GPU Consolidation for Cloud Games: Are We There Yet?

Hua-Jun Hong¹, Tao-Ya Fan-Chiang¹, Che-Run Lee¹, Kuan-Ta Chen², Chun-Ying Huang³, Cheng-Hsin Hsu¹

¹Department of Computer Science, National Tsing Hua University
²Institute of Information Science, Academia Sinica
³Department of Computer Science and Engineering, National Taiwan Ocean University
What’s Cloud Gaming?

• Run the games on server, stream the videos to clients side.

• Who’s providing this kind of service?

(image source: CiiNOW)
Who’s Providing These Service?

• OnLive was estimated to worth 1.8 billion in 2011
• Gaikai was sold to Sony for 380 million in 2012

Why would I want to use this service?
Merits Of Cloud Gaming

• Play the most advanced game on any device, anywhere, any time.
  – No tedious installation
  – No need to buy expensive components
  – Continue the gaming experience on any device

This looks amazing, why can’t I see it everywhere?

(image source: Gaikai)
Challenges of Cloud Gaming

• Inadequate bandwidth
• Low latency requirement
• Resource virtualization
• …
Challenges of Cloud Gaming

• Inadequate bandwidth
• Low latency requirement
• Resource virtualization
• ...


Challenges of Cloud Gaming

• Inadequate bandwidth
• Low latency requirement
• **Resource virtualization**
  – CPU
  – Network
  – GPU
  – ...
• ...
• ...

Challenges of Cloud Gaming

• Inadequate bandwidth
• Low latency requirement
• Resource virtualization
  – CPU
  – Network
  – GPU
  – ...
• ...
• ...
Restructuring of OnLive

• OnLive laid off all of it’s employees in Aug 2012
  – “We didn't go bankrupt, we didn't shut down, we just restructured”
  – Sold for 4.8 million(Just 1/375 of the once estimated 1.8 billion)

How did this happened?
Lack of GPU Virtualization

• One of the reasons: no GPU virtualization
  – One physical GPU for very few gamers (as few as 1)
  – Low utilization, high operating expense

GPU virtualization, which is still considered experimental is critical to cloud gaming
ARE MODERN GPUS READY?
Outline

• Motivation
• Methodology
• Experiments Results
• Conclusion
Outline

• Motivation
• Methodology
• Experiments Results
• Conclusion
Testbed Setup
Two Type Of Experiments

• GPU-only
  – Focus on the performance of GPU

• End-to-end experiment
  – Evaluate cloud gaming platform with GamingAnywhere
GPU-Only Experiments

Fig. 1: Our testbed consists of a cloud gaming server running multiple GA servers and games, a dummynet router, and several GA clients.
Fig. 1: Our testbed consists of a cloud gaming server running multiple GA servers and games, a dummynet router, and several GA clients.
The Server

Fig. 1: Our testbed consists of a cloud gaming server running multiple GA servers and games, a dummynet router, and several GA clients.
The Server

• Runs XenServer 6.2
  – A patched CentOS
  – Access VMs with XenCenter
• 2 Intel Xeon E5 2.1 GHz 12-core CPU
  – 6 Physical core each with 2 HyperThreading
• 64 GB memory
• Allocate 1 CPU core and 2GB RAM to Hypervisor (Dom0)
  – The rest are equally divided among the VMs
Fig. 1: Our testbed consists of a cloud gaming server running multiple GA servers and games, a dummyernet router, and several GA clients.
GPU Virtualization Type

• Software-based (vSGA)
  – More compatible and flexible
  – Arbitrary number of clients

GPU Virtualization Type (Cont.)

• Pass-through
  – Better performance
  – Further classification
    • One-to-one fixed pass-through (PassThrough)
    • One-to-many mediated pass-through (vGPU)

GPU Virtualization (Cont.)

• Shea and Liu studied the performance of PassThrough in 2013
  – Poor performance in Xen, KVM
  – Partially due to excessive context switch
  – Some of the observations don’t apply to modern GPU
• Our main focus is on vGPU
  – Never been measured in literature
• We quantify the performance of GPUs shared by multiple VMs
  – Shea and Liu [Netgames ‘13] focus on the comparisons between bare-metal and one virtual machine

What GPUs did you test?
Tested GPU

- **NVIDIA Quadro 6000**
  - Released in 2010
  - 1 instance, support
    - PassThrough
    - vSGA

- **NVIDIA Grid K2**
  - Released in 2013
  - 2 independent instances, support
    - PassThrough
    - vSGA
    - vGPU\(_2\), vGPU\(_4\), vGPU\(_8\)
      - Only enable one instance in paper

Different generation, modern K2 supports vGPU\(_x\)
Virtual Machines

Fig. 1: Our testbed consists of a cloud gaming server running multiple GA servers and games, a dummynet router, and several GA clients.
Guest VM

- Windows 7 64bit Enterprise as guest OS
- Runs
  - Game or Benchmark
  - GamingAnywhere server (When doing end-to-end experiments)
Fig. 1: Our testbed consists of a cloud gaming server running multiple GA servers and games, a dummynet router, and several GA clients.
Workload Generators

• Game
  – Fear 2: Project Origin
  – Lego Batman: The Videogame
  – Limbo

• Benchmark
  – Sanctuary
  – Cadalyst

• TinyTask
  – Used to ensure fairness in experiments

Diverse Genre
Workload Generators - Game

• Fear 2: Project Origin (2009)
  – 3D First Person Shooter
Workload Generators - Game

• Lego Batman: The Videogame (2008)
  – 3D Action-adventure
Workload Generators - Game

• Limbo (2010)
  – 2D Side-scroller
Workload Generators - Benchmark

• Sanctuary
  – Overall benchmark
Workload Generators - Benchmark

• Cadalyst
  – Detailed 2D, 3D benchmark
  – 4 different types of test for both 2D and 3D
Maintaining Fairness in Experiments

• We vary conditions in our experiments one at a time
  – Different players’ behavior may incur different workloads
  – “How do you maintain the fairness?”

• We use TinyTask to record keyboard and mouse inputs
  – Replay them to perform the exact same player behavior
Fig. 1: Our testbed consists of a cloud gaming server running multiple GA servers and games, a dummynet router, and several GA clients.
End-to-end Test with GamingAnywhere

Fig. 1: Our testbed consists of a cloud gaming server running multiple GA servers and games, a dummynet router, and several GA clients.
Components

Fig. 1: Our testbed consists of a cloud gaming server running multiple GA servers and games, a dummynet router, and several GA clients.
Fig. 1: Our testbed consists of a cloud gaming server running multiple GA servers and games, a dummynet router, and several GA clients.
What’s GamingAnywhere?

• GamingAnywhere is the first open-source cross-platform cloud gaming platform
• Supports Windows, Linux, Mac OS X, Android
• Official website: http://gaminganywhere.org/
• Github page: https://github.com/chunying/gaminganywhere
Fig. 1: Our testbed consists of a cloud gaming server running multiple GA servers and games, a dummynet router, and several GA clients.
Dummynet

- We add a FreeBSD machine with dummynet to emulate diverse network conditions
  - Bandwidth
  - Delay
  - Packet loss
Performance Metrics

• CPU utilization
  – The CPU utilization of Hypervisor and VMs

• GPU utilization

• Context switch
  – The context switch of XenServer

• Frame Per Second
  – The most common indicator of game performances
  – Low FPS imply low gaming experience
Performance Metrics

- **CPU utilization**
  - The CPU utilization of Hypervisor and VMs
- **GPU utilization**
- **Context switch**
  - The context switch of XenServer
- **Frame Per Second**
  - The most common indicator of game performances
  - Low FPS imply low gaming experience
Performance Metrics

• CPU utilization
  – The CPU utilization of Hypervisor and VMs

• GPU utilization

• Context switch
  – The context switch of XenServer

• Frame Per Second
  – The most common indicator of game performances
  – Low FPS imply low gaming experience
Performance Metrics

• CPU utilization
  – The CPU utilization of Hypervisor and VMs

• GPU utilization

• Context switch
  – The context switch of XenServer

• Frame Per Second
  – The most common indicator of game performances
  – Low FPS imply low gaming experience
Performance Metrics

• CPU utilization
  – The CPU utilization of Hypervisor and VMs

• GPU utilization

• Context switch
  – The context switch of XenServer

• Frame Per Second
  – The most common indicator of game performances
  – Low FPS imply low gaming experience
Performance Metrics (Cont.)

• Frame loss rate on GA client
  – Frame loss caused by network condition
• PSNR(Peak Signal-to-Noise Ratio)
  – Indicator of the picture quality observed at client side
• SSIM(Structural Similarity)
  – Indicator of the picture quality observed at client side
• Response delay
  – The time between the input from player and reaction of the game
  – High response delay incur frustration in gaming experience
Performance Metrics (Cont.)

• **Frame loss rate** on GA client
  – Frame loss caused by network condition
• **PSNR** (Peak Signal-to-Noise Ratio)
  – Indicator of the picture quality observed at client side
• **SSIM** (Structural Similarity)
  – Indicator of the picture quality observed at client side
• **Response delay**
  – The time between the input from player and reaction of the game
  – High response delay incur frustration in gaming experience
Performance Metrics (Cont.)

• Frame loss rate on GA client
  – Frame loss caused by network condition
• PSNR (Peak Signal-to-Noise Ratio)
  – Indicator of the picture quality observed at client side
• SSIM (Structural Similarity)
  – Indicator of the picture quality observed at client side
• Response delay
  – The time between the input from player and reaction of the game
  – High response delay incur frustration in gaming experience
Performance Metrics (Cont.)

• Frame loss rate on GA client
  – Frame loss caused by network condition

• PSNR(Peak Signal-to-Noise Ratio)
  – Indicator of the picture quality observed at client side

• SSIM(Structural Similarity)
  – Indicator of the picture quality observed at client side

• Response delay
  – The time between the input from player and reaction of the game
  – High response delay incur frustration in gaming experience
Performance Metrics (Cont.)

• Frame loss rate on GA client
  – Frame loss caused by network condition
• PSNR(Peak Signal-to-Noise Ratio)
  – Indicator of the picture quality observed at client side
• SSIM(Structural Similarity)
  – Indicator of the picture quality observed at client side
• Response delay
  – The time between the input from player and reaction of the game
  – High response delay incur frustration in gaming experience

How do you measure these metrics?
Measurement Utilities

- Fraps – FPS of foreground window
- Sar – Context switches
- Xentop – CPU utilization of hypervisor and VMs
- Nvidia-smi – GPU utilizations under vGPU
- GPU-Z – GPU utilizations under PassThrough, vSGA
  - Nvidia-smi can’t report GPU utilization in PassThrough and vSGA mode
  - Use GPU-Z in guest OS
Measurement Utilities

- **Fraps** – FPS of foreground window
- **Sar** – Context switches
- **Xentop** – CPU utilization of hypervisor and VMs
- **Nvidia-smi** – GPU utilizations under vGPU
- **GPU-Z** – GPU utilizations under PassThrough, vSGA
  - Nvidia-smi can’t report GPU utilization in PassThrough and vSGA mode
  - Use GPU-Z in guest OS
Measurement Utilities

• Fraps – FPS of foreground window
• Sar – Context switches
• Xentop – CPU utilization of hypervisor and VMs
• Nvidia-smi – GPU utilizations under vGPU
• GPU-Z – GPU utilizations under PassThrough, vSGA
  – Nvidia-smi can’t report GPU utilization in PassThrough and vSGA mode
  – Use GPU-Z in guest OS
Measurement Utilities

- Fraps – FPS of foreground window
- Sar – Context switches
- **Xentop** – CPU utilization of hypervisor and VMs
- Nvidia-smi – GPU utilizations under vGPU
- GPU-Z – GPU utilizations under PassThrough, vSGA
  - Nvidia-smi can’t report GPU utilization in PassThrough and vSGA mode
  - Use GPU-Z in guest OS
Measurement Utilities

- Fraps – FPS of foreground window
- Sar – Context switches
- Xentop – CPU utilization of hypervisor and VMs
- **Nvidia-smi** – GPU utilizations under vGPU
- **GPU-Z** – GPU utilizations under PassThrough, vSGA
  - Nvidia-smi can’t report GPU utilization in PassThrough and vSGA mode
  - Use GPU-Z in guest OS
Outline

• Motivation

• **Methodology**

• Experiments Results

• Conclusion
Outline

• Motivation
• Methodology
• Experiments Results
• Conclusion
Motivation

Methodology

Experiments Results
- The edge of vGPU over vSGA
- Overhead of context switches
- vGPU may outperform PassThrough
- Consolidation overhead
- Importance of hardware codec
- Performance under diverse network condition
- Response delay in real world

Conclusion
Outline

• Motivation
• Methodology

• Experiments Results
  – The edge of vGPU over vSGA
  – Overhead of context switches
  – vGPU may outperform PassThrough
  – Consolidation overhead
  – Importance of hardware codec
  – Performance under diverse network condition
  – Response delay in real world

• Conclusion
The Edge of vGPU over vSGA

• Compare the newly supported mediated pass-through(vGPU) and software-based vSGA

• We expect vGPU to be better than vSGA
The Edge of vGPU over vSGA (Cont.)

- K2 outperforms Quadro 6000 up to 3.46
- Way better scalability for K2
- No longer consider Quadro 6000(and vSGA) in the rest of the paper

TABLE II: Achieved frame rates on two considered GPUs

<table>
<thead>
<tr>
<th># of VMs</th>
<th>Quadro 6000</th>
<th>K2</th>
<th>Speed-up (times)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 VMs</td>
<td>22.3</td>
<td>25.1</td>
<td>1.13</td>
</tr>
<tr>
<td>4 VMs</td>
<td>13.1</td>
<td>32.3</td>
<td>2.47</td>
</tr>
<tr>
<td>8 VMs</td>
<td>7.0</td>
<td>24.2</td>
<td>3.46</td>
</tr>
</tbody>
</table>
Outline

• Motivation
• Methodology

• Experiments Results
  – The edge of vGPU over vSGA
  – Overhead of context switches
  – vGPU may outperform PassThrough
  – Consolidation overhead
  – Importance of hardware codec
  – Performance under diverse network condition
  – Response delay in real world

• Conclusion
Overhead of Context Switches

• We expect poor performance when the number of context switches grows higher
  – According to the previous mentioned paper
Overhead of Context Switches (Cont.)

• No longer dominant in the performance
  – More context switches does not mean lower FPS
  – Instead, it’s proportional to FPS

<table>
<thead>
<tr>
<th>Game</th>
<th>FPS vGPU₈</th>
<th>FPS vGPU₄</th>
<th>Ratio</th>
<th>No. Context Switches vGPU₈</th>
<th>No. Context Switches vGPU₄</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fear2</td>
<td>45.8</td>
<td>64.9</td>
<td>0.7</td>
<td>9472</td>
<td>14149</td>
<td>0.67</td>
</tr>
<tr>
<td>Batman</td>
<td>43.3</td>
<td>41.6</td>
<td>1.04</td>
<td>4325</td>
<td>3991</td>
<td>1.08</td>
</tr>
<tr>
<td>Limbo</td>
<td>39.2</td>
<td>64.8</td>
<td>0.6</td>
<td>10700</td>
<td>13927</td>
<td>0.76</td>
</tr>
</tbody>
</table>

TABLE IV: Relation Between FPS and Number of Context Switches
Overhead of Context Switches (Cont.)

• No longer dominant in the performance
  – More context switches does not mean lower FPS
  – Instead, it’s proportional to FPS
  – Different from the earlier study by Shea and Liu

<table>
<thead>
<tr>
<th>Game</th>
<th>vGPU₈</th>
<th>vGPU₄</th>
<th>Ratio</th>
<th>vGPU₈</th>
<th>vGPU₄</th>
<th>Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fear2</td>
<td>45.8</td>
<td>64.9</td>
<td>0.7</td>
<td>9472</td>
<td>14149</td>
<td>0.67</td>
</tr>
<tr>
<td>Batman</td>
<td>43.3</td>
<td>41.6</td>
<td>1.04</td>
<td>4325</td>
<td>3991</td>
<td>1.08</td>
</tr>
<tr>
<td>Limbo</td>
<td>39.2</td>
<td>64.8</td>
<td>0.6</td>
<td>10700</td>
<td>13927</td>
<td>0.76</td>
</tr>
</tbody>
</table>

TABLE IV: Relation Between FPS and Number of Context Switches
Outline

• Motivation
• Methodology
• Experiments Results
  – The edge of vGPU over vSGA
  – Overhead of context switches
  – vGPU may outperform PassThrough
  – Consolidation overhead
  – Importance of hardware codec
  – Performance under diverse network condition
  – Response delay in real world
• Conclusion
vGPU May Outperform PassThrough

• Comparison between vGPU and PassThrough
vGPU May Outperform PassThrough

• Comparison between vGPU and PassThrough
• One-to-one fixed pass-through (PassThrough) should outperform One-to-many mediated pass-through (vGPU) in every workload generator
  – Since it’s a whole GPU dedicated to one VM
vGPU May Outperform PassThrough

• However, this is not the case
vGPU May Outperform PassThrough

• However, this is not the case **WHY?**
Detailed Benchmark with Cadalyst

- vGPU outperforms PassThrough in all 2D and part of 3D tests
Detailed Benchmark with Cadalyst

- vGPU outperforms PassThrough in all 2D and part of 3D tests
- Sharing a GPU among multiple VMs is now a reality for the right type of games.

(b) 2D benchmark scores.

(c) 3D benchmark scores
Outline

• Motivation
• Methodology

• Experiments Results
  – The edge of vGPU over vSGA
  – Overhead of context switches
  – vGPU may outperform PassThrough
  – Consolidation overhead
  – Importance of hardware codec
  – Performance under diverse network condition
  – Response delay in real world

• Conclusion
Consolidation Overhead

- Use vGPU$_8$
- Increase the number of VMs up to 8
- Goal: Verify the scalability of K2 GPU
Consolidation Overhead

• Limbo and Batman do not suffer from overhead
  – Obvious consolidation overhead for Fear2 and Sanctuary
Consolidation Overhead

- Limbo and Batman do not suffer from overhead
  - Obvious consolidation overhead for Fear2 and Sanctuary  **WHY?**
Resource Usage

- Fully loaded time of resources under 8 VMs
  - Sanctuary and Fear2 are bounded by GPU
  - Batman and Limbo are not bounded
Resource Usage

- Fully loaded time of resources under 8 VMs
  - Sanctuary and Fear2 are bounded by GPU
  - Batman and Limbo are not bounded

More detailed analysis on these two applications
Fear2 Consolidation Overhead

- Frame per second and GPU utilization
Dynamically Resource Allocation

- Under the same vGPU mode, fewer VM means higher FPS and lower GPU utilization

- Dynamically allocate resources among all VM
FPS-aware GPU Scheduling

- Even when GPU utilization is not saturated
  - vGPU₈ never exceeds 48 FPS, 66 FPS for others.
  - FPS-aware GPU Scheduling Algorithm implemented
Sanctuary Consolidation Overhead

- Same observation can be made on the result from Sanctuary
Sanctuary Consolidation Overhead

- Dynamically allocate resources among all VMs

Higher FPS and Lower GPU Utilization
Sanctuary Consolidation Overhead

- FPS-aware GPU Scheduling Algorithm implemented

![Graph showing FPS and GPU Utilization with various VM configurations.](image-url)
Observations

• Under the same vGPU mode, fewer VMs means higher FPS and lower GPU utilization
  – Dynamically allocate resources among all VM
• Even when GPU utilization is not saturated
  – vGPU₈ never exceeds 48 FPS
  – Others never exceeds 66 FPS
  – FPS-aware GPU scheduling
Observations

• Under the same vGPU mode, fewer VMs means higher FPS and lower GPU utilization
  – Dynamically allocate resources among all VM
• Even when GPU utilization is not saturated
  – vGPU₈ never exceeds 48 FPS
  – Others never exceeds 66 FPS
  – FPS-aware GPU scheduling
• K2 is highly scalable
• vGPU is suitable for sharing GPUs among VMs
Outline

• Motivation
• Methodology

• Experiments Results
  – The edge of vGPU over vSGA
  – Overhead of context switches
  – vGPU may outperform PassThrough
  – Consolidation overhead
  – Importance of hardware codec
  – Performance under diverse network condition
  – Response delay in real world

• Conclusion
End-to-end experiment with GA

- Pass-through and vGPU$_2$
- 8 vCPU cores for the VM
- The outcome is less ideal for high-quality cloud gaming
End-to-end experiment with GA

- Pass-through and vGPU\textsubscript{2}
- 8 vCPU cores for the VM
- The outcome is less ideal for high-quality cloud gaming  

**WHY?**
Importance Of Hardware Codec

- GA server relies on CPUs for real-time video encoding.
- Limitation of free version of XenServer and Win7 combination.
  - Uses up to two CPU for each VM, the rest stays idle

![Diagram](image_url)

Never exceeds 200%
Importance Of Hardware Codec

- GA server relies on CPUs for real-time video encoding.
- Limitation of free version of XenServer and Win 7 combination.
  - Uses up to two CPU for each VM, the rest stays idle
- Leverage hardware codec is an attractive option
Outline

• Motivation
• Methodology
• Experiments Results
  – The edge of vGPU over vSGA
  – Overhead of context switches
  – vGPU may outperform PassThrough
  – Consolidation overhead
  – Importance of hardware codec
  – Performance under diverse network condition
  – Response delay in real world
• Conclusion
Performance Under Diverse Network Conditions

• Test end-to-end performance with different network condition using dummynet
  – Number of clients: 1, 2, 4, 8
  – Delay: 0, 25, 50, 100, 200 ms
  – Bandwidth: 10, 15, 20, 40 Mbps
  – Packet Loss Rate: 0, 0.05, 0.1, 0.5%

• Refer to paper for detailed results

• Network condition clearly effect the end-to-end performance
  – Our platform gives stable performance
Outline

• Motivation
• Methodology

• Experiments Results
  – The edge of vGPU over vSGA
  – Overhead of context switches
  – vGPU may outperform PassThrough
  – Consolidation overhead

  – Importance of hardware codec
  – Performance under diverse network condition
  – Response delay in real world

• Conclusion
Response Delay in Real World

- Goal: measure the difference between native environment and our cloud gaming platform
Response Delay In Real World

- Testbed between Taiwan and California
  - RTT is around 140 ms
- Measure response delay by pressing ESC in-game then find the first frame with pop-up menu from recorded in-game video
- Response delay of our server is close to native environment

<table>
<thead>
<tr>
<th>Game</th>
<th>Platform</th>
<th>Delay (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limbo</td>
<td>XenServer</td>
<td>250 260 265 275 310</td>
</tr>
<tr>
<td></td>
<td>Native</td>
<td>250 255 265 265 270</td>
</tr>
<tr>
<td>Fear2</td>
<td>XenServer</td>
<td>250 290 314 350 355</td>
</tr>
<tr>
<td></td>
<td>Native</td>
<td>250 254 280 300 305</td>
</tr>
<tr>
<td>Batman</td>
<td>XenServer</td>
<td>215 220 230 250 260</td>
</tr>
<tr>
<td></td>
<td>Native</td>
<td>225 235 240 250 260</td>
</tr>
</tbody>
</table>
Outline

• Motivation
• Methodology
• Experiments Results
• Conclusion
Outline

• Motivation
• Methodology
• Experiments Results
• Conclusion
Conclusion

• We have found that modern mediated pass-through GPU virtualization is suitable for sharing among multiple VMs
  – Outperform dedicated GPU in some cases
  – Scalable, can support multiple VMs

• CPUs may become the bottleneck
  – Leveraging hardware codecs → future work
Conclusion

• Evaluate the end-to-end performance of GamingAnywhere in both dummynet testbed and live Internet
  – Stable performance under diverse network conditions
  – Small response overhead, close to native environment
Try It Yourself!

- Official Website: http://gaminganywhere.org
BACKUP SLIDES
Different Number of Client

• At least 18 FPS.
• PSNR is always higher than 21, 32 and 42 db for Batman, Fear2 and Limbo.
Different Delay

• FPS and SSIM is always higher than 25 and 0.9
Different Bandwidth

- Fairly consistent after bandwidth exceeds 8Mbps

<table>
<thead>
<tr>
<th></th>
<th>FPS</th>
<th>PSNR</th>
<th>SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limbo</td>
<td>26.86</td>
<td>42.61</td>
<td>0.9887</td>
</tr>
<tr>
<td>Fear2</td>
<td>28.49</td>
<td>32.08</td>
<td>0.9038</td>
</tr>
<tr>
<td>Batman</td>
<td>42.14</td>
<td>27.67</td>
<td>0.8737</td>
</tr>
</tbody>
</table>
Different Packet Loss Rate

• Quality drops as the packet loss rate rises.
• Frame loss rate are 4.97%, 5.12% and 6.88% under 0.5% packet loss rate.