Enhancing Cloud Mobile 3D Display Gaming User Experience by Asymmetric Graphics Rendering

Yao Lu, Yao Liu, Sujit Dey
Mobile Systems Design Lab, Dept. of Electrical and Computer Engineering
University of California, San Diego
{luyao, yal019}@ucsd.edu, dey@ece.ucsd.edu

***Abstract*—With the arrival of auto-stereoscopic 3D displays for mobile devices, and emergence of more 3D content, there is much anticipation for 3D mobile multimedia experiences, including 3D display gaming. Simultaneously, with the emergence of cloud computing, more mobile applications are being developed to take advantage of the elastic cloud resources. In this paper, we explore the possibility of developing Cloud-based 3D Mobile Gaming, where the 3D video rendering and encoding is performed on cloud servers, with the resulting 3D video streamed over wireless networks to mobile devices with 3D displays for a true 3D mobile gaming experience. However, with the significantly higher bit rate requirement for 3D video, ensuring user experience may be a challenge, both in terms of 3D video quality and network delay (response time), considering the bandwidth constraints and fluctuations of wireless networks.**

In this paper, we propose a new asymmetric graphics rendering approach which can significantly reduce the video encoding bit rate needed for a certain video quality, thereby making it easier to transmit the video over wireless network. However, since asymmetric rendering may also impair the graphics quality, we need to be able to understand and measure its impact. We conduct subjective tests to study and model the impairments due to asymmetric rendering and network delay, thereby developing a user experience model for cloud based mobile 3D display gaming. By conducting subsequent subjective tests, we prove the correctness of the impairment functions and the resulting user experience model. We also conduct experiments using real 4G-LTE network profile. Experimental results show that by making use of the user experience model, it is possible to set appropriate graphics rendering parameters according to network constraints, such that the user experience can be maintained to a high level.

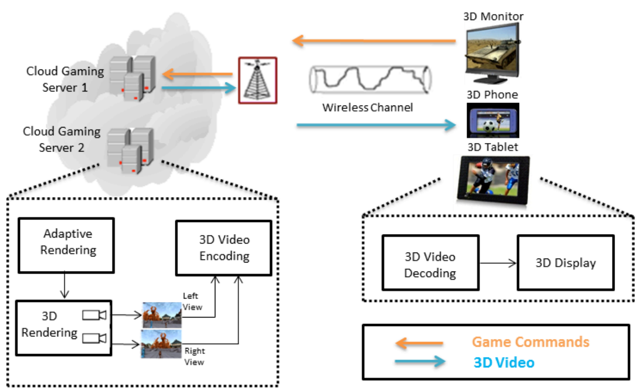

Figure 1. Architecture of Cloud based Mobile 3D Display Gaming System

I. INTRODUCTION

In recent years, 3D display devices such as 3D TV, 3D monitors, 3D tablets, etc. are becoming more and more popular. With the improvements in the quality of these devices, more and more users are starting to enjoy 3D experience offered by them especially for watching 3D videos and playing 3D games. Meanwhile, rapid emergence of smart phones and tablets enables users to play games via mobile devices anywhere anytime. We can see that there is a trend that mobile 3D display gaming will be popular in the future years [6][9][10]. However, among the top-ranked games on mobile devices, especially 3D games (3D rendering with 2D display), most of them need heavy graphics rendering. We can imagine an even bigger challenge for the computational power and battery life of the mobile devices when these games are upgraded to 3D rendering with 3D display in the future. To save computational power and extend battery life on mobile devices but at the same time support high quality 3D gaming (3D rendering with 3D display), Cloud Mobile Gaming (CMG) [1] is a possible solution. Previous work on CMG has focussed on streaming 2D video from the cloud to mobile devices with 2D displays, and therefore cannot provide rich immersive stereoscopic experience. In this paper, we propose Cloud based Mobile 3D Display Gaming architecture (CMG (3D)).

Figure 1 shows the overall architecture and data flow of the proposed CMG (3D) platform. The 3D rendering is performed on cloud servers in response to gaming commands from the mobile device. We place two virtual cameras in the game world to generate a left view and right view of game video. After the two views are generated, they will be encoded and transmitted through wireless network to the mobile device, and displayed on the device 3D screen. On the reverse side, game commands are transmitted from mobile device to the game server through wireless network. In this way, users can play 3D games as if the game is rendered locally.

However, constrained and fluctuating network conditions can create significant challenge for the CMG (3D) architecture described above. Note that for gaming applications, especially those which are highly response time sensitive, transmitting the target video with low latency becomes the key factor which influences user experience [8]. We did an initial study of how large the delay will be under real 4G-LTE mobile network conditions. Figure 2(a) shows a snapshot of a network profile which we captured using iPerf client software on iPhone and iPerf server software on remote CMG server during busy hours at UCSD. It describes how bandwidth is fluctuating during a time period of 300 seconds. Then we use

978-1-4799-2358-8/14/$31.00 ©2014 IEEE

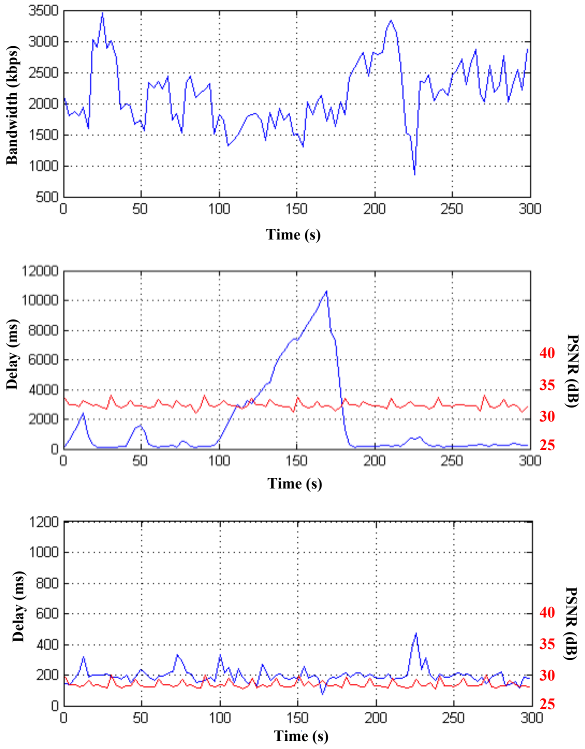

Figure 2. (a) top, Network profile (b) middle, Delay and PSNR performance when using high quality video encoding (c) bottom, Delay and PSNR performance when using low quality video encoding

a network emulator to create a network bandwidth trace exactly as Figure 2(a) shows. Using the emulator and the CMG (3D) platform we developed which runs an online open-source MMORPG game PlaneShift [16], we measured response time which includes upload and download time shown in Figure 2(b). Also shown is the video quality, measured as PSNR. For this kind of MMORPG game, the acceptable delay is 440ms and excellent delay is 120ms [13]. As can be seen from Figure 2(b), the experienced delay can be much higher than acceptable threshold. To solve this delay problem, we can change the video encoding setting and tradeoff video quality and delay. We show in Figure 2(c) that by decreasing the video quality by an average of 4 dB, the experienced delay can be reduced. But the low video quality also impacts user experience greatly. From this example, we can see that it can be challenging to maintain good 3D video quality and response time simultaneously, under constrained and fluctuating wireless network conditions.

Asymmetric 3D video encoding, where the videos of the left and right views are encoded with different bit rates, have been earlier proposed as a way to reduce the 3D video encoding bit rate. Authors of [4, 14, 15] have shown that as long as the primary view is encoded at sufficiently high quality and the auxiliary view is encoded at a quality above some threshold, no noticeable quality degradation will be caused on user perceived 3D video quality. While using asymmetric 3D video encoding may be a way to reduce the bitrate required to stream 3D video from the cloud to the mobile device, and thereby reduce the delay, it may not be enough to bridge the large delay gap seen from Figure 2(b). Our previous research with cloud mobile 2D display gaming has shown that adapting the graphic rendering richness can significantly reduce the encoding bit rate needed for the resulting game video, and hence can enable streaming of high quality 2D video over constrained wireless networks while meeting delay constraints [2, 3]. In this paper, to address the challenge of streaming high quality 3D gaming video with low delay constraint, we propose a new asymmetric graphics rendering approach where the rendering richness of one of the views can be degraded below that of the primary view. We show our proposed approach can significantly reduce the 3D video encoding bit rate needed for a certain video quality, thereby making it possible to transmit the 3D video over wireless network with the desired user experience, that is high video quality while meeting low delay constraint.

However asymmetric graphics rendering technique may cause graphics artifacts, which may also influence user experience. So we first develop a model for Cloud mobile 3D display gaming user experience (CMG (3D)-UE) to quantitatively measure user experience and then using this model, we can show that asymmetric graphics rendering technique can greatly improve user experience.

The rest of the paper is organized as following: Section II introduces technical details about asymmetric graphics rendering and shows how promising it is to enhance user experience. Section III derives and validates the impairment functions for graphics rendering parameters introduced in Section II. Section IV further takes network delay impairment function into consideration and derives a complete CMG (3D)-UE model. Section V presents the result of enhancing user experience based on the CMG (3D)-UE model under real wireless network conditions. Section VI concludes the paper.

II. Asymmetric Graphics Rendering

In this paper, we study two rendering parameters that have great impact on user experience and rendered video bit rate: texture detail and view distance. Texture detail defines the quality of the images on the surface of the objects. View distance determines which objects will be included in the rendered game video frame. For a given view distance value, the game engine will only render the objects which is within the distance from the camera to the object, and all the objects beyond this view distance value will be excluded. Since for 3D display games we generate two views using two virtual cameras (Figure 1), we can potentially render each view with different texture detail and view distance, leading to *asymmetric graphics rendering*. Figure 3(a) shows an example of asymmetric view distance in which left view distance is set as 150 meters and right view distance is set as 100 meters. This means the objects which are more than 100 meters away from the camera will be only rendered in left view but not in right one. Consequently, any object further than 100m (and less than 150m) can only be seen by the viewer's left eye, but not by the right eye. Figure 3(b) shows another example which is called asymmetric texture detail. In this example, left view texture detail is set as high quality and right one is medium quality. This means the image quality in the left eye will be slightly higher than that in the right eye.

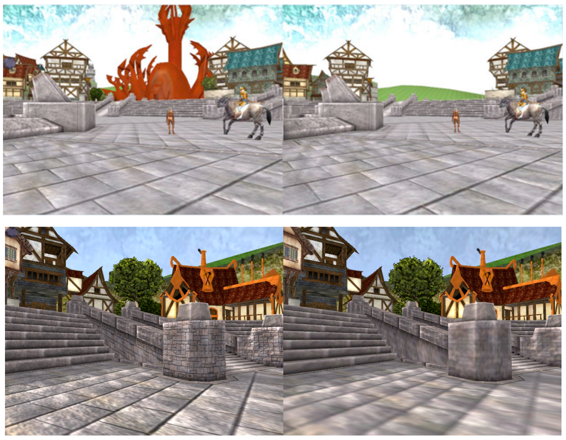

Figure 3. (a) top, left view and right view with asymmetric view distance (b) bottom, left view and right view with asymmetric texture detail

We will show in the following why asymmetric graphics rendering is promising to enhance user experience. Figure 4(a) presents the rate-distortion comparison between different view distance settings and 4(b) shows the rate-distortion comparison between different texture detail settings. By enabling asymmetric graphics rendering, user experience can be potentially increased. For example, the setting that has one view with medium quality and the other with low can have roughly 2dB PSNR gain over the setting where both the views have medium quality under the same encoding bit rate. In this way, user experience can be greatly improved whether under the same network condition (increase video quality) or the same video quality (decrease network delay). However, while asymmetric graphic rendering can clearly enhance the resulting video quality and/or reduce delay, it can also impair the graphics rendering quality, thereby impacting the overall user experience.

After introducing the concept of asymmetric rendering, in the following sections, we will derive impairment functions for the rendering parameters which can quantitatively measure how asymmetric rendering will impact overall user experience.

III. Derivation And Validation Of Impairment Due To Asymetric Rendering

In order to learn how graphics rendering parameters affect CMG (3D)-UE quantitatively, especially when asymmetric graphics rendering is applied, we define an impairment function I_R as Equation (1).

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

I_R indicates the impairment due to rendering. It takes value between 0 and 100. Higher I_R value indicates larger impairment. Similarly, I_{VD} represents the impairment due to view distance, and I_{TD} indicates the impairment due to texture detail. The definition of I_R, I_{VD} and I_{TD} are similar as in [3], but here we take 2 views into consideration. After defining the impairment functions, we conduct subjective experiments to derive and validate the impairment functions.

A. Subjective Experiment Setting

Figure 5 shows the testbed used for the subjective tests. We use a 3D monitor with a laptop to substitute mobile 3D

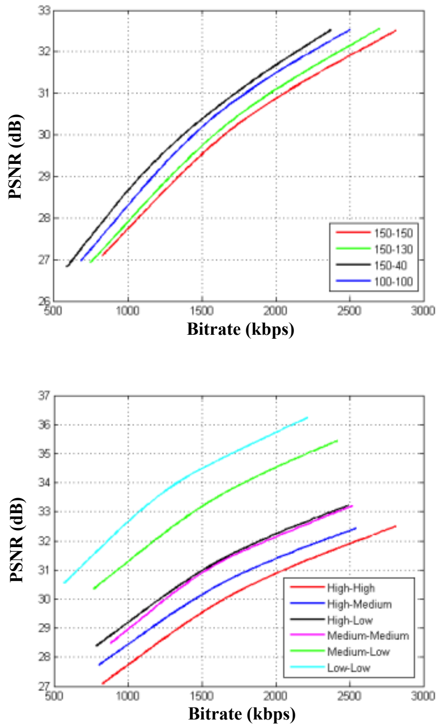

Figure 4. (a) top, Rate-distortion performance with different view distance setting for left view and right view (b) bottom, Rate-distortion performance with different texture detail setting for left view and right view

display devices because off-the-shelf mobile 3D display devices do not have as good quality as 3D monitors up to now. The laptop is connected to a network emulator through a wireless AP. The network emulator is connected to the game server over Internet. By changing the parameters such as bandwidth, delay and packet loss, etc. using the network emulator, we can construct different kinds of network environment. The selected game which runs upon the above framework is an online open source MMORPG game: PlaneShift. To investigate how asymmetric graphics rendering parameters affect user experience, we set the video QP to be 25 which is a high quality and network conditions to be sufficiently large so that only graphics rendering parameters can cause impairment. Then, we invited 16 UCSD students to participate in our subjective experiments. We manually set the graphics rendering parameters according to Table 1 one at a time and the testers are asked to give scores according to the criterion listed in Table II when playing the game. After they are done, we collect data and analyze it to derive the

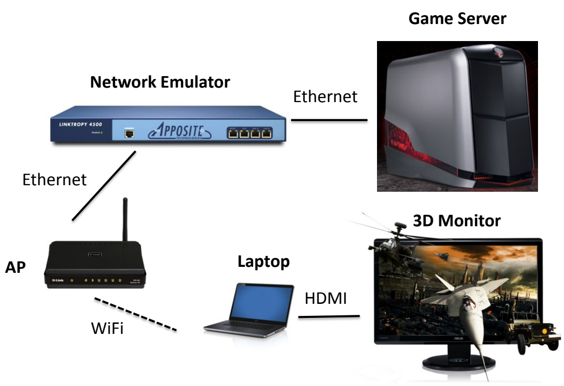

Figure 5. Testbed for Subjective Experiments

impairment functions.

TABLE I. GRAPHICS RENDERING PARAMETER SETTING

Parameters	*Experiment Values*
Texture Detail(Down Sample)	High(0) Medium(2) Low(4)
View Distance(meter)	150 140 130 120 100 80 70 60 40 20 0

TABLE II. 3D GRAPHICS QUALITY AND CTITERION FOR I_R

I_R	*Description*
0	Excellent depth perception, Excellent visual quality, no visual impairment
0-20	Good depth perception, Good visual quality, minor visual impairment, will continue the game
20-40	Acceptable depth perception but noticeable visual impairment, might quit the game
40-60	Can still get feeling of depth perception but clearly visual impairment, usually quit the game
60-100	No feeling of depth at all, unacceptable quality, definitely quit the game

B. Impairment Function Derivation

To derive I_{VD} and I_{TD}, we use the results from the experiments where we only changes one of the graphics rendering parameters while keeping the other at the best value so that only the one which is changing causes impairment.

Impairment to perceived graphic quality is caused in 2 ways: (a) the value of the larger view distance, and (b) the difference of view distance in the two views. The larger value stands for how many objects are missing in both views. The value of difference represents how many objects are seen by one eye but not the other.

We found that the impairment function for I_{VD} is different depending on whether the difference of view distances in two views is zero or not. Figure 6(a) shows the relationship between I_{VD} and view distance value, when two views have the same view distance. We can see the relationship on the left of D_1 is different from right. We use linear regression to get the following equation:

$$I_{VD}=\begin{cases}0 & (d>D_2)\\ \alpha\left[(D_2-d)/(D_2-D_1)\right] & (D_2>d>D_1)\\ \alpha+\beta(D_1-d) & (D_1>d>0)\end{cases} \quad (2),$$

in which D_1 =40, D_2=150, α =35, β =1.625.

Figure 6(b) shows the other case where the difference of view distance in two views is not zero. In this case, the impairment I_{VD} depends on two factors: the view distance difference between left and right view, denoted as *f*; and the larger view distance among the two views, denoted as *d* Notice that there is also a threshold DD_1 here. The relationship on the left of DD_1 is different from right. We get this value by calculating every point and choose the one which has the greatest sum of gradient. We then use linear regression for both sides and derive I_{VD} as following:

$$I_{VD}=\begin{cases}\alpha * f+\theta * d & (DD_1>f>0)\\ 30+\beta(f-DD_1)+\theta * d & (f>DD_1)\end{cases} \quad (3),$$

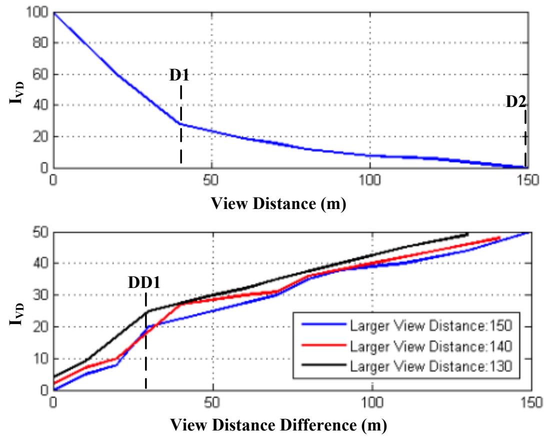

Figure 6. (a) top, Impairment function of view distance when difference of view distance in two views is zero (b) bottom, Impairment function of view distance when difference of view distance in two views is not zero

in which *f* represents difference value and *d* represents larger value. DD_1 = 30, α =1, β =0.25, θ =0.2.

A similar method is used to derive the impairment function of texture detail, I_{TD}, where we vary the texture detail setting of left and right views, and collect participants' judgments. The results are summarized in Table III. Note that since we have only studied 3 options for texture detail, the results of I_{VD} turn out to be a table of discrete values instead of a continuous function. Compared to the I_{TD} function derived for 2D game [3], we can see the impairment is less than 2D version for the same game. It's because even when texture detail in two views are both low, the depth information is still there and the experience is combined with both image quality and depth quality. Also, we can get the following observations:

1. The additional impairment due to asymmetric texture detail of left and right views is nominal (only about 7.5) when the asymmetry is of one level (like High—Medium, or Medium-Low); however, the impairment can go up significantly if the asymmetry is higher (like High-Low)

2. The combination of High-Low causes more impairment than Low-Low, but it will also require more bit rate than Low-Low, so this setting won't be used in our CMG (3D) system.

3. The impairment when texture detail of one of the views is reduced by one level is less than the impairment due to reducing one-level texture detail in both views.

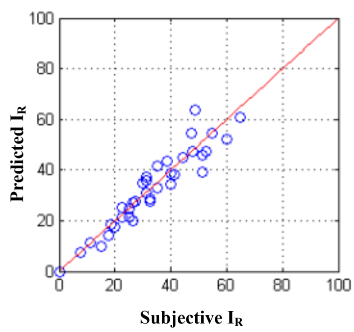

Figure 7. Relationship between predicted and subjective I_R value

C. Impairment Function Validation

In order to validate the functions (Equations (1) (2) (3) and Table III) derived in the previous subsection, we conducted another set of experiments with a new group of 15 participants, playing the same game, and using the same evaluation criterion; however, in this set of experiments, both categories of graphics rendering parameters are changed at the same time. Figure 7 shows the relationship between predicted I_R computed by the derived impairment function (y-axis) and subjective I_R given by human players (x-axis). The correlation is 0.94. This result proves the accuracy of the impairment functions we derived earlier.

TABLE III. DIFFERENT TEXTURE DETAIL COMBINATION AND I_{TD}

Texture Detail Combination	I_{TD}
High-High	0
High-Medium	7.625
High-Low	27.5
Medium-Medium	11.5
Medium-Low	18.75
Low-Low	25

IV. DERIVATION AND VALIDATION OF CLOUD MOBILE 3D DISPLAY GAMING USER EXPERIENCE MODEL

In [1][2][3], the factors affecting Cloud Mobile Gaming User Experience have been analyzed, and a UE Model has been proposed which takes into account the impairments due to the factors. Though the above applies to 2D video streamed and displayed on 2D mobile devices, the category of factors are the same in the 3D case. As is described in [3], there are three major categories of objective factors: graphics rendering parameters, video encoding parameters, and mobile network parameters. As discussed earlier, graphics rendering parameters include texture detail and view distance which affects the user perceived visual quality of the graphics. Video encoding parameters include video quality, video bit rate, etc. which affect both visual quality and response time. Mobile network parameters include packet loss rate and bandwidth which will affect visual quality or response time. In this paper, we will focus on studying graphic rendering parameters and network delay parameter, while keep the encoding parameters at high level such that there will be no impairment caused by video encoding. As for graphics rendering parameters, impairment functions have already been derived in Section III. As for network delay parameter, we want to show in the following that impairment functions are the same as what has been derived for 2D case [1]. After getting impairment functions separately for graphics rendering parameters and network delay parameter, we will derive and validate an overall user experience model for 3D cloud mobile gaming.

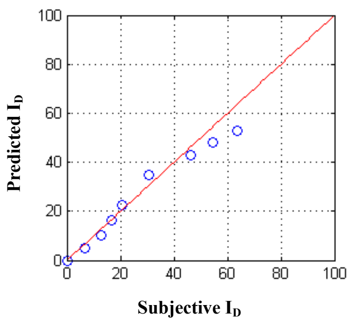

Figure 8. Relationship between predicted and subjective I_D value

A. Impairment Function Validation for Network Delay parameter

In our previous work [1], the impairment caused by network delay for 2D cloud gaming has been studied and the impairment function is as follows:

$$I_D = \begin{cases} 0 & (T_1 > Delay > 0) \\ 40\left[(Delay - T_1)/(T_2 - T_1)\right] & (T_2 > Delay > T_1) \\ 40 + \alpha(Delay - T_2) & (Delay > T_2) \end{cases} \quad (4),$$

in which T_1=120, T_2=440, α =0.05

We validate the accuracy of this function for 3D cloud mobile gaming by conducting subjective tests with the same people who validate the graphics rendering impairment functions using our CMG (3D) testbed shown in Figure 5. The value of network delay parameters are shown in Table IV. Figure 8 shows the relationship between subjective impairment given by people with the objective impairment computed using Equation (4). The correlation between predicted I_D (y-axis) and subjective I_D (x-axis) is 0.98. This high correlation indicates that although Equation (4) is derived for 2D cloud gaming, we can also apply it in 3D cloud gaming. We next derive the UE model based on these two impairment functions.

TABLE IV. NETWORK DELAY PARAMETER SETTING

Parameters	*Experiment Values*
Network delay (ms)	80 120 150 200 250 300 400 500 600

B. CMG (3D)-UE Model Derivation

According to ITU-T E-model [5], we propose to model our CMG (3D)-UE model as Equation (5). Notice that R-factor is a metric indicating how good the user experience is., which takes value from range [0, 100] and can be calculated by Equation (6). CMG (3D)-MOS is related with R through non-linear mapping, and is within the range of [1, 4.5].

$$\text{CMG }(3\text{D}) - \text{MOS} = 1 + 0.035R + 7\times10^{-6} R(R-60)(100-R) \quad (5)$$

$$R = 100 - I_R - I_D + C_{DR}(\sqrt{I_D \times I_R}) \quad (6),$$

in which C_{DR}=0.4.

C. CMG (3D)-UE Model Validation

We conducted a new series of subjective experiments to verify the correctness of the model proposed in the last section, using the same subjects used earlier to validate the impairment functions (Section IIIC). This time we change simultaneously both graphics rendering parameters at the game server, and network delay parameter at the network emulator, and let the testers give their scores according to Table V. For every setting, we use Equations (5) and (6) to compute our predicted UE score and compare it with the subjective UE score given by the testers. Figure 9 shows our result. We can see from the figure that the correlation between predicted UE score and ground truth UE score is 0.93, indicating adequate accuracy of our CMG (3D)-UE model.

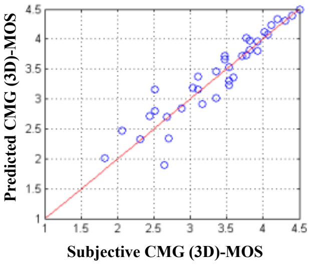

Figure 9. Relationship between predicted and subjective CMG (3D)-MOS

TABLE V. 3D GRAPHICS QUALITY AND CTITERION FOR IR

CMG (3D)-MOS	*Description*
4.5	Excellent depth perception, Excellent visual quality, no visual impairment
4.0-4.5	Good depth perception, Good visual quality, minor visual impairment, will continue the game
3.0-4.0	Acceptable depth perception but noticeable visual impairment, might quit the game
2.0-3.0	Can still get feeling of depth perception but clearly visual impairment, usually quit the game
1.0-2.0	No feeling of depth at all, unacceptable quality, definitely quit the game

V. ENHANCING CMG (3D) USER EXPPERIENCE

As have been discussed before, meeting the strict response time requirements of cloud mobile gaming can be challenging, in particular for 3D Cloud Mobile Gaming where 3D video has to be streamed from cloud servers to 3D mobile devices. As shown by the experiments reported in Figure 2 in Section I, we can potentially reduce video encoding bitrate to achieve lower network delay, but this compromises the 3D video quality and thereby the user experience. In this section, we show how we can utilize the CMG (3D)-UE model we developed in Section IV to dynamically select the optimal rendering settings for the left and right views according to network conditions, thereby meeting response time requirements and enhancing overall user experience.

Due to the dynamic nature of wireless network conditions, our proposed technique will work in a real-time manner, i.e., we will decide on the optimal rendering setting at the end of each time interval to be applied to the next time interval. During a cloud mobile gaming session, at each time t, the total experienced delay will consist of 3 parts, server delay caused by graphics rendering and video encoder tasks, network delay due to bandwidth constraint, and propagation delay indicating the natural delay from a source to a destination depending on geographic distance and transmission medium. Equation (7) shows this relationship.

$$D(t) = D_N(t) + D_S(t) + D_P(t) \quad (7),$$

where $D_N(t)$ stands for network transmitting delay at time t, $D_S(t)$ represents server (computation) delay and $D_P(t)$ means propagation delay. In our work, we assume encoding delay and propagation delay are constant and set their value to be: $D_S(t)$=0.06s and $D_P(t)$=0.02s at any time. All the delay components are in units of second. Equation (8) and (9) shows how $D_N(t)$ is calculated.

$$D_N(t) = (S_P(t) + S_C(t)) / BANDWIDTH(t) - 1 \quad (8),$$

$$S_P(t+1) = S_P(t) + S_C(t) - BANDWIDTH(t) \quad (9),$$

in which $S_C(t)$ stands for current generated data size and $S_P(t)$ represents data size of previous accumulated data in the streamer buffer. For every time period, current data whose size is $S_p(t)$ will firstly be queued in streamer buffer. However, the buffer will have some existing data whose size is $S_C(t)$. As shown in Equation (8), it will take $D_N(t)$ seconds for this new generated data to be streamed out. Considering time interval of 1 second, we have one second duration to transmit, and hence there is a minus 1 term in the expression of $D_N(t)$.

The proposed asymmetric rendering adaptation technique consists of the following 3 steps:

a) Collect heuristically how much video bit rate will be generated for each rendering setting. This corresponds to the $S_C(t)$ term in Equation (8). These values can be collected offline, and only need to be collected once.

b) During a cloud gaming session, keep sending network probing packets [1] to monitor the network condition, and keep estimating the network bandwidth value for each time interval (one second for the experiments reported in this paper). This corresponds to the term *BANDWIDTH(t)* in Equation (8)(9).

c) During a cloud gaming session, calculate the theoretical delay (using Equations (7)(8)(9)). and Use the delay value and different possible rendering settings in the UE model to find the rendering setting that leads to the highest CMG (3D)-MOS value. Use this rendering setting in the graphic engine for the next time interval.

We have implemented the proposed CMG (3D)-UE model as well as the asymmetric graphics rendering technique in the CMG (3D) platform shown in Figure 1. Figure 10 shows the

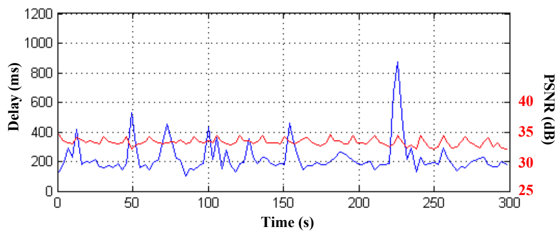

Figure 10. Delay and PSNR performance by setting best graphics rendering parameters

delay and PSNR performance when we use the proposed technique to perform asymmetric 3D rendering and 3D video streaming for the wireless network profile shown in Figure. 2(a) (for the same game Planeshift used in section I). We can see that the delay achieved mostly meets the 440ms acceptable response time threshold for this game, in contrast with the unacceptable delay achieved by the original CMG (3D) system as reported in Figure 2(b). At the same time, a high video quality (PSNR) can be maintained, in contrast to the alternative approach of using lower bit rate for 3D video encoding to reduce delay, as reported in Figure 2(c). Figures 11 (a) and (b) show the CMG (3D)-MOS scores corresponding to the performances reported in Figure 2(b) (without asymmetric rendering) and Figure 10 (with asymmetric rendering) respectively, the latter incorporating impairments due to asymmetric graphic rendering. As can be seen, in spite of impairments caused by asymmetric rendering, application of our proposed technique utilizing the CMG (3D)-UE model is able to ensure a high and stable user experience under the fluctuating wireless network conditions, in contrast to the original system where user experience can be severely compromised.

VI. CONCLUSION

In this paper, we have proposed an asymmetric rendering technique for cloud mobile 3D display gaming, in order to address the bandwidth fluctuation challenge of wireless network. In order to study the feasibility of this asymmetric rendering technique, we have also developed a user experience model to quantitatively measure the user experience, including impairments due to asymmetric graphics rendering and network delay. Extensive subjective experiments have been carried out to derive and validate this model. The experiment results show that our model correlates well with human observers' experiences. Moreover, based on the CMG (3D)-UE model, we propose a technique to dynamically select the optimal graphics rendering settings, asymmetric for each of the views, according to the network condition. Last but not least, we show that by applying our technique proposed above, we can keep UE score to be constantly high by dynamically selecting the optimal asymmetric rendering settings.

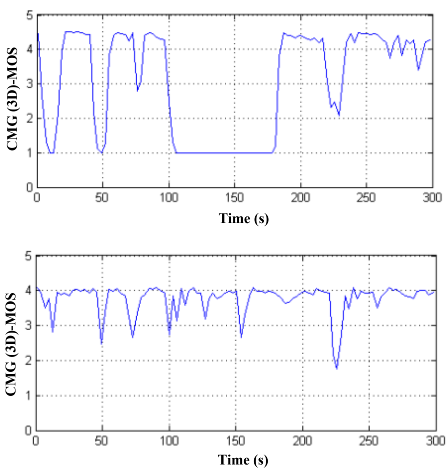

Figure 11. (a) top, CMG (3D)-MOS for Figure 2(b) (b) bottom, CMG (3D)-MOS for Figure 10

The proposed asymmetric graphics rendering technique can be used for other cloud based 3D rendering applications besides 3D gaming. In the future, we will develop an algorithm which can automatically select the optimal rendering parameters according to the network conditions. Secondly, we will study how to combine asymmetric 3D encoding technique together with asymmetric graphics rendering technique to further enhance user experience under given network constraints.

VII. ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant No. CCF-1160832.

VIII. REFERENCES

[1] S. Wang and S. Dey, "Cloud mobile gaming: modeling and measuring user experience in mobile wireless networks," *ACM SIGMOBILE Mobile Computing and Communications Review* 16.1 (2012): 10-21.

[2] S. Wang and 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.

[3] Y. Liu, S. Wang and S. Dey, "Modeling, Characterizing, and Enhancing User Experience in Cloud Mobile Rendering," *in Proc. of IEEE International Conference on Computing, Networking and Communications* (ICNC'12), Maui, Jan, 2012.

[4] G. Saygili, C. Gurler and A. Tekalp, "Quality assessment of asymmetric stereo video coding," *in Proc. of IEEE International Conference on Image Processing* (ICIP'10), Hong Kong, Sep, 2010.

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

[6] A. Ojala and P. Tyrvainen, "Developing Cloud Business Models: A Case Study on Cloud Gaming," *IEEE Software,* 28.4 (2011): 42-47.

[7] M. Jarschel, D. Schlosser, S. Scheuring, T. Hoßfeld, "An evaluation of QoE in cloud gaming based on subjective tests," *in Proc. of IEEE International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing* (IMIS'11), 2011.

[8] K. T. Chen, Y. C. Chang, P. H. Yseng, C. Y. Huang and C. L. Lei, "Measuring the latency of cloud gaming systems." *in Proc. of the 19th ACM international conference on Multimedia*(MM'11), 2011.

[9] J. Gaudiosi, "Future of Cloud Gaming: Industry Leaders' Thoughts." *FC Business Intelligence* (2011).

[10] P. E. Ross, "Cloud Computing's Killer App: Gaming." *IEEE Spectrum,* 46.3 (2009): 14-14.

[11] T. Hobfeld, R. Schatz, M. Varela and C. Timmerer, "Challenges of QoE management for cloud applications," *IEEE Communications Magazine*, *50*(4), 28-36.

[12] 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.

[13] S. Wang and 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.

[14] G. Saygili, C. G. Cihat, and A. M. Tekalp. "Evaluation of asymmetric stereo video coding and rate scaling for adaptive 3D video streaming," *IEEE Transactions on Broadcasting* 57.2 (2011): 593-601.

[15] F. Lu, H. Wang, X. Ji, and G. Er, "Quality assessment of 3D asymmetric view coding using spatial frequency dominance model," *IEEE 3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video, 2009* (pp. 1-4).

[16] http://www.planeshift.it/