Ubiquitous Control Framework for Delivering Perceptual Satisfaction of Multimedia Traffic

K. L. Eddie Law[1] and Jacek Ilow[2]
[1]*Kirin Cloud Solutions, Ltd.*
[2]*Dalhousie University*
[1]*Hong Kong*
[2]*Canada*

1. Introduction

With the latest computing technology, people can store information including audio, video, and data on the Internet. And the advanced networking protocol designs can easily enable interactive communications among users, applications, and services through wireless tablets and mobile devices (Satyanarayanan, 2001). However, people like to connect to the Internet while moving, the connectivity may vary during the course of an active session of a user. Although many portable device can provide high-speed connectivity, a user may move into area with weak signal or thin bandwidth, which possibly hinder the reception of satisfactory multimedia traffic. In today's business, it is important to make sure the satisfaction of content subscribers. Thus, it is adamant to develop a ubiquitous control framework to offer perceptual satisfaction of multimedia subscribers.

The ubiquitous computing platform should be designed for quality control of multimedia and data context through the Internet. The framework should manage the network and computing resources, such that the delivered information should at least meet the expected minimal perceptual quality of multimedia traffic stream of an end-user. In this chapter, a few basic design parameters for justifying the performance and design of the control framework will be elaborated. Quantitatively, different Quality of Service (QoS) parameters, e.g., packet loss rate, have been widely used for session transmission control. But for visual evaluation, the terminology known as Quality of Experience (QoE) has recently been widely used. QoE is a measure of performance expectations of the end-user; it may augment QoS by providing the quantitative link to user perception. Indeed, the only way to know how customers see your business is to look at it through their eyes.

Nowadays, due to the widespread use of mobile devices, the rapidly increasing demand on network resources impacts the underlying investment on high-performance hardware devices, which then affects the cost of a network architecture. Then, on the other hand, higher visual quality can then impact the number of subscribers and subsequently the top line income of a networking firm. As a result, a good visual quality control system with effective utilization of network resources for a networking firm is desirable. In the following, we shall elaborate the QoS and QoE design issues. Then a multimedia control framework will be proposed to offer satisfactory perceptual to ubiquitous multimedia subscribers. Its improvements will be thoroughly discussed.

2. Visual metrics and control frameworks

2.1 Measuring metrics

From the signal processing algorithmic design, multimedia sessions may consist of voices, images, videos, and data. Different signal types use different encoding/decoding algorithms for storage and transmissions. For example, MPEG (Motion Picture Experts Group) is a family of standards used for coding audio-visual information, e.g., movies, video, music, in a digital compressed format. The JPEG (Joint Photographic Experts Group), GIF (Graphic Image File), BMP (bitmap) are examples of image encoding data formats. Among them, bitmap image takes more memory spaces with sharper imaging quality because it has 256 quantization levels for each of the three base colors. An JPEG image is coded with a lossy Discrete Cosine Transform (DCT). It uses less memory space with a lower visual quality.

Peak signal-to-noise ratio (PSNR) is an easy-to-use error measurement metric, and is widely used for providing quantitative evaluation of receiving multimedia quality. Indeed, the PSNR ratio is more or less a subjective measurement technique, and it may fail to reflect what appear in images. As shown in Fig. 1, two images with identical PSNRs. But the one shown in Fig. 1(b) appears to give inferior visual quality (Winker & Mohandas, 2008). This is due to the local accumulation of errors on some nearby pixels. With the nonlinear functionality of retina in human vision system, the perceived quality can be drastically misleading. With the errors spread evenly across all pixels in the image, the one shown in Fig. 1(a) may be considered with better encoded quality.

(a)   (b)

Fig. 1. Human vision system and images with identical PSNRs (Winker & Mohandas, 2008).

As a result, PSNR may not be able to reflect the visual perceptual quality of multimedia content. That is, some perceptually poor and appealing images may have identical PSNRs (Grega et al., 2008). At this current moment, there is no conclusive measure that should be commonly accepted as the right measure for quantifying QoE. Hence, the Mean Opinion Score (MOS) is then recommended by the International Telecommunication Union (ITU) (ITU-T Recommendation P.800, 1996). It is a subjective measure, and a number of users should

rate the quality using a five-point scale from 1 to 5, inclusively, as listed in Table 1. The MOS is the arithmetic mean of all individual scores for judging the quality of a delivering video.

MOS	Quality	Perception
5	Excellent	Imperceptible
4	Good	Perceptible
3	Fair	Slightly annoying
2	Poor	Annoying
1	Bad	Very annoying)

Table 1. Mean Opinion Score (MOS).

Then from the perspective of network architectural design, the quality of a transmission session through the Internet is usually characterized by the term Quality of Service (QoS) using parameters, such as packet loss rate, transmission bandwidth, queue length, jitter, and delay etc. Each of these parameters can be used for performance analysis. Currently, there are a few standardized QoS associated network designs, for example, the Differentiated Service (Blake et al., 1998) system model through the recommendation of the Internet Engineering Task Force (IETF). In general, network operators have to monitor and manage network resources properly in order to deal with network congestion problems and packet loss issues so as to meet different QoS requirements. Network-introduced errors may be the sources of decoded signal errors. For example, as observed from the two rugby team pictures shown in Fig. 2, both of them suffer identical overall loss conditions in networks. The errors are spread across all pixels in Fig. 2(a) which gives a more appealing appearance. However, the errors are localized in Fig. 2(b), which may irritate the acceptance of a subscriber.

(a)   (b)

Fig. 2. Network condition: 1% packet loss rate, 10 ms delay, 50 μs jitter, 500 kbps bandwidth (Winker & Mohandas, 2008).

Through these observations, QoE and QoS may be related but not in a linear way. What can be the proper way to assert the quality of online multimedia service? Objective evaluation methods are simpler because user inputs are usually not required. For example, data can be retrieved from network-level measurements, e.g., packet loss rate, or media-level measurements, e.g., PSNR, as the input for quality assessment. However, as discussed, they may not be able to judge delivered multimedia quality.

The subjective method using MOS for QoE can be a feasible solution. However, each subscriber may give a completely different MOS result. Furthermore, the same person under emotional stress can give a completely different score. As a result, we can consult some nonlinear functional methods for judging delivering visual quality. And some parameters are not considered in these methods. For example, the loss of volume is not considered by the Perceptual Evaluation of Speech Quality (PESQ), which is recommended by the ITU for determining the quality of a speech signal, in order to make the model tractable. Also, the latency between viewers and the video is not considered in Video Quality Measurement (VQM) model (Rajagopalan, 2010). Higher-order variations, i.e., the burstiness, of end-to-end delay and loss are not considered in many models for assessing VoIP quality.

Similar to PESQ, there are a few subjective methods that can assist in judging visual quality. One of the them is called Visual Differences Predictor (VDP). It is used to characterize the retina response curve. Although the computation complexity is relatively high, it can be used for quantifying image quality based on a reference image. Details of the VDP design can be found in Appendix A.

2.2 Examples of quality control frameworks

There are some basic architectural designs (Agarwal et al., 2008; Huang et al., 2008; Lum & Lau, 2002) for serving multimedia traffic adaptively according to varying network condition. The common goal is to improve the visual quality of multimedia traffic at recipients. In (Lum & Lau, 2002), proxy server or intermediate network server is used to relay and re-adapt information content for changed networking condition. As of today, there are a large amount of proxy video servers deployed across the Internet today. However, most of these proxies are for relay purpose only.

As reported in (Agarwal et al., 2008), a controlled testbed for experimenting video traffic delivery using peer-to-peer (P2P) streaming has been used. The results have indicated that tested P2P streaming systems carry significant overhead (up to 35% over the video stream size) with an average start-up delay of about 11 sec. Besides, an additional video buffering time of 30 sec is needed to combat packet arrival jitters for video playback. Despite these drawbacks, the P2P systems are robust regarding peer churn, and generate larger captured P2P bandwidth than using an over-provisioned server. Furthermore, they have found that quantitative measure such as PSNR, which is used among many P2P video streaming research reports, can not properly reflect the QoE.

Another investigation on P2P streaming can be found in (Huang et al., 2008). The paper offers a generic design framework, and identifies different building blocks in a system, for example, the file segmentation strategy, replication strategy, content discovery and management, piece/chunk selection policy, transmission strategy and authentication. The goal is to achieve a scalable system with efficient replication strategies for offering user-level satisfaction. A new fluency index has then been introduced as a performance measure for evaluating the health of the systems and the user satisfaction. Typically, the index measures the fraction of time a user spent watching a movie with respect to the total time spent on both the waiting and watching times. This design closely relies on the underlying network performance, instead of attempting to interpret and serve different types of content information. That is, the accuracy of the fluency index regarding perceptual quality has not been examined in the paper.

Then in (Law & Leung, 2003), a set of application programming interfaces (APIs) has been designed for programmable nodes in networks. This implies nodes on the Internet can

function together in the form of loosely-coupled computing devices. This indicates that adaptation of traffic can be made inside the Internet. But the operation details for programs to execute must be carefully controlled by network and content service providers. In general, overlay networks provide better security control to network providers and code distribution flexibility to application providers. However, the response times may be slightly longer than those with programmable node concepts. For example, BitTorrent is one simple broadcasting mechanism for code distribution across end-users' computers, which operate as virtual network servers. To advance the design, structured overlays, such as using Distributed Hash Tables (DHTs) (Stoica et al., 2001), can be used.

In (Chen et al., 2009), a proposed framework with QoE consideration known as `OneClick` is proposed. Its operations is trivial. The client informs the server system directly regarding receiving perceptual quality. Upon viewing the multimedia content, when a user finds the receiving quality of content annoying, he or she can click certain button repeatedly to indicate his or her dissatisfaction. Therefore, a session with a larger number of clicks indicates a poorer receiving perceptual quality. The `OneClick` design can be considered as a reciprocal of Mean Opinion Score (MOS) (ITU-T Recommendation P.800, 1996). The `OneClick` may offer real-time response to the server system, although it has not been examined (Chen et al., 2009).

3. Multimedia agent framework

A network infrastructure is shown in Fig. 3. Initially the traffic communicating between a mobile client *C* and service provider *S* is traveling over the Path 1 as shown. Upon moving, the client *C* may have arrived at another location, and the communicating path has been switched to Path 2. The associated networking parameters might have completely changed, for example, the bottleneck link bandwidth and propagation delay, etc. As a result, the amount of data flow and traveling latency could be completely different, which can then impact the perceptual quality of receiving multimedia traffic.

Our proposed quality control framework is called Multimedia Agent Framework (MAF). The goal is to adapt traffic content to changing network constraints dynamically. The foundation of the framework is based on agent technology. A few basic components are defined for the system to operate properly. In Fig. 3, a functional connection is consisted of at least a content provider *S*, a mobile client *C*, and two agents, which sit at the edges on the Internet. Depending on the direction of the traffic flow, the agents are generically named the Ingress Agent (*IA*) and Egress Agent (*EA*). For the depicted traffic flow from a server to a client, the agents connecting to the source and destination are called ingress agent and egress agent,
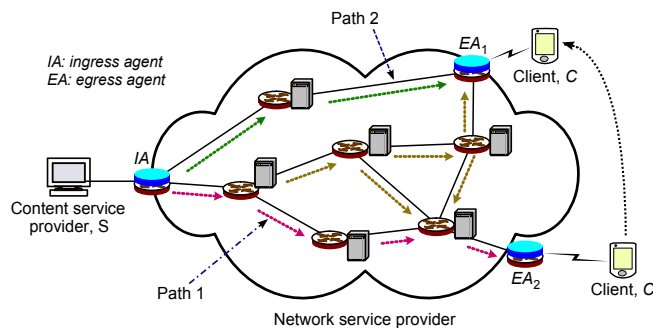

Fig. 3. An operating model of the multimedia agent framework.

respectively. Since wireless connections are not always stable, a mobile client may encounter different perceptual experiences while traveling. For example, the access bandwidth between C and egress agent EA_2 is 54 Mbps for 802.11g, while the one between C and EA_1 can be 11 Mbps for 802.11b. And in this paper, we assume that the bottleneck always occurs at the wireless link between egress agent and client.

To sustain a satisfactory user experience of a mobile client under changing networking conditions, traffic context can be adjusted accordingly (Banavar & Bernstein, 2002; Noble, 2000; Lum & Lau, 2002) through the proposed quality control framework. Apart from the ingress and egress agents, the framework allows content alternation within the Internet upon permitted by both the content service providers and subscribers. But at the current stage, we focus on the basic operating model which functions between the two agents.

3.1 Communication model

A traditional client-server transaction model is shown in Fig. 4(a). Suppose that a client tries to retrieve a web page, which is consisted of one HTML text document, and one picture file. Two separate HTTP GET requests from the client can be used to obtain the two files from the server. But in multimedia agent framework, a high-level conceptual design feature is enabled through the two agents, as shown in Fig. 4(b). Upon receiving the HTTP GET request from client, the egress agent also delivers information regarding the associated resource constraint of the wireless link to the ingress agent. Then the ingress agent can represent the client to collect both files and examine if the latest available resources are sufficient to receive both files in perfect condition. If not, the ingress agent can adjust the file context to meet the latest available QoS constraints for the client.

To have the framework worked as expected, structural control message flows between the egress and ingress agents have been designed. And these control messages are called "capsules" messages. In order to create a lightweight and effective design, multiple operating phases are introduced, which include: initialization, QoS negotiation, and provision phases. When a client moves from one access point to another access point, packets may be lost temporarily due to incorrect routing. This error can be reduced if the path change indication can be saved within the Internet. At the moment, this layer 3 operation is not investigated in this paper. Our current focus is on the changes of resource constraints, whereas multimedia information may not be able to provide the expected quality at the recipient.

For a newly moving-in mobile user, an egress agent may not be aware of any existing active connections. It starts to take notice, if this client sends a new request, retransmits a request, or the agent receives an incoming redirected messages from a server through the mobile IP protocol (Perkins, 2002; Johnson et al., 2004; Law & Lau, 2010). In either of these cases, the egress agent should start an initialization phase. That is, the egress agent attempts to establish and register a relationship with the corresponding ingress agent. The next stage is to begin a negotiation phase. At the current stage, the information being passed between the agents is for QoS monitoring. This enables the ingress agent to make decision on service class selection with an estimated performance for a specific traffic type. In future, information regarding service subscription should be integrated. This indicates that the right of use of a service class must also be verified if it is covered in its paid services. Then afterward, based on measured QoS parameters, the ingress agent makes decision if any modifications should be applied to the traffic, which should be delivered with satisfaction expectation of the subscriber. The role
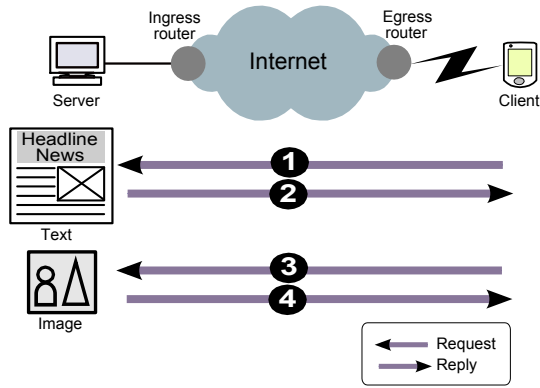
(a) Traditional client-server model.
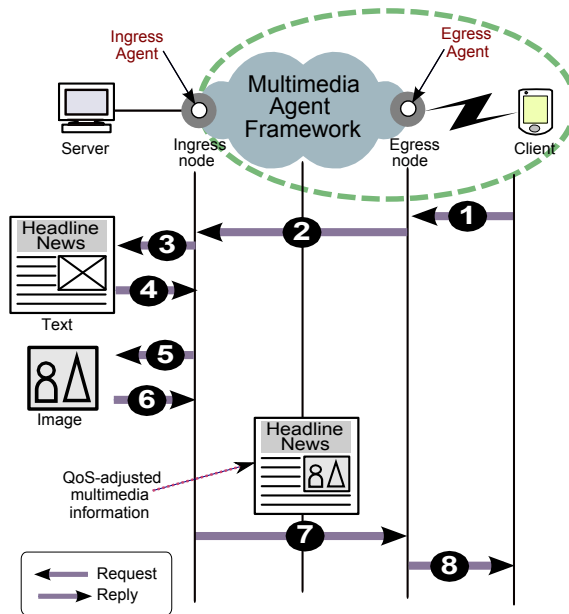
(b) Conceptual model of multimedia agent framework.

Fig. 4. Network transaction models for multimedia traffic.

of QoS monitoring plays an important role in determining the method of adaptations which should be carried out.

With the ubiquity of wireless devices, browsers have been the common tools for accessing different information on the Internet. It becomes naturally and important that the multimedia agent framework should start extending the protocol into REpresentational State Transfer (RESTful) model in future.

3.2 Delivering content adaptation

The multimedia agent framework can tailor the content to meet the QoE expectations of subscribers. At the moment, content adaptations are carried out at the edges of the Internet, i.e., the ingress and egress agents. The adaptation should depend on the business contract between content and network providers, which may be outside the scope of this chapter. Collectively, we call these nodes the adaptation nodes. For carrying out meaningful operations in these nodes, then a number of information must be learnt and communicated among the ingress and egress agents. As shown in Fig. 5, the types of information should include: 1) user's QoE expectations, 2) properties of content material, 3) latest network status, 4) available network access interfaces of devices.
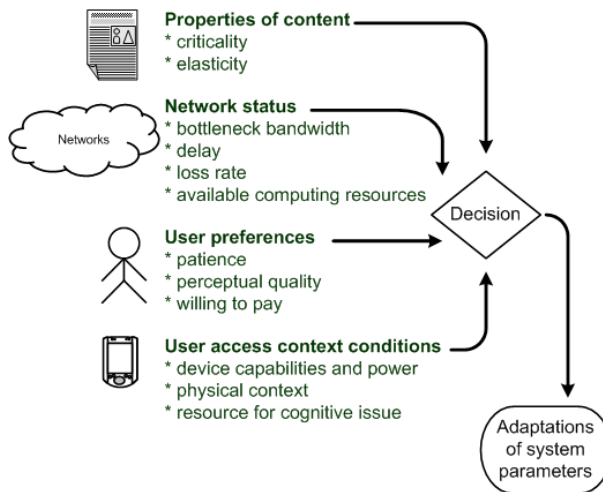

Fig. 5. Decision parameters.

3.3 User preferences and expectation

In order for the infrastructure to work seamlessly and meet the QoEs of subscribers, their preferences and expectations should be set at the initial phases and passed back to the content providers. The associated agents can obtain this information for adaptation purpose, if needed. There are certainly other methods available for this type of information retrieval, e.g., the Service Level Agreement (SLA) through policy-based management. The capability description can follow the defined syntax structure, such as the Composite Capability/Preference Profile (CC/PP) from World-wide Web Consortium (W3C), or Media Feature Sets from IETF.

Some parameters can assist the network to adapt to the QoE expectation of the multimedia traffic:

- Does a subscriber want the content to be retrieved as quickly as possible?
  - In video streaming, significant variations in delivery time result in jerky video and choppy audio. The resulting video has a lower QoE value than a smoothly playback video.
- Does the subscriber want the critical content to be retrieved unfailingly?
  - Ranking or priority of content may be desirable.

For multimedia information, more thorough query should be carried out. Questions may have to be asked regarding, for example, the acceptable choices for picture sizes and compression granularities, etc. These collected data can then be combined into a set of meta-information.

3.4 Content: meta-information

The purpose of having meta-information in the setting is to assist all components in the framework to parse and retrieve desired parameters as quickly as possible. For data files, meta-information may contain file size, version, title, language, and authors. For multimedia content, additional meta-information may include minimal required and desired transmission bit rates, display size, compression ratio, and encoding methods. These extra data can assist the adaptation agents to carry out appropriate operations, if needed. Extensible Markup Language (XML) can be a possible choice for embedding the meta-information regarding the requirements of services and QoEs of subscribers.

3.5 Network status

In general, it is common to characterize a network path between two end-systems using available channel bandwidth, end-to-end delay (or round-trip delay), and packet loss rate. With adaptation capability, computation power can be added to indicate if an adaptation node can handle a large number of connections simultaneously. The agents in network may offer computation services for information being delivered from a sender to a receiver. These parameters are the traditional network layer QoS parameters.

3.5.1 Delays and available resources

For the framework to operate smoothly, we should establish methods to measure available resources in the networks. For example, in a client-server model, packets from server are relayed from one node to other until they reach the egress node that the client is connecting to. In the following, we examine different delay components incurred along the network infrastructure. A packet has to spend times and delays while traveling through nodes and links along the path, respectively. The delay components are additive, and the aggregate delay across the networks is the summation of delay values in various nodes and connecting links. Typically, four different types of delays are introduced in networks:

- Propagation delay, f_{G_i}, for a link i;
- Transmission delay, $f_T(s, B)$, for packet size s and available bandwidth B;
- Processing delay, $f_C(c_p, c_b, s)$, where c_p for incoming packet processing, c_b for outgoing interface determination given a packet with size s;
- Queuing delay, f_{Q_j}, for the queue at node j.

The total one-way delay across the networks is the summation of all these delay components of all nodes and links along a path, as shown in (1). With these parameters, other network characteristics such as bandwidth and computation power can be implicitly reflected in the transmission and processing delay components, respectively.

$$T_D = \sum_{i=1}^{m} \left(f_{T_i}(s, B_i) + f_{G_i} \right) + \sum_{j=1}^{n} \left(f_{C_j}(c_{p_j}, c_{b_j}, s) + T_{Q_j}(\phi_j) \right) \tag{1}$$

for m links and n nodes.

3.6 Adaptation for real-time delivery

Subscribers always expect information being retrieved should arrive briefly after they click the service requests. But they have no idea if they have moved into regions with poor connectivities. Real-time communications are more desirable features for some mobile users, e.g., stock traders. Therefore, in this case, the content carried by the late arrivals of these packets may become unimportant. Hence, in the framework, a parameter W is known as "expected real-time constraint." A subscriber should set the W according to his or her limit of patience on waiting time in the user preference profile; if not, it can be assigned to certain default wait time in system.

Suppose there is a connection being established between a subscriber and a server. A proper path has already determined in the initial setup phase and meta-information has been exchanged. If the total round-trip delay between the client and server is T, such that the forwarding and returning delays are identical, then we have $T = 2T_D$ from Eqn. (1). The expected real-time constraint of a client should not be shorter than the round-trip delay, i.e., $W < T$; otherwise, the retrieved multimedia content can never meet the QoE of the subscriber.

In the case that $W > T$, the T_D may vary due to other traffic in networks. It may have chances to violate the real-time constraint requirement. In this scenario, the agent in the framework acts and attempts to adapt the content in order to meet the W requirement. For example, a sudden change of connecting speed, for example, from wired to wireless access link, has happened. Then content adaptation should be carried out in the network core, for example, by reducing the amount of multimedia traffic with compression and reduced frame size, in order to meet the real-time constraint. For example, the packet size has been modified, and the final delivered packet size is changed from s to s_A bytes when it reaches the subscriber.

In general, many multimedia session is composed of more than one traffic stream. In the example shown in Fig. 6, there are three traffic types in one session, and the importance of each of them is ranked. The rank 1 traffic may contain critical data of size 6,600 bytes; the rank 2 traffic may contain compressible multimedia traffic; and the rank 3 traffic carries unimportant data traffic. The ideal curve indicates that the available bandwidth changes linearly. When there is sufficient bandwidth, all traffic in this session can get the network without any adaptations, i.e., when bandwidth is larger than or equal to 50,000 bytes. But the available bandwidth starts decreasing linearly, the rank 3 data traffic should be removed, and then the rank 2 multimedia traffic should be adapted. The size reduction of the rank 2 traffic due to adaptation is not continuous. This hence leads to the staircase structure as shown. When the available bandwidth is small and then only the rank 1 critical data traffic must always be kept for delivery. This happens in this graph when s_A is below 6,600 bytes, and both the ideal and real curves should stay flat.
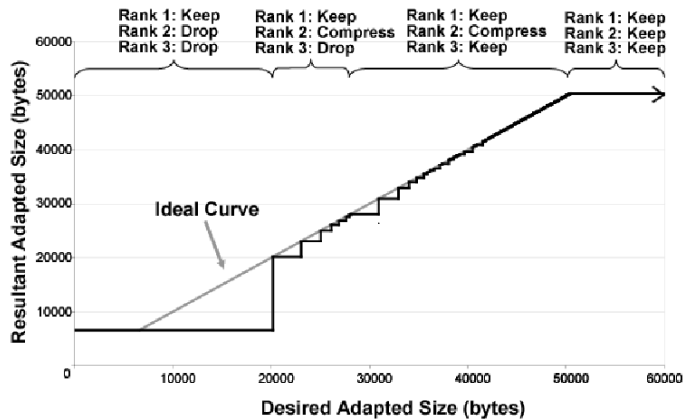
Fig. 6. Desired adapted sizes.

4. Testbed evaluation

A testbed consists of ten nodes has been set up to validate the QoS control framework for multimedia traffic. Experiments have been carried out to confirm if the proposed QoS control framework can adapt traffic content to meet the expectation of subscriber. One set of experiments is to examine the delivery of web pages with real-time information component. Another set of experiments is to adapt video stream to meet the link bandwidth constraint in real-time. For both experiments, classification of traffic types have been preset for carrying out expected component adaptation accordingly.
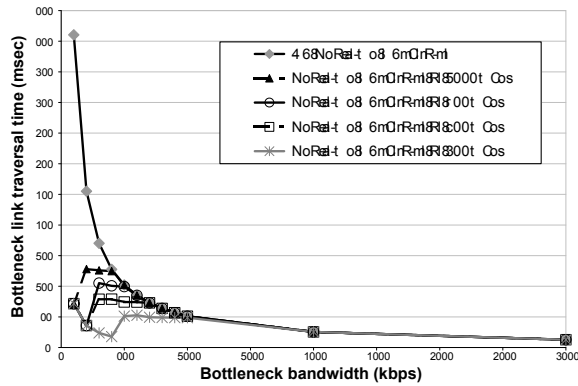
4.1 Real-time delivery of web pages

In this set of experiments, a web page consists of multiple informative components is sent every 2.5 seconds. As listed in Table 2, these components are pre-classified into three ranking classes based on their relative importance. Rank 1 information is considered the most important, and it should be sent to the subscriber whenever possible. Then there is a picture in the page. It belongs to rank 2 class, and can be compressed with lower resolutions upon needed. The other rank 3 components in the HTML page are for creating an appealing look of the page only. They are not as important and can be dropped if resources are running low.

During the experiments, the ingress agent computes the desired adapted size, $s_{ADAPTED}$, after learning the limiting bottleneck bandwidth through receiving the result capsules from the associated egress agent. Measured data are averaged through the last 100 samples. Certainly, wireless access link bandwidth varies due to the mobility of the user, which is simulated through changing the Linux traffic control function. Different access bandwidths are used for testings, which include 115 kbps, 1 Mbps, and 2.4 Mbps. Furthermore, we have imposed different real-time constraints for the delivery of this web page information. The patience limit of a user is interpreted as the real-time delivery constraint. For the tests, the constraints with values of 1000, 800, 600, and 400 msec are used in testbed. This limit setting indicates that the user shall re-click or reload the page when the time is reached. This is the goal of the framework to deliver at least the rank 1 traffic to the subscriber within this time constraint.
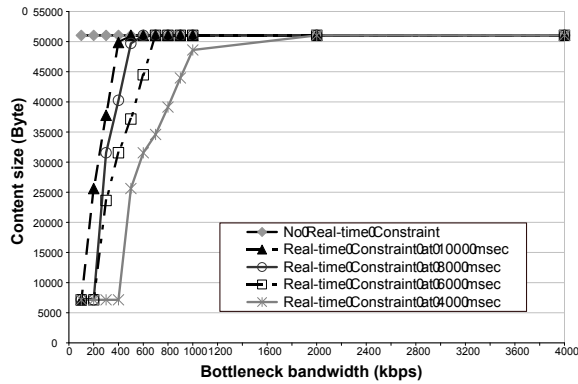
(a) Adapting sizes of web pages.

(b) Link traversal times.
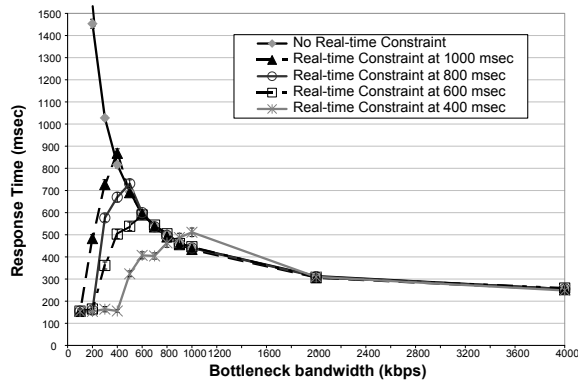
(c) Response times.

Fig. 7. Web page delivery: a) delivery size; b) traversal time; c) response time.

Rank	Content	Size (bytes)
1	real-time data	6600
2	picture	13590 - 25958
3	other data	7900

Table 2. Sizes of various components in a web page: original picture size is 25958 bytes, maximally compressed size is 13,590 bytes.

In Fig. 7(a), the measured $s_{ADAPTED}$ is shown under different user patience limits and bottleneck bandwidth. Fig. 7(b) is obtained by dividing the measured $s_{ADAPTED}$ with the bottleneck link bandwidth, thereby giving the bottleneck link traversal duration values. When selective delivery is triggered, the bottleneck link traversal time remains constant while the bandwidth is shrinking. This trend continues until only the rank 1 component is delivered. Thus, the bottleneck link traversal is the reciprocal of the bottleneck bandwidth multiplied by the size of the rank 1 component. Fig. 7(c) shows the resultant response time across the networks. If there is no limit on the patience threshold, the response time exponentially increases when the bottleneck bandwidth linearly decreases. In fact, when the bottleneck bandwidth is reduced to 100 kbps, the response time surges to $12,016 \pm 339$ msec due to the exponential increase in the backlog of data waiting to be delivered to the client at the last-mile link.

The response time characteristics under various user patience limits are similar. In general, the response time increases exponentially when the bottleneck bandwidth decreases. This trend continues until they reach their respective adaptation thresholds. From then on, the response time decreases linearly with the reduction of bandwidth. When the bandwidth decreases even further, the response time reaches a point beyond which ranks 2 and 3 in-page components are dropped. The response time stays flat at about 150 milliseconds.

4.2 Real-time delivery of images

The next set of experiments is to delivery images across networks. The bottleneck link is the access link of a subscriber. The real-time constraint W is passed from the client preference list to the server, and it can be discovered by the agents in networks. They can detect the existent of real-time constraint of a communication session, and the delivery of images based on the meta-information. If there are multiple good quality paths existing between the two agents, the ingress agent can then have multiple choices in carrying out appropriate operations inside the infrastructure. An exemplary operational details of using a single path computation in the infrastructure is shown in Fig. 8. The goal of the infrastructure is to fit the image delivery to receiver within the duration of W sec.

The image scaling operation starts when the ingress agent receives the incoming Lena BMP images, as shown in Fig. 9(a). In the experiments, the ingress agent converts them into JPEG images, forwards them along a path, and the running times are measured against the time constraints. Regularly, the egress agent reports the measured bottleneck link bandwidth to the ingress agent. Through the measurements, the image compression ratio can be estimated through operations in the infrastructure. Different scaling parameter, λ, can be set between 1 and 16 inclusively, where 16 is the best quality and 1 is the worst. The next question is to determine if the generated images with selected compression ratio can meet the real-time and bandwidth constraints. Then it leads to design of an initial calibration process. The goal of the process is to find the lowest acceptable bound of the compression ratio for a subscriber.
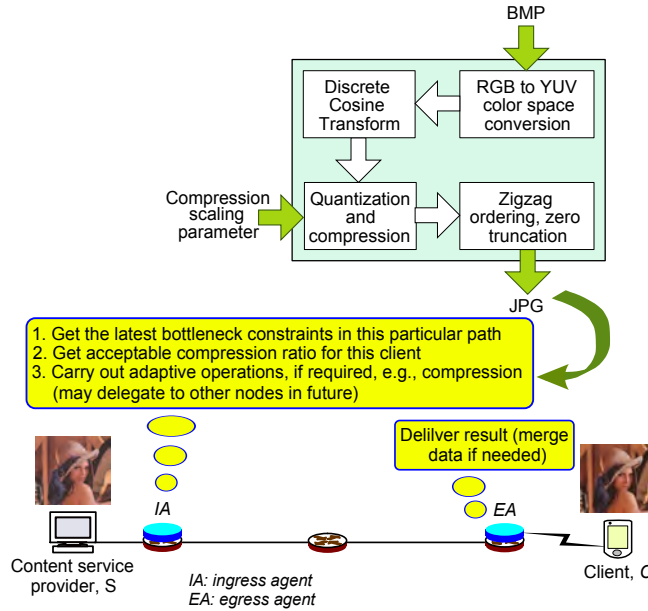
Fig. 8. Real-time image scaling operations.

λ	size (% of original)	$R_{RMS}(C, R)$	# in Fig. 9
1	24%	5.026%	(b)
2	36%	3.985%	(c)
3	45%	3.376%	(d)
4	51%	2.984%	(e)
5	55%	2.607%	(f)
6	63%	2.393%	(g)
10	77%	1.806%	(h)
13	89%	1.562%	(i)
16	100%	1.409%	(j)

Table 3. Visual quality and $R_{RMS}(C, R)$.

With different compression ratios, sizes of resulting compressed images are listed in Table 3. For example, the visual qualities of some compressed images are shown from Fig. 9(b) to Fig. 9(j). Through the initial calibration process, a subscriber can actually select the worst quality image that he or she can accept, and inform the content provider. With this information, the infrastructure can then be aware of the minimum QoE expectation of the subscriber. And from the Table, an ingress agent can pick a lowered but acceptable bounding value of the compression scaling parameter to meet the latest networking conditions, if they have just turned worse. There are two parameters that should be considered before carrying out compression operations. The first one is to determine if the real-time constraint can be met. The second is to determine if the image quality can meet the quality expectation of a user. Both conditions must be met; otherwise, the images are dropped at ingress agent, because there are no good reasons to send an unacceptable low quality images.

(a)   (b)   (c)   (d)

(e)   (f)   (g)   (h)

(i)   (j)

Fig. 9. Visual quality of compressed images: (a) original in bitmap format, (b)-(j) compressed in JPEG formats

4.3 Real-time delivery of streaming content

Multimedia applications, such as IPTVs, and online conference meetings, belong to the real-time streaming traffic class with temporal relationship among sending information. Typically, a streaming movie session may consist of three traffic types: video, voice, and/or data such as the subtitle. Actions are spread across multiple consecutive video frames. Apart from the blocking effects in images, video motion may not be reconstructed properly if some motion vectors for decoding are lost. As a result, subscriber could have trouble interpreting the video content. When a video is transmitted across networks, some video frames can be dropped. Then, long overdue frame group not displaying at all causes viewer waiting for long delay, and the the content may appear out-of-order. Hence, when there are more frames dropped, the reproducing video becomes jerky and gives poor perceptual interpretation.

It is important that our proposed framework can offer satisfactory experience of real-time video delivery. In the experiments, video frames are extracted from a theatrical screen

advertisement promoting the 1955 Chevrolet models. The advertisement movie has long sequences of car movements. Inter-frames smoothness can be observed easily, and the perceived quality can be assessed easily.

For demonstration purpose, this movie clip does not carry voice component which is actually replaced by subtitle. Each frame in video is a 24-bit RGB bitmap image with a size of 63,414 bytes and a visual dimension of 176 × 120 pixels. The sizes of the accompanying subtitles range from 97 to 120 bytes. Compression ratio may differ frame by frame depending on the networking conditions. With the highest compression ratio, compressed frame images have sizes ranging from 8,806 bytes (13.89% of the originals) to 12,674 bytes (19.99%). Two frames are extracted every second for creating one new accompanying subtitle. Eighty-eight frames are extracted. At the server, the video clip is replayed continuously in the experiments. The relationship between compressed size and compression scaling parameter, λ, is encoded using meta-information. Thus, the network infrastructure can efficiently choose the closest adapted size, s_A, with an appropriate compression parameter λ. Then the network nodes execute compression operations according to λ. The real-time delivery scheme of the video experiment is based on a 2-level ranking levels: 1) subtitle is classified as rank 1, R_1, 2) frame image is classified as rank 2, R_2; where rank 1 has higher priority than rank 2 traffic.
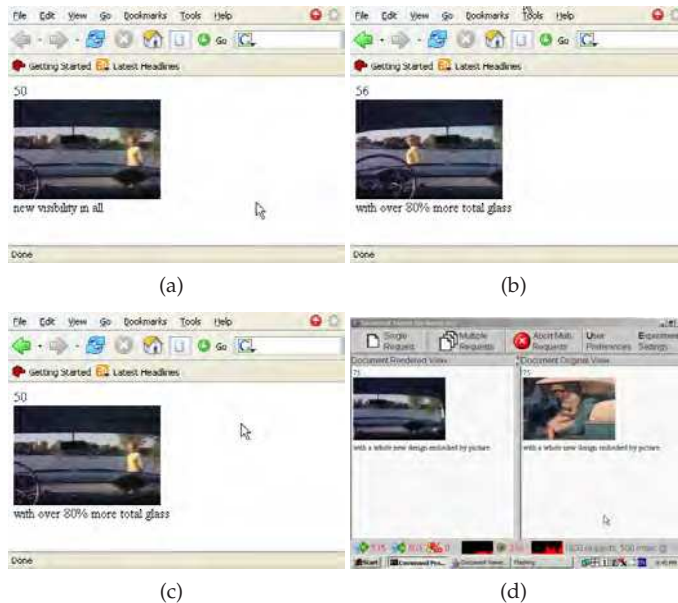

Fig. 10. (a) Frame 50, (b) frame 56, (c) time at frame 56, picture at frame 50, (d) comparing viewer with traffic load monitoring.

For video communication, the ingress agent can have choices on delivering frames across the networks. Depending on the latest measured network conditions, the agent can send frames unchanged, compressed, or simply drop the frames. Certainly, frames sent may still get dropped inside the networks due to congestion, and these are called frame lost events.

Two snapshots of the movie are shown in Fig. 10(a) and 10(b). When the network delay is too long with a real-time delay constraint being applied. Then frames are dropped at the ingress

agent to reduce the traffic load in testbed. For example, as shown in Fig. 10(c), the movie is stuck at frame number 50, but the subtitle has arrived at frame 56. This implies that the proposed pervasive infrastructure can enforce certain minimal QoS quality. In this case, the receiver is still able to see the latest subtitle section, or he or she can listen to the voice part if the voice section exists. An MPEG-1 viewer at the subscriber has been created to compare a locally stored copy with the remotely retrieving video. The viewer, as shown in Fig. 10(d), also reports the packet loss and CPU consumption.

For performance measurements, the link speed of the simulated wireless access shrinks from 3 Mbps to 100 kbps. With different real-time constraints applied, different operations on video frames have been carried out. If bandwidth is abundant, frames are sent uncompressed, as in all constraint cases with link speed of 3 Mbps. But when the link rate reduces, and the real-time constraints is shorter, then more frames are compressed, or dropped. It is always important to observe that, for the cases reported in Fig. 11, the amount of frame losses inside the networks are small. This satisfies the goal of the proposed QoS control framework for multimedia traffic, that is, to deliver as much information as possible to the subscribers.
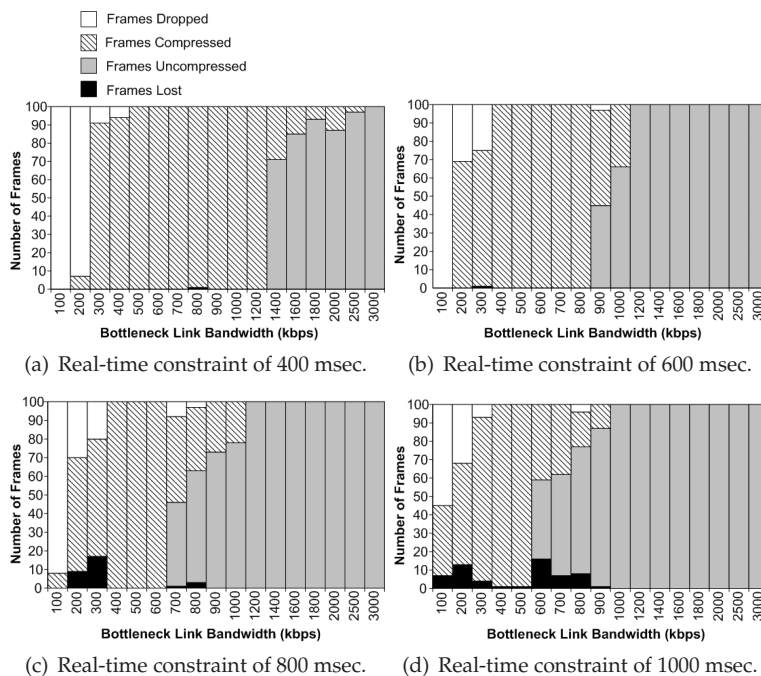

(a) Real-time constraint of 400 msec.   (b) Real-time constraint of 600 msec.

(c) Real-time constraint of 800 msec.   (d) Real-time constraint of 1000 msec.

Fig. 11. Video stream modifications.

5. Conclusion

In this chapter, an agent-based quality control framework for delivering satisfactory multimedia traffic across the Internet has been designed. The framework is currently built on top of existing networking protocols. The major components in the platform consists of ingress and egress agents. QoS monitoring capsules are regularly sent between agents in order

to enable the ingress agent to adapt the content information, while meeting the requirements and expectations of subscribers and end-users. At the current phase, communicating protocol designs between the two agents at the edges of networks have been tested. Furthermore, different traffic types have been thoroughly experimented. Through multimedia content classification, whether it is for real-time or non-real-time, important or unimportant, traffic can be sent to subscribers to meet their expected multimedia quality in our framework. And in future, more thorough investigation shall be carried out to enable routers inside the Internet to assist and relieve the loads of the ingress and egress agents.

6. Appendix A: Visual difference predictor

One possible filter responsive function is called Visual Differences Predictor (VDP) (Daly, 1993), which can be adopted to model the retinal sensitivity of human being. The retinal response to luminance is a non-linear function, and it is used to evaluate the quality of distorted images. Algorithm derived for VDP attempts to indicate the probability of detecting a difference between the two images on a pixel-by-pixel basis. In our testing, we have adopted a simplified VDP version using a relative mean calculation. These computations may not be needed to run at all times; the image difference computations can be carried out during quality calibrations or upon user requests.

The luminous response, $l(i,j)$ shown in (2), is called amplitude non-linearity value,

$$l(i,j) = \frac{L(i,j)}{L(i,j) + c_1(L(i,j))^b} \tag{2}$$

where $L(i,j)$ is the luminance of the pixel (i,j) in dimension cd/m^2. Typically assigned values for the parameters b and c_1 are 0.63 and 12.6, respectively. Spatial frequencies $f_l(u,v)$ can be obtained through the Fast Fourier Transform of the luminance sensitivity spatial pattern $l(i,j)$. It gives the magnitude at horizontal spatial frequency at u and vertical spatial frequency at v. Human being is aware of changes in signal contrast. The contrast sensitivity function (C_S) is used to model the retinal contrast sensitivity to a spatial frequency.

$$C_S(u,v) = (0.008\, z^{-1.5} + 1)^{-0.2} \times$$
$$1.42\sqrt{z}\, e^{-0.3\sqrt{z}} \times \sqrt{1 + 0.06 e^{0.3\sqrt{z}}} \tag{3}$$

where $z = u^2 + v^2$.

The C_S, in (3), gives a two-dimensional spatial frequency plane (u,v). As shown in (4), the C_S function filters the $f_l(u,v)$ values and obtains the parameter Filtered Magnitudes of the Amplitude Non-Linearity Value (FMANLV), $g(u,v)$. The $g(u,v)$ measures the luminance sensitivity filtered by the retinal response with respect to the spatial frequency (u,v), which is

$$g(u,v) = f_l(u,v) * C_S(u,v). \tag{4}$$

A.1 Calibration operations

There are no needs to run these computations in real-time. In the proposed platform, the framework can initiate an initial calibration phase between subscriber and content provider. In other words, the ingress and egress agents can subsequently set appropriate preferences for a subscriber.

The multimedia content provider may send a reference image R with a reference $g_R(u,v)$ value. Then it sends other compressed testing image, C, with $g_C(u,v)$ at the recipient under the assumption of certain network conditions. Suppose this is the least accepted quality of a subscriber, then the egress agent can compute a relative image metric through normalizing the $g_C(u,v)$ of the test image with respect to the $g_R(u,v)$ of reference image. The resulting measure is called the relative root-mean square (RMS) error of the reference content,

$$R_{RMS}(C,R) = \sqrt{\frac{\sum_{u,v}(g_C(u,v) - g_R(u,v))^2}{\sum_{u,v} g_R(u,v)^2}} \times 100\%. \tag{5}$$

A smaller $R_{RMS}(C,R)$ value indicates that the received image is visually closer to the original one. This quality bound of a streaming multimedia content can be set through finding the largest accepted $R_{RMS}(C,R)$ value of a content subscriber. This bounding measure is returned to the ingress agent for determining appropriate compression operations in future.

7. References

Satyanarayanan M. (2001). Pervasive computing: vision and challenges. *IEEE Personal Communications*, Vol. 8, No. 4, pp. 10-17.

Winkler S. & Mohandas P. (2008). The Evolution of Video Quality Measurement: From PSNR to Hybrid Metrics, *IEEE Trans. Broadcasting*, Vol. 54, No. 3, pp. 1–8

Chen K.-T.; Tu C.-C.; Xiao W.-C. (2009). OneClick: A Framework for Measuring Network Quality of Experience, *IEEE Infocom'09*, Brazil.

Grega M.; Janowski L.; Leszczuk M.; Romaniak P.; Papir Z. (2008) Quality of Experience Evaluation for Multimedia, *Telecommunication Review*, pp. 142-158, No. 4, 2008, Poland.

Methods for Subjective Determination of Transmission Quality, Inter. Telecommunication Union (ITU-T)

Rajagopalan R. (2010). Video Quality Measurements for Mobile Networks. Openwave. http://openwave.com/sites/default/files/docs/solutions/Video Quality MeasurementsWPfinal0710.pdf

Rahrer T.; Fiandra R.; Wright S. (2006). *Triple-play Services Quality of Experience (QoE) Requirements*, DSL Forum, Architecture & Transport Working Group, Technical Report TR-126

Chen B. & Cheng H.H. (2010). A Review of the Applications of Agent Technology in Traffic and Transportation Systems, *IEEE Trans. Intelligent Transportation Systems*, Vol. 11, No. 2

Lee J.-S. & Hsu P.-L. (2007). Implementation of a Remote Hierarchical Supervision System Using Petri Nets and Agent Technology, *IEEE Trans. Systems, Man, Cybernetics - Part C*, Vol. 37, No. 1

Blake S.; Black D.; Carlson M.; Davies E.; Wang Z.; Weiss W. (1998). *An Architecture for Differentiated Services*, RFC 2475. Internet Engineering Task Force, 1998.

Law K. L. E.; Leung R. (2003). Design and Implementation of Active Network Socket Programming, *Microprocessors and Microsystems J.*, Vol. 27, Issues 5-6, pp. 277-284, June 2003.

Agarwal S.; Singh J. P.; Mavlankar A.; Bacchichet P.; Girod B. (2008). Performance of P2P Live Video Streaming Systems on a Controlled Test-bed, *Tridentcom'08*, Mar. 18-20, 2008, Innsbruck, Austria.

Huang Y.; Fu T. Z. J.; Chiu D.-M.; Lui J. C. S.; Huang C. (2008). Challenges, Design and Analysis of a Large-scale P2P-VoD System, *ACM Sigcomm'08*, Aug. 17-22, 2008, Seattle, USA.

Jrad Z. E.-F.; Benmammar B.; Correa J.; Krief F.; Mbarek N. (2005). A User Assistant for QoS Negotiation in a Dynamic Environment Using Agent Technology, *2nd IFIP Inter. Conf. Wireless Optical Communications Networks (WOCN)*, pp. 270-274

Pratistha I. M.; Zaslavsky A.; Cuce S.; Dick M. (2005). Improving Operational Efficiency of Web Services with Mobile Agent Technology, *IEEE/WIC/ACM Inter. Conf. Intelligent Agent Technology*, pp. 725-731

Banavar G. & Bernstein A. (2002). Software infrastructure and design challenges for ubiquitous computing, *Communications of the ACM*, Vol. 45, No. 12, pp. 92-96

Noble B. D. (2000). System support for mobile, adaptive applications, *IEEE Personal Communications*, pp. 44-49

Lum W. Y. & Lau F. C. M. (2002). A context-aware decision engine for content adaptation," *IEEE Pervasive Computing*, pp. 41-49

Byers J. W.; Considine J.; Mitzenmacher M.; Rost S. (2004). Informed Content Delivery Across Adaptive Overlay Networks, *IEEE/ACM Trans Networking*, Vol. 12, No. 5

Perkins C. (2002). *IP Mobility Support for IPv4*, IETF, RFC 3344

Johnson D.; Perkins C.; Arkko J. (2004). *IP Mobility Support for IPv6*, IETF, RFC 3775

Law K. L. E. & Lau C. (2010). Mobility Service Agent, *3rd IEEE Workshop on Wireless and Internet Services (WISe)*, in conjunction with *35th IEEE Conference on Local Computer Networks (LCN)*

Daly S. (1993). The Visible Differences Predictor: an Algorithm for the Assessment of Image Fidelity, *Digital Images and Human Vision*, Editor: A.B. Watson, pp.179-206, MIT Press, Cambridge MA

Stoica T.; Morris R.; Karger D.; Kaashoek F.; Balakrishnan H. (2001). Chord: A scalable Peer-To-Peer lookup service for internet applications, *ACM SIGCOMM'01*, Aug. 2001.

Advanced Topics in Multimedia Research

Edited by Dr. Sagarmay Deb

As multimedia has become a very important technology, significantly improving people's lives, this book provides an up-to-date scenario of various fields of research being carried out in the area. The book covers topics including web-based co-operative learning, effective distance learning through multimedia, quality control of multimedia on the internet, recovery of damaged images, Network-on-Chip (NoC) as a global communication vehicle, and Network GPS for road conditions (such as traffic and checkpoints). We believe that the book will help researchers in the field to proceed further in their research on multimedia.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

K. L. Eddie Law and Jacek Ilow (2012). Ubiquitous Control Framework for Delivering Perceptual Satisfaction of Multimedia Traffic, Advanced Topics in Multimedia Research, Dr. Sagarmay Deb (Ed.), ISBN: 978-953-51-0078-2, InTech, Available from: http://www.intechopen.com/books/advanced-topics-in-multimedia-research/ubiquitous-control-framework-for-delivering-perceptual-satisfaction-of-multimedia-traffic