to provide a video streaming service with good QoE on a smart phone using cloud computing is the need of the day. A novel framework using Cloud Computing

Architecture has been designed to provide good QoE for mobile video streaming service on a smartphone.

## 2. RELATED WORK

Many researchers have discussed about Bandwidth and Device Monitoring, Transcoding of video contents, QoE Analysis, Caching mechanism, P2P technology and Tracking methodology. Armbrust et al [2] discussed about a detailed view of cloud computing and the resources available on cloud environments. He also discussed the impact of the cloud computing on business, research, social and educational side. Viktor Mauch et al [3] analyzed the power of cloud resources and future technologies with the broad comparison of existing technologies and limitations. Y.K. Lai et al [4] had done the detailed analysis on mobile device power consumption by various hardware and software utilities and prediction methods for power usage statistics. J.M.Kang et al [5] analyzed the battery lifetime of the mobile devices with the usage pattern of the user, and analyzed how the hardware and software resources consume power resources for the particular usage pattern. Fangming Liu et al [6], N.Davies [7], S.Kosta et al [8] are all discussed how the limitations of the mobile devices are overcome by the cloud resources using job offloading and challenges during offloading methods. Also they analyze the performance improvement of the processes by using cloud computing resources. B.Aggarwal et al [9] proposed the framework for energy in mobile communication by using cloud environment. Seungjun Yang et al [10] analyzed the transfer and cost minimization of the offloading methodology by using cloud. [11]-[16] discussed and analyzed the application performances on computation, storage and energy based analysis by using cloud computing environment.

W.Zhu et al [17] discussed the how cloud resources can be used for multimedia operations. S. Heiko et al [18] and W.Mathias et al [19] gave a detailed introduction about the

Scalable and Adaptive video coding technologies and working principles. [20]-[22] proposed a model for cloud based video Transcoding services for mobile devices. L.Yao et al [24] measured a server side workload for mobile streaming service provision. The survey covers the Transcoding of the video contents, audio video formats, device heterogeneity and hardware, software variations and practical limitations on service provision. W. Yu et al [25] proposed a model called CloudMOV a highly interactive mobile TV environment with social interactivity. W.Xiaofei et al on [26] and [27] introduced a framework for video streaming works under the principle of caching and social interactivity also called popularity based content caching mechanism. They proposed a burst transmission mechanism for the content injection with the user device by analysing the users network conditions. X.Jin et al [28] and X.Ke et al [29] discussed the peer assisted streaming services using network and proxy caching method. S.Kalpna et al [30] and Chikkerur et al [31] discussed the subjective and objective video quality assessment models. The accuracy of those models and practical limitations are also discussed with the performance comparison. In [32] and [33] the impact of QoS on the user experience by QoS/QoE mapping model and QoE driven adaptive streaming services were discussed.

## 3. VIDEO STREAMING FRAMEWORK USING CLOUD

Video streaming is one of the multimedia delivery models. The streaming can broadly differentiated into two categories, live video streaming and stored video streaming. The video file was separated into various chunks on the server side, these chunks are delivered to the end user based on their network quality. Before delivering the video contents the raw video and audio contents were compressed with various encoders.

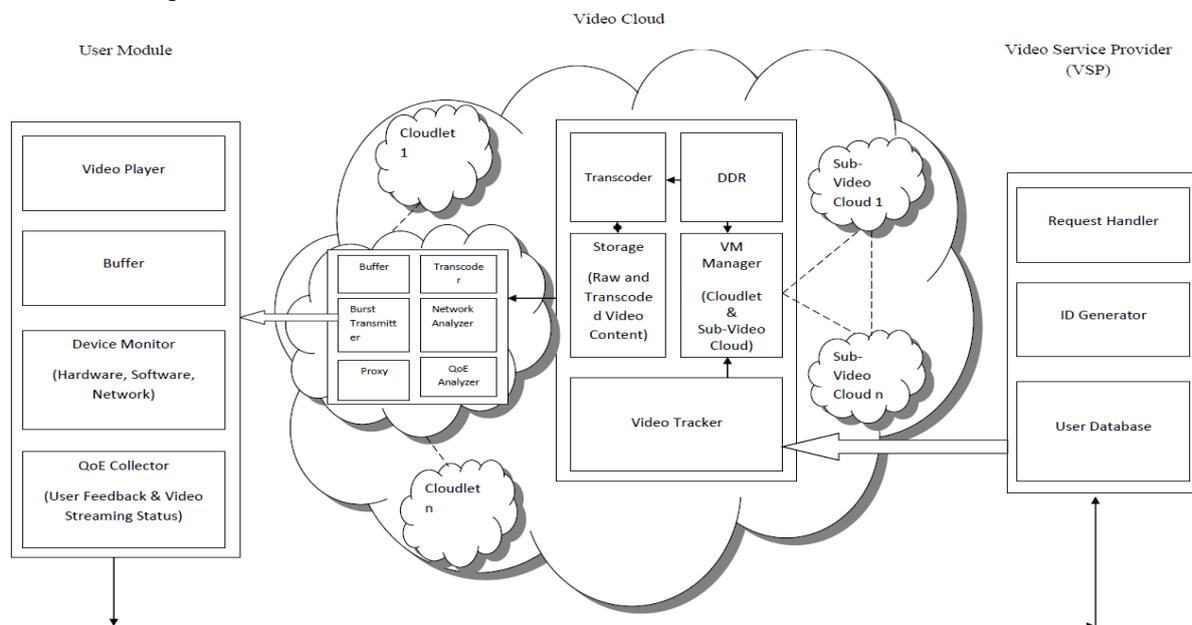


Fig 1: Video Streaming Framework

The compression was applied separately on audio and video files. Then these compressed files are packed with various containers mp4, flv, matroska (.mkv), etc.,. The user module decode the received video/audio contents and play the

subsequent file using video players, embedded players in the browser or the applications capable to play the video file.

A novel framework to stream the requested video contents to the user device that has been proposed was shown in the Figure.1. It includes three main modules namely User Module (UM), Video Service Provider (VSP) Module and Video Service Cloud (VSC) Module.

The video streaming service starts with the request from the user module which is given to the Video Service Provider (VSP). The VSP hands over the process to Video Service Cloud (VSC) to allocate a separate virtual machine module called cloudlet to perform the requested operation on behalf of the user module. Video Service Cloud takes the burden of the user module and it acts as a surrogate for user device. The requested video file was processed by the central cloud and cloudlet inside the VSC by using the parameters taken from the user module via the HTTP request. Finally the video streaming was started from the cloudlet to user module.

### 3.1 User Module

The User Module (UM) is a handheld device of the user with the combination of software, hardware and network capabilities. It ranges from the low end mobile devices to high end laptops. The user module uses a video player or video plug-in enabled web browser with HTML5 support or any application capable to play the video file. The buffer module is to store the incoming video packets during streaming process. The device status monitor collects the runtime information about the hardware, software and network status of the device. The information includes power consumption rate, power source (Battery or Direct power source), processor status (amount of processor consumed by the applications), storage (% of free memory on the buffer), sensor status, location and status of the network used by device (2G, 3G, LTE, WIFI). When the user requests process the required parametric values are invoked from the Device Monitor to build HTTP Request Message.

The QoE Collector module is an experience monitor which collects the experience of the user. The QoE collector collects the information from the delivered video contents such as Initial Buffering Time (IBT), Post Buffering Time (PBT), and Video Discontinuity (VD). It also collects the information about the location and network conditions from Device Monitor and submits it to the QoE Analyzer in Cloudlet.

The information about the user and their device is collected from the User Module (UM) in two ways. First time during the registration process and the second time during the time of video request. The information collected during the registration is mostly static values given by the user, Device Monitor is not involved in this process. The information along with the video request is dynamic in nature which is collected from the Device Status Monitor.

The static information given by the user during registration process can be categorized into personal details, career details, personal interests, language, location, device model expected services and control, video player placement are collected during the registration process. The above detail comes under the category of static information related to the user and service.

The video request launched from the user device stuffed with the following parameters in HTTP POST Request. It contains the current location of the user using GPS, network type (2G,3G,LTE,WIFI), power information about the device, sensor details, connected peripherals, device model and process and memory occupation on the hardware devices like

Processor, buffer etc. The buffer module in the UM collects the video chunks provided by the cloudlet and supplies it to the video playback software. The playback device may be a Video Player, the browser or any application with video playback capability.

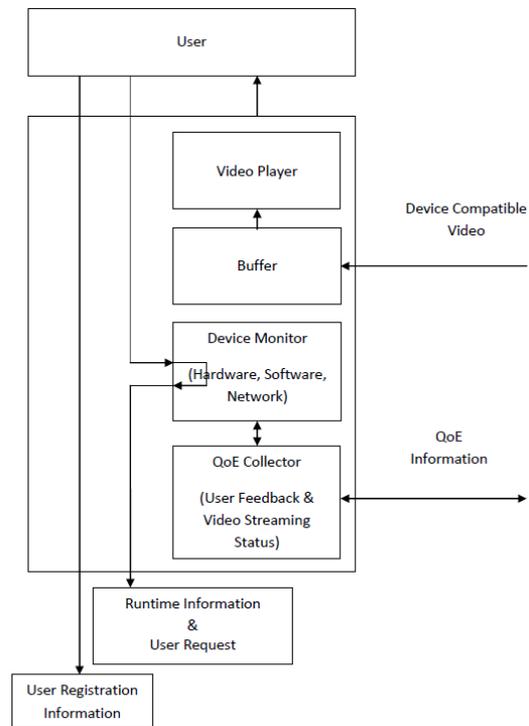


Fig 2: User Module

### 3.2 Video Service Provider

The Video Service Provider (VSP) collects the request given by the User via the registration process and runtime HTTP request URL. After getting the information, Video Service Provider generates identification String (ID) to every user. The ID was stored in the server database and the replica was given to the Video Service Cloud (VSC) to be stored it in a tracker DB.

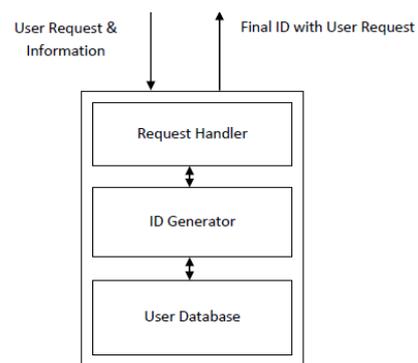


Fig 3: Video Service Provider

### 3.3 Video Service Cloud

Video Service Cloud (VSC) is created to handle the video streaming operations on behalf of the Video Service Provider (VSP) and User Device (UD). In order to escape from the real-time service provisioning problems like dynamic request scaling, single point failure, power and resource (Processing, Storage and Networking) constraints Service Provider handover the request to cloud environment. Due to the

limitations in the mobile devices it also requests assistance from the cloud computing environment particularly here for decoding the video contents. It Transcodes the Video files depending upon the users device model and network conditions. The Virtual Machine Manager (VMM) in VSC is to allocate the new Virtual Machines, depending upon the user request given by the Tracker module. The capacity of the new VM is based on the configuration of the User Device. It allocates two flavors of Virtual Machines (VMs) one is cloudlet and another one is Sub-Video Cloud (SVC). It performs routing operations using tracker module inside the cloud environment to collect the distributed video files in SVC and Cloudlets on various locations.

### 3.4 The Transcoder

The Transcoder performs an operation like audio/video compression and conversion using high end processor cores. The Transcoding here is carried over by invoking the values from the DDR with the user's device model as a key. Before Transcoding, first collects the required parameters from the user request for the conversion. The collected parameters are categorized into two based on device model and current network type. The device considerations include supporting Audio/Video Bit Rate (BR) and Compression Factor (CF). It requires the hardware parameters like Width in Pixels (wp), Height in Pixels (hp), Diagonal resolution in Pixels (dp) and Pixel Per Inch (PPI). The BR is calculated from the values of height, width, Color Depth (CD) and Frames per Second (FPS). The supported Compression Factor (CF) was applied on the Color Depth of each video frames to minimize the size of the video file. The audio file for the corresponding video was also computed for conversion based on the values from the request. For the audio conversion the sampling rate (Hz), Depth and the number of Channels are taken for consideration in Transcoding. On the network side the Transcoder considers the type of network for conversion because the bit rate was limited within the boundary of each network type.

The Transcoder performs these conversions when the given request for the first time for the particular video content or the request is anonymous. Also if the requested file having very low QoE value and vary rarely requested then it was placed in the central storage.

### 3.5 The DDR

The DDR is the Device Description Repository that contains the specifications of most of the devices in the world. The VUClip server based survey [34] states that there is more than 3465 mobile models with 109 resolutions. The well known DDR's are WURFL, AMD etc. It can be invoked by using specified API's like Apache Device Map which offers API's for several languages. By using the device model, the static parameters like screen size, supporting codecs, available sensors, players, power source and capability, etc, can be retrieved from the DDR for conversions and operations.

### 3.6 The Storage

The Storage here is used to store raw video contents and the compressed video contents with the different format from the Transcoder module. In order to maintain the high availability it maintains the very rarely requested video contents with very low QoE rank. It acts as a permanent storage for the raw video contents.

### 3.7 The Sub-Video Cloud (SVC)

The Sub-Video Cloud is also a virtual machine but it is an enhanced version of the cloudlets, its computation and storage capability is far higher than the cloudlets. It was used to store

the video contents based on the location, QoE Score and Quality of the video file. The sub video cloud contains Transcoded video files. The video file with acceptable quality which was required highly will be placed on SVC and distributed among the SVCs and Cloudlets located in various places. Which contains the metadata of the stored files and it will be shared with the tracker module. The metadata inside the SVC is dynamic in nature because for every video transaction the log was updated frequently.

### 3.8 The Cloudlet

The cloudlet is the virtual machine assigned by the Virtual Machine Manager (VM Manager) based on the user request. The capacity of the cloudlet is chosen based on the device parameters of the user. It performs storage and computation operations on behalf of the user device. When the user login to the service, separate cloudlet was allocated for the user based. The cloudlet communicates with the video tracker to identify the location of the video file and based on the information given by the video tracker, it collects the video files from Sub Video Cloud distributed on various locations. The collected video chunks are arranged on the cloudlet by using the proxy module and placed on its buffer. The Transcoder in the cloudlet analyze the current network condition of the user and transcode the video content on the fly with adaptive streaming support. Then the content was placed again in buffer storage. The network analyzer requests the user module about the network condition at the time of instance. Based on the network condition the burst transmitter creates the video chunks and delivers it to the user. During the chunk separation the burst transmitter considers the network condition of the user, quality of the video file and duration of the video file content. Based on the above three parameters, the size of the chunk or the number of chunks were scaled.

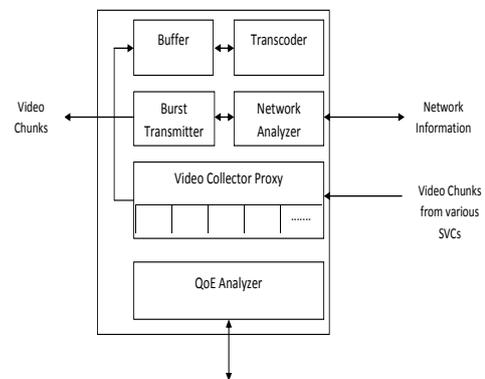


Fig 4: The Cloudlet

During the streaming process the QoE analyzer collects the information from the user device about the distortions happening in the video and alters the cloudlet to correct the QoS parameters like delay, jitter and packet loss. Then modules in the cloudlet once again analyze the video content to enhance the QoS. Finally QoE analyzer collects the feedback from the user about the quality and experience of the delivered video content from QoE collector on user module. Then it provides the QoE rank to the video file based on the location, network condition, user satisfaction and video quality. The QoE ranking was also given the tracker module as a reference for the future usage.

### 3.9 The Tracker Module

The tracker module in the central cloud is the heart of the Video Service Cloud (VSC). It maintains the location of the

sub video clouds and its metadata, cloudlet information, Video service provider information, user details, quality of the video files and QoE rank of the videos in its database. It collects the user request from the service provider and maps it to the database it maintains. After analyzing the database it informs the VM Manager to allocate the separate cloudlet for the user and update the information about the cloudlet in its database.

Once the service gets finished, it analyzes the QoE ranking from the cloudlet. If the QoE ranking and the video request frequency are very low then it was stored in the central DB to improve the availability. When a similar request was received again the same pattern will be suggested to the cloudlets.

### 3.10 Virtual Machine Manager (VMM)

VMM is the Virtual Machine Manager is responsible for allocation, maintenance and de-allocation of Virtual Machines on cloud environment. It receives the instructions from the tracker module and allocates the new Cloudlet and Sub-Video Cloud (SVC). It assigns a new cloudlet with computation, storage and networking capability when users get signed in, based on the preferences like location and device model the cloudlet is allocated for each user. If the user with similar request, device model and nearby to the location of the existing cloudlet then it will be shared by both users. The cloudlet will be deleted by the VM Manager as soon as the user gets signed out from the service. The capacity of the cloudlet VM's is purely based on the device specifications invoked from the Device Description Repository (DDR) on central cloud.

## 4. IMPLEMENTATION

For the prototype implementation Wowza Streaming Engine on server machine has been installed with Ubuntu 14.04 Operating System. It is the cloud based streaming services which transcode the requested video contents into desired formats. During the Transcoding process memory, CPU and other resources are allocated automatically to carry out the conversion process.

The client side machine is installed with Windows 10 Operating system and the client software is the browser with HTML 5 support. We choose chrome browser for client request. During the requisition process the client can choose the amount of resources to transcode the video contents or else it was automatically allocated by the streaming server. After the request launch the video file was converted into various video formats including mobile supported format. The streaming can be performed for any type of device including mobile devices.

The streaming implementation on Linux, Windows and Android mobile platforms. The streaming server converts the given video file and separated into multiple chunks. The chunk size varies from 500 KB to 1 MB depends upon the input video file. These chunks are supplied to the video player during streaming process. Here the separated chunks are tested with the VLC media players, it was played well on those applications too.

Due to the cloud based implementation it provides same output stream for both the low end mobile devices equipped with 1 GB of RAM and 1.5 GHZ Asus ZenFone and the high end Dell Laptop with 16 GB of RAM and 3.0 GHZ Processor despite of client configuration.

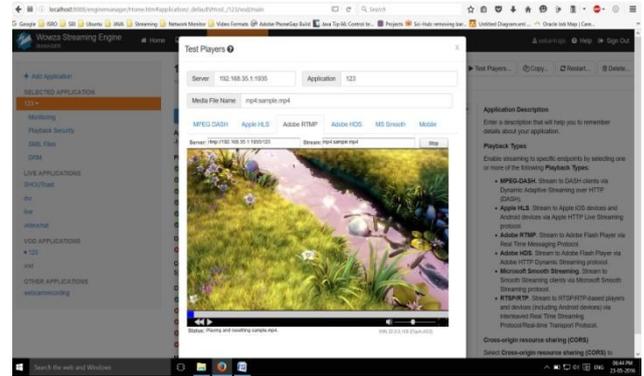


Fig 5: Video Streaming on Wowza

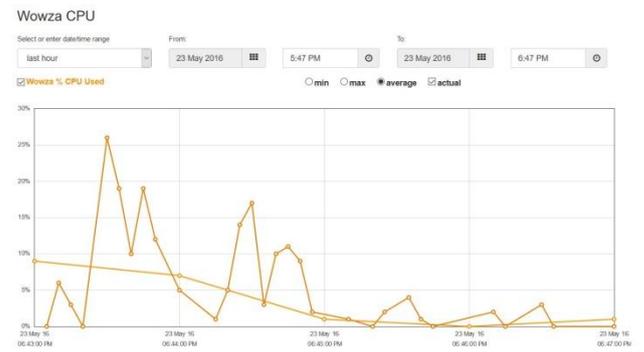


Fig 6: Wowza CPU Load

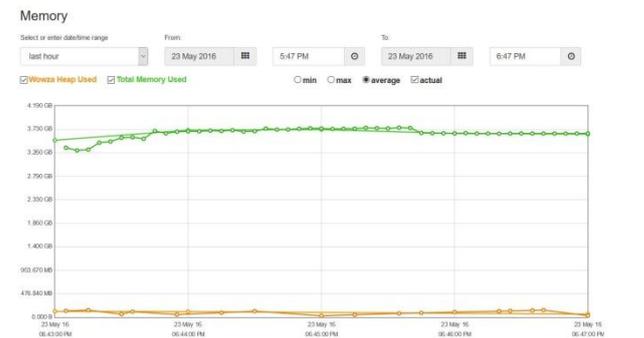


Fig 7: Wowza Memory Allocation

## 5. CONCLUSION

The cloud based streaming engine converts the given video files into different format and effectively streams it to the client machine. The results shows that the cloud based video streaming reduced the workload and memory consumption in server side and also provide high accurate results for heterogeneous users and their devices. The framework mentioned here can be extended to geographical based video placement based on user views can further speed up the streaming process and reduce the delivery time.

## 6. REFERENCES

- [1] "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2011–2016," CISCO, 2012.
- [2] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," Commun. ACM, vol. 53, pp. 50-58, Apr. 2010.
- [3] Viktor Mauch, Marcel Kunze, Marius Hillenbrand, "High Performance cloud computing," Future

- Generation Computer  
System.Elsevier,vol.29,No.6,pp.1408-1416, Aug.2013.
- [4] Y. K. Lai, Y. F. Lai, and P. Y. Chen, "Content-based LCD backlight power reduction with image contrast enhancement using histogram analysis," *J. Display Technol.*, vol. 7, no. 10, pp. 550–555, 2011.
- [5] J. M. Kang, S. S. Seo, and J. W. Hong, "Personalized battery lifetime prediction for mobile devices based on usage patterns," *J. Computing Sci. Eng.*, vol. 5, no. 4, pp. 338–345, 2011.
- [6] Fangming Liu, Peng Shu, Hai Jin, Linjie Ding, Jie Yu, Di Niu, and Bo Li, "Gearing resource-poor mobile devices with powerful clouds: architectures, challenges, and applications," *Wireless Communication.IEEE*, vol.20,No.3,pp.14-22,June.2013.
- [7] N. Davies, "The case for VM-Based Cloudlets in mobile computing," *IEEE Pervasive Computing*, vol. 8, no. 4, pp. 14–23, Oct.-Dec. 2009.
- [8] S. Kosta, A. Aucinas, P. Hui, R. Mortier, and X. Zhang, "Thinkair: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading," in *Proc. IEEE INFOCOM*, 2012.
- [9] B. Aggarwal, N. Spring, and A. Schulman, "Stratus: Energy-efficient mobile communication using cloud support," in *Proc. ACM SIGCOMM*, 2010, pp. 477–478.
- [10] Seungjun Yang, Donghyun Kwon, Hayoon Yi, Yeongpil Cho, Yongin Kwon, Yunheung Paek, "Techniques to Minimize State Transfer Costs for Dynamic Execution Offloading in Mobile Cloud Computing," *Mobile Computing.IEEE Transactions*, vol.13,No.11,pp.2648-2660,Nov.2014.
- [11] X. Zhang, A. Kunjithapatham, S. Jeong, and S. Gibbs, "Towards an elastic application model for augmenting the computing capabilities of mobile devices with cloud computing," *Mobile Netw. Applicat.*, pp.1–15, Apr. 2011.
- [12] Dijiang Huang, Tianyi Xing, and Huijun Wu, "Mobile Cloud Computing Service Models: A User-Centric Approach," *IEEE Network*, vol.27, pp. 6-11, Sep-Oct.2013.
- [13] Yi Xu and Shiwen Mao, "A survey of mobile cloud computing for rich media applications," *Wireless Communications, IEEE*, vol.20,No.3,pp.46-53, June-2013.
- [14] Shaoxuan Wang and Sujit Dey, "Adaptive Mobile Cloud Computing to Enable Rich Mobile Multimedia Applications," *Multimedia.IEEE Transactions*, vol.15,No.4,pp.870-883,June.2013.
- [15] Y. G. Wen, W. W. Zhang, K. Guan, D. Kilper, and H. Y. Luo, "Energy- optimal execution policy for a Cloud-assisted mobile application platform," Sep. 2011.
- [16] Antero Taivalsaari and Kari Systa, "Cloudberry: An HTML5 Cloud Phone Platform for Mobile Devices," *Software.IEEE*, vol.29,No.4,pp.40-45,July-Aug.2012.
- [17] W. Zhu, C. Luo, J. F. Wang, and S. P. Li, "Multimedia cloud computing," *IEEE Signal Process. Mag.*, vol. 28, no. 3, pp. 59–69, May. 2011.
- [18] Heiko Schwarz, Detlev Marpe and Thomas Wiegand, "Overview of the Scalable Video Coding Extension of the H.264/AVC Standard," *Circuits and Systems for video Technology.IEEE Transactions*, vol.17,No.9,pp. 1103 – 1120, Sep.2007.
- [19] A. Garcia and H. Kalva, "Cloud transcoding for mobile video content delivery," in *Proc. 2011 IEEE Int. Conf. Consumer Electronics (ICCE)*, pp. 379–380, 2011.
- [20] Z. Huang, C. Mei, L. E. Li, and T. Woo, "CloudStream: Delivering high-quality streaming videos through a cloud-based SVC proxy," in *Proc. IEEE INFOCOM Mini-conf.*, 2011, pp. 201–205.
- [21] Weiwen Zhang, Yonggang Wen ; Jianfei Cai ; Wu, D.O., "Toward Transcoding as a Service in a Multimedia Cloud: Energy-Efficient Job-Dispatching Algorithm," *IEEE Vehicular Technology, IEEE Transactions*, vol.63,No.5,pp.2002-2012,Jun.2014.
- [22] Muhamad Felemban, Saleh Basalamah, and Arif Ghafoor, "A distributed cloud architecture for mobile multimedia services," *Network, IEEE*, vol.27, No.5, pp.20-27, Sep-Oct 2013.
- [23] Yao Liu, Fei Li ; Lei Guo ; Bo Shen ; Songqing Chen ; Yingjie Lan, "Measurement and Analysis of an Internet Streaming Service to Mobile Devices," *Parallel and Distributed Systems.IEEE Transactions*, vol.24, No.11, pp.2240-2250, Nov.2013.
- [24] Yu Wu, Zhizhong Zhang, Chuan Wu, Zongpeng Li and Francis C.M.Lau, "CloudMoV: Cloud-Based Mobile Social TV," *Multimedia.IEEE Transactions*, vol.15, No.4, pp.821-832, June.2013.
- [25] Mathias Wien, Renaud Cazoulat, Andreas Graffunder, Andreas Hutter, and Peter Amon, "Real-Time System for Adaptive Video Streaming Based on SVC," *Circuits and Systems for video Technology.IEEE Transactions*, vol.17, No.9, pp. 1227 - 1237, Sep.2007.
- [26] Xiaofei Wang, MinChen, Ted Taekyoung Kwon, Laurence T. Yang, and Victor C. M. Leung, "AMES-Cloud: A Framework of Adaptive Mobile Video Streaming and Efficient Social Video Sharing in the Clouds," *Multimedia.IEEE Transactions*, vol.15, No.4, pp.811-820, June.2013.
- [27] Xiaofei Wang and Min Chen, "PreFeed: Cloud-Based Content Prefetching of Feed Subscriptions for Mobile Users," *Systems Journal, IEEE*, vol.8, No.1, pp.202-207, March-2014.
- [28] X. Jin and Y. K. Kwok, "Cloud assisted p2p media streaming for bandwidth constrained mobile subscribers," in *Proc. IEEE Int. Conf. Parallel and Distributed Syst.*, 2010, pp. 800–805.
- [29] Ke Xua, Ming Zhangb, , Jiangchuan Liuc, Zhijing Qind and Mingjiang Yea, "Proxy caching for peer-to-peer live streaming," *Computer Networks.Elsevier*, vol.54, No.7, pp.1229-1241, May.2010.
- [30] Kalpana Seshadrinathan, Rajiv Soundararajan, Alen Conrad Bovik, and Lawrence K.Cormack, "Study of Subjective and Objective Quality Assessment of Video," *Image Processing.IEEE Transactions*, vol.19, No.6, pp.1427-1441, June.2010.

- [31] Chikkerur, S., Sundaram, V., Reisslein, M., and Karam, L.J., "Objective Video Quality Assessment Methods: A Classification, Review, and Performance Comparison," *Broadcasting. IEEE Transactions*, vol.57, No.2, pp.165-182, June-2011.
- [32] Weiwen Zhang, Yonggang Wen, Zhenzhong Chen, and Ashish Khisti, "QoE-Driven Cache Management for HTTP Adaptive Bit Rate Streaming Over Wireless Networks," *Multimedia.IEEE Transactions*, vol.15, No.6, pp.1431-1445, Oct.2013
- [33] Wu-Hsiao Hsu and Chi-Hsiang Lo, "QoS/QoE Mapping and Adjustment Model in the Cloud-based Multimedia Infrastructure," *IEEE Systems Journal*, vol.8, No.1, pp.247-255, March.2014.
- [34] Vuclip.[Online].Available:  
<http://www.vuclip.com/index.html>