Modeling, Characterizing, and Enhancing User Experience in Cloud Mobile Rendering

Yao Liu, Shaoxuan Wang, Sujit Dey
Mobile System Design Lab, Dept. of Electrical and Computer Engineering
University of California, San Diego
{yal019, shaoxuan, dey}@ece.ucsd.edu

***Abstract*—Cloud Mobile Rendering (CMR), where compute intensive rendering is performed on cloud servers instead of on mobile devices, can be a promising approach to enable rich rendering based multimedia applications on battery and CPU constrained mobile devices. However, since the video rendered in the cloud has to be transmitted to the mobile device over a wireless network with fluctuating and constrained bandwidth, the resulting user experience can be impacted. In [9], an adaptive rendering approach was proposed, wherein multiple rendering factors can be adapted such that the bit rate of the encoded rendered video is compatible with the available network bandwidth. However, changing the rendering factors may itself have adverse impact on user experience, which has not been studied earlier. In this paper, we analyze the impairment of rendering factors on the quality of user experience, and combine the rendering impairments with impairments due to video encoding factors (like bit rate and frame rate) and network factors (like bandwidth and delay) to formulate a model to measure the user experience during a Cloud Mobile Rendering session. We term this model CMR-UE in this paper. We describe our method to derive the CMR-UE model, and demonstrate its accuracy through subjective testing using participants at UCSD. We use the CMR-UE model to study the trade-off between the impact of rendering and video encoding factors on user experience, and find the optimal rendering settings that maximize CMR-UE for any given network condition. Next, we use the CMR-UE model to measure user experience during CMR sessions on a live cellular network. We demonstrate how user experience can be significantly enhanced by using appropriate rendering settings under fluctuating network bandwidth conditions.**

I. INTRODUCTION

With the growth and success of mobile applications, there is growing interest in enabling rich multimedia applications, which involve 3D graphic rendering, like multi-player Internet gaming [1][8], virtual reality and augmented reality [2], on mobile devices. Graphic rendering is not only very computation intensive, it can also impose severe challenges on the limited battery capability of an always-on mobile device. While mobile devices are expected to have increased rendering capabilities in the future, one can expect a growing gap between the rapid advances and requirements of 3D rendering, and the capabilities of mobile devices, in particular in the context of limited battery capabilities. Hence, it may be promising to investigate an alternative approach, Cloud Mobile Rendering (CMR), where compute intensive rendering is performed on cloud servers instead of on mobile devices, and the rendered video is encoded and streamed to the mobile devices. Figure 1 shows the overall architecture of the Cloud Mobile Rendering system. When a client starts a CMR session from a mobile device, a Cloud Mobile Rendering (CMR) server will initialize a rendering engine and encoder/streamer for this mobile device. Then the CMR server starts to process the application data to render raw video, which is encoded and streamed to the mobile client through wireless networks. On the other hand, mobile client can send his/her control commands from mobile device to the CMR server.

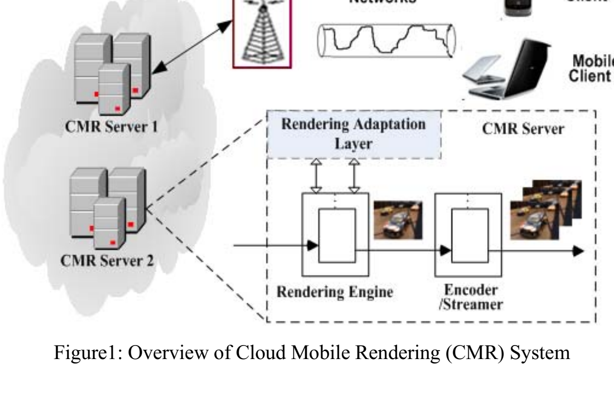

Figure1: Overview of Cloud Mobile Rendering (CMR) System

Compared with enabling rendering based applications where the rendering is performed on the mobile device itself, the Cloud Mobile Rendering approach has several advantages. Firstly, it has high scalability across various mobile platforms. Take cloud-based mobile gaming as an example. Different versions of the same game will not need to be developed to accommodate the differences among various mobile hardware platforms and operating systems. With the CMR approach, only a single version of the game needs to be executed on a CMR server, while mobile users can play this game across multiple platforms. Secondly, a CMR based application will be able to use the most advanced rendering technologies, without concern about computation constraints, and hence be able to provide the richest graphic rendering quality. Thirdly, CMR based applications will have significantly less battery consumption than if rendering is performed on the mobile device itself, thus making such rendering applications feasible and popular on mobile devices.

Though CMR can be a promising approach to provide rich rendering multimedia services for mobile devices, ensuring user experience can be challenging, considering (a) rendering video will have to be streamed from the CMR servers to the mobile devices through error-prone wireless networks, and (b) streaming video for each user from CMR server to his/her device will consume significant backhaul and wireless network bandwidth. Motivated by the above challenges, we proposed a rendering adaptation technique [9] for cloud mobile gaming (a type of CMR service), wherein multiple rendering factors can be adapted such that the bit rate of the encoded rendered video is in accordance with the available network bandwidth, thus minimizing user experience artifacts due to wireless network factors like bandwidth availability, delay, and packet loss. However, while adapting rendering settings can mitigate network artifacts, changing the render settings can itself have adverse impact on overall user experience. Therefore, it is vital to derive a user experience model, which can be used to measure and understand the effect of rendering factors on user experience, and determine whether and when mitigating network artifacts using rendering adaptation can lead to enhancement of overall user experience.

In [8], we proposed a Mobile Gaming User Experience (MGUE) model to measure user experience for Cloud Mobile Gaming (CMG) service, which is one kind of CMR service. The MGUE model takes into account the impact on user experience of video encoding factors (like bit rate and frame rate) and network factors (like bandwidth and delay). In this paper, we adopt and enhance MGUE model to take into account the impact of rendering factors described later, together with network and video encoding factors, in developing the CMR-UE model.

Several approaches have been developed to model the quality of experience of video streaming [3][4][5]. According to these models, user experience is dependent on video quality factors like PSNR, and network factors like delay and packet loss. However, these models are not suitable to model the quality of experience of CMR applications with rendering adaptation technique, since they haven't considered the impact of graphic rendering factors on user experience. There have been some other literature which has characterized the impact of graphic rendering on processing load [6] [7], but the effect of graphic rendering on user experience has not been studied.

In this paper, we will develop a comprehensive model which can characterize simultaneous effects of graphic rendering, video encoding and networking artifacts on user experience for CMR service. The remainder of this paper is organized as following: in section II, we introduce the rendering factors that will affect user experience of CMR service. Section III describes the impairment of these factors on user experience. In section IV, we first introduce CMR-MOS metric as a way to measure CMR-UE, and then derive the CMR-UE model and validate it using subjective testing. In section V, we use the CMR-UE model to study the trade-off between the impact of rendering and video encoding factors on user experience, and find the optimal rendering settings that maximize CMR-UE for any given network condition. Section VI shows the user experience improvement achieved by using those optimal rendering settings to adapt video bit rate in real mobile wireless network. Section VII summarizes our findings in this paper.

II. Factors Affecting Mobile User Experience of Cloud Mobile Rendering Service

The first step to study and implement the CMR-UE model is to identify the impairment factors, which impacts user experience of a CMR session. In this section, we first introduce the impairment factors of CMR service, and then provide a detailed explanation of how rendering factors affect user experience.

As described in section I (figure 1), during a Cloud Mobile Rendering session, graphic rendering is performed at a cloud server, with the resulting video subsequently encoded and transmitted over the bandwidth constrained and error-prone wireless network. Therefore, as shown in figure 2, there are three major categories of objective factors: graphic rendering factors, video encoding factors, and mobile network factors. And the user experience mainly depends on three subjective factors: graphic quality, video quality, and response time. All these objective factors would affect the subjective factors in different manners. Graphic rendering factors will only affect the user perceived visual quality of the graphics, while encoding factors and network factors affect user perceived video quality and response time in a complex manner. For instance, network bandwidth will affect user perceived video quality as well as the response time, while video codec can affect both video quality and video encoding time thus the total application response time.

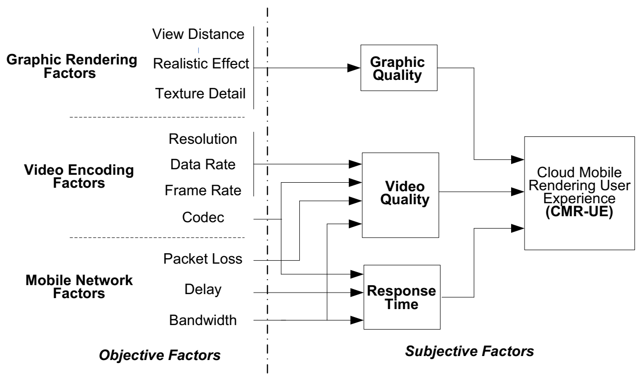

Figure2: Objective and subjective factors affecting user experience of CMR applications

Figure 3: Visual Effect in different settings of *View Distance* and *Texture Detail*: (a) top-left, 300m view and high texture detail level (b) top-right, 60m view and high texture detail level (c) bottom-left, 300m view and medium texture detail level (d) bottom-right, 300m view and low texture detail level

We next describe the impact of different rendering factors (figure 2) on user experience. Figure 3 shows visual quality differences when different settings are used for different rendering factors in a cloud mobile gaming application, which is an example CMR application. For instance, figure 3(a) and (b) compare the visual effect differences between two different view distance settings (300m and 60m). A shorter view distance (figure 3(b)) will have less video bit rate needed, such that we can leverage it to avoid network congestion especially when the network bandwidth is extremely low. However, obviously the shorter view distance has impairments on user perceived gaming quality. Therefore, we need to study how view distance affects user experience, and if there is a certain acceptable view distance threshold, below which user cannot accept the quality of his/her experience any more. The above understanding will help guide the CMR system to perform the right tradeoff between increasing view distance and possibly affecting graphic visual quality, versus having network congestion affecting video quality and response time. Similarly, figures 3(a), (c), (d) have compared visual quality differences for different rendering texturing detail levels. When rendering texturing detail level is reduced, the bit rate needed to encode the resulting video is also reduced. But a similar question will arise: how good will the user experience be for the different texturing detail levels? Motivated by the above purpose, the major contribution of this paper is to develop (section III and IV) a user experience model (CMR-UE), which takes into account rendering factors in addition to network and video encoding factors [8], to quantitatively measure the user perceived CMR service quality.

III. Derivation And Validation of Impairement Functions for Rendering Factors

In this section, we focus on studying the effects of rendering factors on CMR-UE. Based on the experiment with a group of test subjects, we derive an impairment function I_R indicating the graphic rendering quality as discussed in section II (figure 2). In the next section (section IV), we will derive a complete CMR-UE model with this new I_R, and video encoding impairment function (I_E) and network impairment functions (I_D and I_L) which were derived in [8].

A. Impairment Function for Each Rendering Factor

We first define a rendering impairment function, denoted as I_R, to indicate the impairment caused by graphic rendering. I_R is a metric that takes value from range [0, 100], and the relationship between I_R and the perceived graphic quality is listed in Table I (the lower the I_R, the better the graphic rendering quality).

TABLE I. Graphic Quality Rating Criteria and I_R Values

I_R	Description
0	Excellent experience, no rendering impairment at all
0-20	Minor rendering impairment, will continue CMR session
20-40	Noticeable rendering impairment , might quit CMR session
40-60	Clearly rendering impairment, usually quit CMR session
60-100	Annoying experience, unacceptable quality,definitely quit

TABLE II. Experiment Parameters for PlaneShift

Parameters		Experiment Values		
Realistic Effect		H(High)	M(Medium)	L(Low)
	color depth	32	32	16
	multi-sample (factor)	8	2	0
	texture-filter (factor)	16	4	0
	lighting mode	Vertex light	Lightmap	Disable
Texture Detail (down sample)		0, 2, 4		
View Distance (meter)		150, 120, 100, 70, 60, 40, 20		

As shown in figure 2, the graphic quality for most CMR applications is determined by 3 factors: *View Distance*, *Realistic Effect* and *Texture Detail level*. Each of these factors would affect graphic quality from a different aspect, and the total rendering impairment I_R can be formulated with three impairment sub-functions:

$$I_R = I_{VD} + I_{RE} + I_{TD} \quad (1),$$

where I_{VD} represents the impairment caused by changing view distance, I_{RE} indicates the impairment caused by changing realistic effect, and I_{TD} indicates the impairment of changing texture detail.

In the next subsections (section III.B and III.C), we derive the impairment sub-functions I_{VD}, I_{RE}, and I_{TD} by conducting a set of subjective quality assessment experiments, and the accuracy of the linear model (Equation 1) will be validated with another set of subjective experiment in section III.D.

B. Subjective Quality Assessment Experiment

To derive the impairment functions (I_{VD}, I_{RE} and I_{TD}) in equation 1, we developed and conducted a subjective quality assessment experiment, using a group of participants comprising of 18 students at UCSD. We used a cloud mobile gaming application as a demonstration example of CMR application. The selected experimental game is a MMORPG game PlaneShift. The experiment environment is shown in figure 4, where the mobile device is connected to the game server through a network emulator, which can be used to control the network condition. In the experiment environment, the game logic and rendering process are executed by the server, and mobile device only sends control command and displays gaming video. We kept the network bandwidth to be sufficiently large, and only changed the graphic rendering factors on the server. Each experiment participant played the game on the mobile device with different combinations of the rendering factors shown in Table II, and provided assessment of the rendering impairments I_R using an impairment rating system shown in Table I. Finally, the results of the study

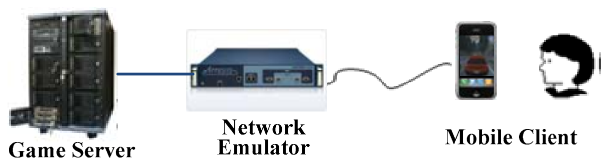

Figure 4: Experiment Framework

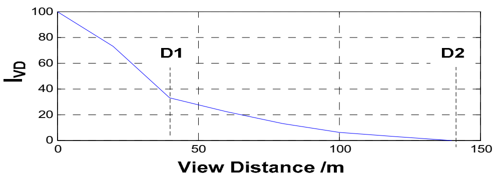

Figure 5: Impairment function of view distance for PlaneShift

group were tabulated for further analysis and derivation of impairment functions.

C. Derivation of Impairment Function I_{VD}, I_{RE} and I_{TD}

To derive I_{VD}, I_{RE} and I_{TD}, we use the result of subjective tests where we only change one of rendering factors while keeping the other two rendering factors at their best values. For example, to derive I_{VD}, we only change view distance while keeping realistic effect at high level (color depth is 32; multi-sample is 8; texture filter 16; and lighting mode is vertex light) and texture detail (down-sample rate) at 0 (without any texture detail down-sample).

Figure 5 depicts the impairment for rendering factor view distance, I_{VD}. Based on the experiment results, we have two observations: first, there is no impairment when view distance is bigger than a certain threshold D2 (for this specific game video, D2= 140m); second, there is an obvious slope change at the point where view distance equals a certain value D1 (D1= 40m for this game). D1 and D2 divide the view distance value into different segments. In the first segment ($0<d<D1$, where d is the view distance value), I_{VD} decreases from highest 100 to 30. In the second segment ($D1<d<D2$), I_{VD} decreases from highest 30 to 0. Consequently, the impairment sub-function I_{VD} can be derived, using linear regression analysis, as the following:

$$I_{vd} = \begin{cases} 0 & (d > D2) \\ \alpha \times [(D2 - d)/(D2 - D1)] & (D2 > d > D1) \\ \alpha + \beta \times (D1 - d) & (D1 > d > 0) \end{cases} \quad (2).$$

The values of coefficients in Equation (2) are listed in Table III. Note that for other rendering applications, these values will be different.

Similarly, we use the same method above for deriving I_{VD} to derive I_{RE} and I_{TD} (where only realistic effect or texture detail

TABLE III. VALUE OF COEFFICIENTS IN EQUATION 2 FOR PLANESHIFT

Game	D1	D2	α	β
PlaneShift	40	140	30	1.75

TABLE IV. IMPAIRMENT FUNCTION OF REALISTIC EFFECT FOR PLANESHIFT

Realistic Effect	High	Medium	Low
I_{RE}	0	4	15

TABLE V. IMPAIRMENT FUNCTION OF TEXTURE DETAIL FOR PLANESHIFT

Texture Detail (down-sample)	0	2	4
I_{TD}	0	12	34

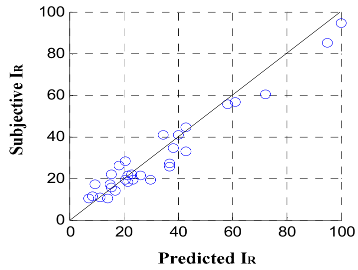

Figure 6: Relationship between predicted and subjective I_R value

level is varied), and obtain the value of I_{RE} and I_{TD} as shown in Table IV and Table V. Derivation of impairment sub-functions I_{VD}, I_{RE}, and I_{TD}, allows derivation of the impairment function I_R according to equation 1. Next, we validate the accuracy of I_R using another set of subjective assessment experiments.

D. Validation of Impairment Function I_R

To validate impairment function I_R, we conducted another set of experiments with a new set of participants, where the participants now provide assessment of rendering impairment I_R when multiple rendering factors, I_{VD}, I_{RE}, and I_{TD} are varied simultaneously. For each such experiment, we compare the subjective I_R scores the participants give and the predicted I_R values calculated from our impairment function I_R (Equation 1 and 2, Table III, IV and V). As shown in figure 6, the correlation between predicted I_R (x-axis) and subjective I_R (y-axis) is 0.93. This result demonstrates that our impairment function I_R, derived in section III-C, has enough accuracy to predict the total impairment of rendering factors used during the CMR session.

Having established impairment function I_R for rendering factors, in the next section we will combine it with the impairment functions of network and encoding factors to formulate a complete CMR-UE model which can quantitatively measure mobile client's user experience during a CMR session.

IV. DERIVATION OF CLOUD MOBILE RENDERING USER EXPERIENCE MODEL

In [8], we had derived a quantitative model, which considers the impairment of video encoding and wireless network factors, to measure the user experience of Cloud-based Mobile Gaming. In this section, we take a similar approach to develop a Cloud Mobile Rendering User Experience (CMR-UE) model, incorporating the effect of rendering factors in addition to video encoding and wireless network factors. We begin by presenting CMR-UE model and the associated metrics, R and CMR-MOS, followed by the subjective experiments conducted to validate the CMR-UE model.

A. Derivation of Model

Similar to [8], we define a CMR Mean Opinion Score (CMR-MOS) as a way to model and measure CMR-UE. We use the function of ITU-T E-model [10] to model CMR-MOS, because this E-model has a good framework for the transmission planning. Equation 3 is the proposed CMR-UE model, where R-factor is a metric that indicates how good the

TABLE VI. COMPENSATION COEFFICIENTS IN EQUATION (4)

C_{ED}	C_{EL}	C_{ER}	C_{DL}	C_{DR}	C_{LR}
0.4	0.3	0.4	0.2	0.4	0.1

CMR-UE is. R takes value from range [0, 100] (the higher R, the better CMR-UE). CMR-MOS is related with R through non-linear mapping, and it is within the range of [1, 4.5].

$$CMR\text{-}MOS = 1 + 0.035R + 7\times10^{-6}R(R-60)(100-R) \quad (3).$$

In [8], we developed impairment functions for video encoding, I_E, network delay, I_D, and network packet loss, I_L, and used them to define the R-factor in Equation 3. In this paper, we extend R to include the impairments (I_R) of graphic rendering factors. The proposed R-factor function is shown in equation 4, where R is determined by four impairment functions I_R I_E I_D and I_L. Cij is the coefficient used for compensating and adjusting the R value. The values of these Cij coefficients, listed in Table VI, are calculated by performing regression analysis using Equation (4) and the results of subjective test, which is described later in section IV.B.

$$R = 100 - I_E - I_D - I_L - I_R + \sum_{\substack{i,j\in\{E,D,L,R\} \\ i\neq j}} C_{ij} * \sqrt{I_i * I_j} \quad (R>0) \quad (4).$$

The functions I_E, I_L and I_D have been derived and presented in [8], while I_R has been derived in section III. Table VI shows the values for the compensation coefficients used in Equation 4. Therefore we have established a complete CMR-UE model (Equation 3, Equation 4, and Table VI). Next step is to conduct a new set of subjective quality assessment experiments to validate the accuracy of this model.

B. Validation of Cloud Mobile Rendering User Experience Model

We use the same experiment framework as in section III to conduct this new set of subjective experiments. This time, we change multiple factors (encoding, rendering and network factors) simultaneously. The encoding factors (frame rate, video bit rate, etc) and rendering factors (view distance, realistic effect, etc.) are varied at the game server, and the network factors are controlled by the network emulator. Each experiment participant played the game with different encoding, rendering and network factors and gave evaluation on his/her user experience using a CMR-MOS score rating

TABLE VII. CMR-MOS RATING AND "R" VALUES

CMR-MOS	R	Description
4.5	100	Excellent experience , no impairment at all
4.0—4.5	80-100	Minor impairment, will not quit
3.0—4.0	60-80	Noticeable impairment, might quit
2.0—3.0	40-60	Clearly impairment, usually quit
1.0—2.0	0-40	Annoying environment, definitely quit.

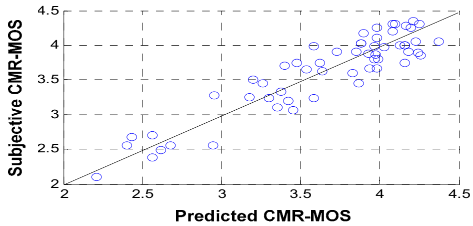

Figure 7: Relation between predicted and subjective CMR-MOS

system shown in Table VII. Table VII has already been presented in our previous work [8] and we need to use the same rating criteria here in this experiment.

For any combination of factors, we can compare the value of subjective CMR-MOS score which is given by participants, with the predicted CMR-MOS score which is computed using Equations 3 and 4. The relation between these two scores are shown in figure 7, we can see that the correlation between predicted CMR-MOS (x-axis) and subjective CMR-MOS (y-axis) is 0.90. This result demonstrates that our CMR-UE model can accurately predict the mobile clients' user experience, under different rendering, encoding, and network factors.

V. APPLICATION OF CMR-UE MODEL: DERIVATION OF OPTIMAL RENDERING SETTINGS

Having derived CMR-UE model and demonstrated its accuracy, in this section, we discuss how the CMR-UE model can be used to study and determine optimal trade-offs between the quality of experience due to rendering, video frame quality, and overall user experience. This can be useful in determining the optimal rendering factors to be used to provide the best user experience depending on the service provider's network.

As we discussed in section II (figure 2), user perceived quality of cloud mobile rendering application (CMR-UE) is mainly based on three categories: graphic rendering factors, video encoding factors, and network conditions. While the network conditions are subjected to the service providers' network which we cannot affect, the rendering settings as well as video encoding settings are under our control. Then a question will arise: what are the optimal rendering and video encoding settings we should use under a certain network condition such that the CMR-UE is maximized?

Given a certain available network bandwidth, the optimal encoding setting would be using Constant Bit Rate (CBR) encoding to make sure the video bit rate is fixed and below the available network bandwidth to avoid network congestion. While the video bit rate is fixed, if we reduce the rendering complexity for instance lowering view distance (impairment of rendering I_R will increase), the content complexity in each video frame will reduce. Thereby the video encoder will have less compression on each video frame, leading to a higher video quality (impairment of encoding I_E will decrease). On the other hand, if we increase rendering complexity (I_R decreases), the video quality will be reduced (I_E increases). Therefore, to decide the optimal rendering settings, we have to

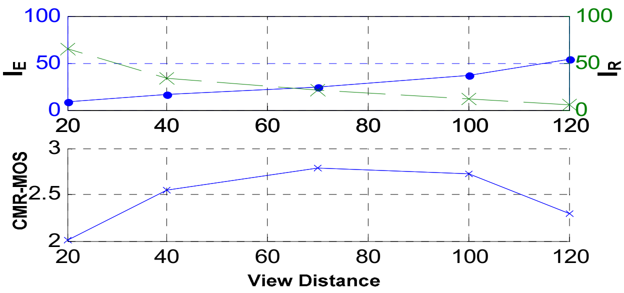

Figure 8: I_E, I_R and CMR-MOS for different view distances used to render and encode a 200kbps CMR video

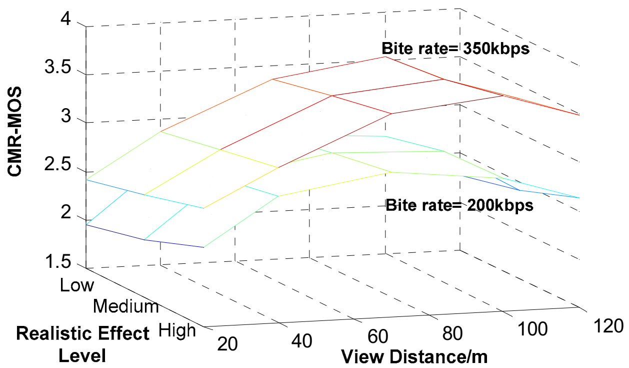

Figure 9: CMR-MOS for different rendering settings in different target video bit rate

understand the relative impacts of rendering quality (I_R) and encoding quality (I_E) on overall CMR-UE.

We next use the CMR-UE model to characterize the effect of changing different rendering factors on the resulting rendering and video quality for different target video bit rates, and consequently on the overall CMR-MOS score. Figure 8 shows results of a representative characterization experiment, for target video bit rate of 200kbps. To find the optimal view distance for 200kbps video streaming, we vary the view distance from 20 meters to 120 meters while computing the I_R, I_E and the CMR-MOS using the CMR-UE model. Figure 9 shows the results – as expected, when view distance increases, I_R reduces, but I_E increases. And because the decreasing slope of I_R is bigger than the increasing slope of I_E, the overall CMR-MOS increases initially with increasing view distance. However, after about 70 meters of view distance, CMR-MOS starts to decrease as the decreasing slope of I_R is smaller than the increasing slope of I_E after that point. Therefore, the optimal view distance when the target video bit rate is 200kbs is 70 meters, where it achieves the maximum CMR-MOS.

It should be also noted that the optimal rendering setting will be different for different network conditions (target video bit rates). Figure 9 shows the simulation results using our CMR-UE model for two rendering settings, view distance and realistic effect, and two network conditions, 200kbps and 350kbps. From Figure 9, we can observe that the optimal view distance for bit rate 200kbps is 70m while it is 120m for 350kbps. For the realistic effect, we can see that a higher

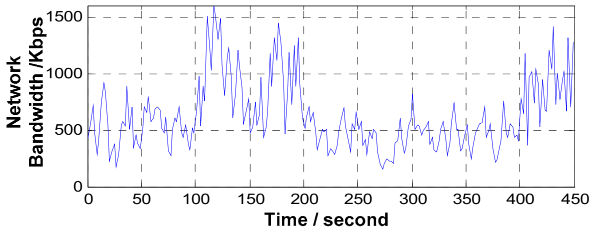

Figure 10: Measurement results of network bandwidth in EVDO network

realistic effect will lead to a higher CMR-MOS score in both bit rates. We can also have the similar mappings like figure 9 to characterize all the rendering settings described in Table II for any CMR application. With those offline characterization, we can find out the optimal rendering settings under any given network condition for any CMR application. Having obtained those optimal rendering settings, in the next section we will describe how they can be used to address the challenges of fluctuating mobile network bandwidth to maximize CMR-UE.

VI. Enhance CMR-MOS in Real Wireless Network

It has been well established that wireless networks are characterized by rapid fluctuations of the network bandwidth experienced by users (we will show a network bandwidth trace captured in a 3G network in figure 10). This challenges the CMR approach, as the rendered and encoded videos streamed from CMR servers to mobile devices can be subject to high and unpredictable delay and packet loss, leading to undesirable increase in response time, and the adverse impact on the quality of the video streamed. The results in section V give us the opportunity to address the above challenge by dynamically changing the rendering settings to the available bandwidth. During the CMR session, we can use a network probing technique [11] to understand what the available network bandwidth is. Then we select the optimal rendering settings for the given network bandwidth from the results in section V, and set the video encoder to Variable Bit Rate (VBR) mode, such that the resulting video bit rate will vary depending on the rendering settings used. By doing this, user experience considering rendering and video quality will be maximized, and moreover, CMR session will not suffer the high packet loss rate and response time due to the network congestion, thereby leading to an optimized CMR-MOS score.

To validate the above approach, we have conducted experiments in a commercial 3G 1x-EVDO network. We again use the Cloud Mobile Gaming application described in section III as a demonstration example, and the selected game is PlaneShift. We set up the CMG server in our lab in the UCSD campus. The video encoding codec is x264, and it is set in VBR mode, with the quantization parameter kept at 25 such that the encoded video quality will not deteriorate user experience. Figure 10 presents a representative sample of EVDO network bandwidth collected by using our network probing technique. It shows that the bandwidth of EVDO network is rapidly fluctuating. While the maximum bandwidth is 1500kbps, it can be dropped to about 200kbps. In such kind of network conditions, we play the CMG game PlaneShift on a

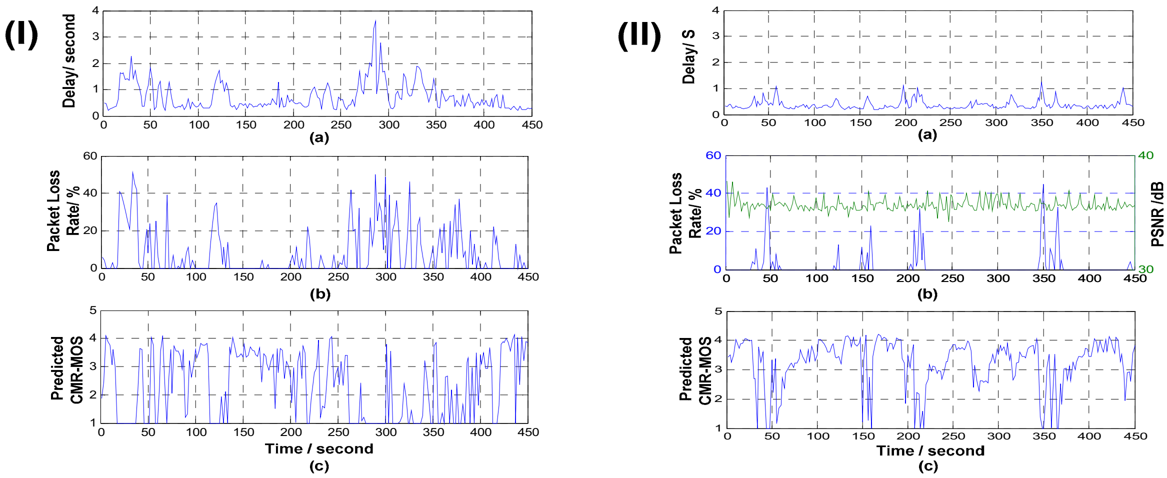

Figure 11: User Experience calculated using CMR-UE model on a representative EVDO network: (I) Using default highest rendering settings (II) Using best rendering settings derived in section V.

mobile device in two different scenarios: using the best (default) rendering settings shown in Table II, and using the optimal rendering settings (derived in section V) which is dynamically updated according to the available network bandwidth captured by network probing technique in real time. The left column in figure 11 presents results using the best rendering settings, while the right column shows results when using the optimal rendering settings. In each column from top to bottom, we sequentially present the results of network round-trip delay, network packet loss rate, PSNR, and resulting CMR-MOS score calculated by our CMR-UE model.

From figure 11, we can observe that if we keep using the best rendering settings, the mobile network link for CMG will be frequently congested, reflected by high and unacceptable network delay and packet loss rate, leading to a very bad CMR-MOS. On the other hand, if we use optimal rendering settings according to the network conditions, the CMR-UE is significantly improved, reflected by the relatively high and stable CMR-MOS score, while only dipping below 3.0 (acceptable user experience threshold) very occasionally. The above data shows how the proposed CMR-UE model, and the optimal rendering settings that can be derived using the model (described in section V), can be used to measure and improve user experience in Cloud Mobile Rendering applications.

VII. Conclusion

In this paper, we have studied a Cloud based Mobile Rendering (CMR) approach which is promising to enable mobile users to experience advanced rendering-based multimedia applications, such as 3D gaming, virtual reality and augmented reality. To assess the user experience while changing the rendering settings, in this paper we have developed a comprehensive CMR-UE model which can characterize simultaneous effects of graphic rendering, video encoding and networking impacts on user experience for a CMR application. With the help of the proposed CMR-UE model, we are able to find the optimal rendering settings for any given network condition. In addition, we describe a method on how to use those optimal rendering settings to address the inherent challenge of rapid fluctuating mobile wireless network. The experiments conducted in real 3G network demonstrate the effectiveness of this method. Therefore, we believe that the CMR-UE model proposed in this paper will serve as a very useful tool for service providers to enable Cloud based Mobile Rendering applications.

References

[1] Richard S. and Gregor S., "Peer-to-Peer Support for Low-Latency Massively Multiplayer Online Games in the Cloud", in *NetGames Workshop 2009,* Paris, France.

[2] Jaewon Ha, Jinki Jung, "Mobile Augmented Reality using Scalable Recognition and Tracking", *IEEE Virtual Reality Conference,* 2011.

[3] Khan,A. and Lingfen Sun, "Video Quality Prediction Model for H.264 Video over UMTS Network and Their Application in Mobile Video Streaming," in *Proc. of IEEE International Conference on Communications(ICC'10),* 2010.

[4] Mukundan V. and Maninak C., "Evaluate Quality of Experience for Streaming Video in Real Time," *in Proc. of IEEE Global Communications Conference (GLOBECOM'09)*, 2009.

[5] M.Ries, O.Nemethova, M.Rupp, "Video Quality Estimation for Mobile H.264/AVC Video Streaming," *Journal of Communications*, Vol.3, pp.41-50, 2008.

[6] W.Van, et al., "Scalable 3D Graphics Processing in Consumer Terminals", *IEEE International Conference on Multimedia and Expo,* 2002.

[7] N.Tack, "3D Graphic Rendering Time Modeling and Control for Mobile Terminals", *in Porc. of Int. Conf. on 3D Web Technology*, 2004.

[8] S. Wang, S. Dey, "Modeling and Characterizing User Experience in a Cloud Server Based Mobile Gaming Approach," *in Proc. of IEEE Global Communications Conference (GLOBECOM'09),* Honolulu, 2009.

[9] S. Wang, S. Dey, "Rendering Adaptation to Address Communication and Computation Constraints in Cloud Mobile Gaming," *in Proc. of IEEE Global Communications Conference (GLOBECOM'10),* Miami, Dec. 2010.

[10] ITU-T Rec. G.107, "The E-model, a computational model for use in transmission planning," Mar.2005.

[11] S. Wang, S. Dey, "Addressing Response Time and Video Quality in Remote Server Based Internet Mobile Gaming," *in Proc. of IEEE Wireless Communication & Networking Conference (WCNC'10),*Sydney, Mar. 2010.