

A MEASUREMENT STUDY OF OCULUS 360 DEGREE VIDEO STREAMING

MMSys'17 ,

Chao Zhou, Zhenhua Li, and Yao Liu

INTRODUCTION

- Oculus's
- encoding scheme
- adaptive streaming mechanism

3the offset cubic projection

- Pixels from spherical surface to six cube faces

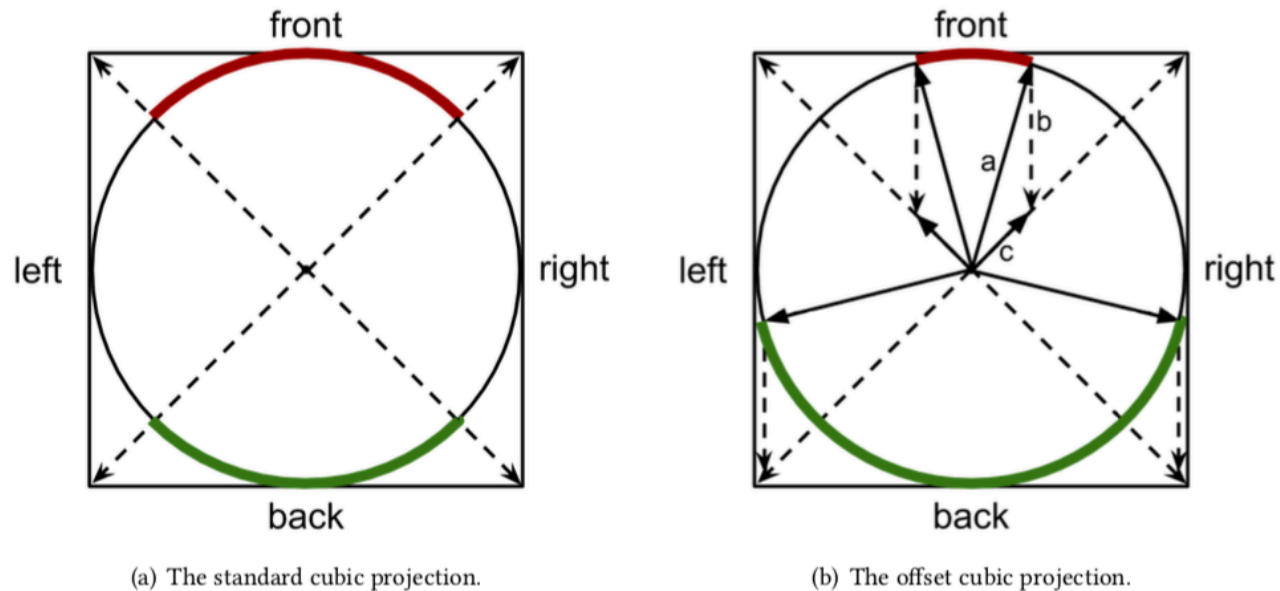


Figure 3: In the offset cubic projection, vector a is a unit vector pointing to a pixel in the standard sphere. Vector b points in the opposite direction of the offset cube's orientation. Vector c is $a + b$. The intersection of vector c and the surface of the cube is the projection destination of the pixel at vector a . The red portion of the circle indicates the portion of the sphere mapped to the offset cube's front face. The green portion of the circle indicates the portion of the sphere mapped to the offset cube's back face.

top	back	bottom
left	front	right

(a) Arrangement of the six cube faces.



(b) The standard cubic projection.



(c) The offset cubic projection.

Offset Cube in Oculus

- the orientation of the offset cubic projection == user's view orientation
- Encode using 22 offset cube orientation

Pitch	Yaw
90	0
45	15, 105, 195, 285
0	0, 30, 60, 90, 120, 150, 180, 210, 240, 270, 300, 330
-45	15, 105, 195, 285
-90	0

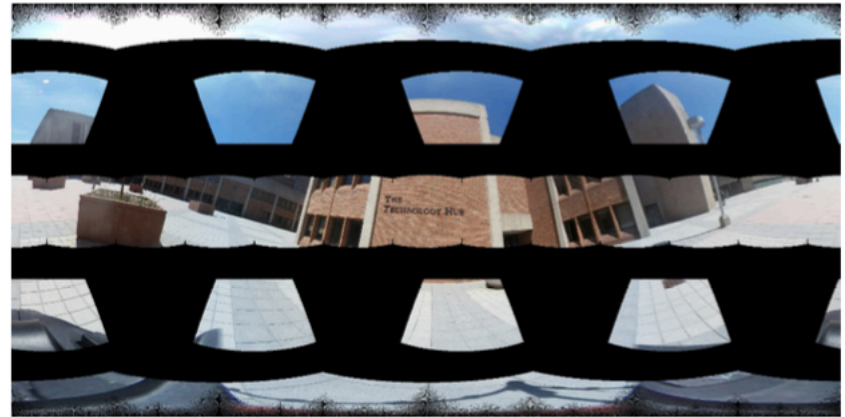


Figure 4: Drawing the front faces of all 22 offset cubes on to an equirectangular image.

- 4 quality levels

Quality	Frame Resolution	Bitrate (bps) Range
272w	1088 × 816	1,789,736 to 2,648,917
400w	1600 × 1200	6,290,250 to 9,613,871
528w	2112 × 1584	9,556,146 to 15,291,141
656w	2624 × 1968	13,512,541 to 22,261,091

- The resolution of each cube face

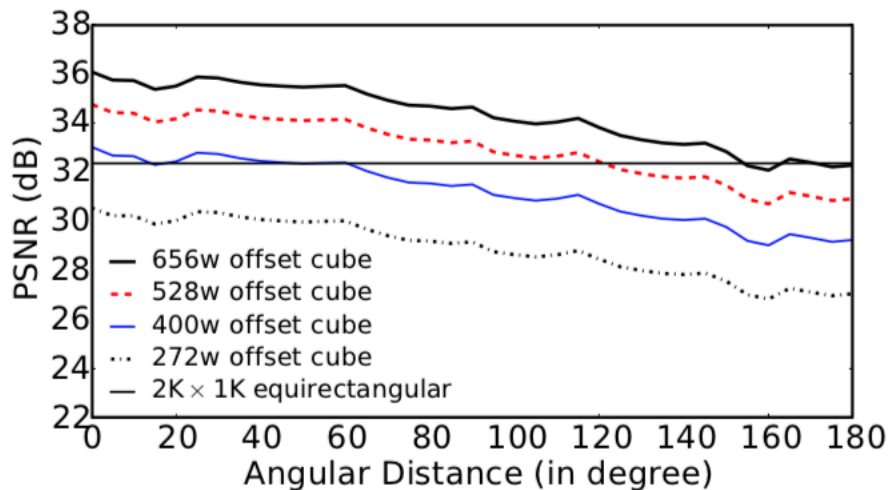
visual quality produced by the offset cubic projection

- views generated from the offset cube representations at standard resolutions used by Oculus against views generated from the original 8K equirectangular video frames.
- high quality equirectangular frames $>$ any views generated from the highest quality Oculus offset cubic projections
- ->
- Front face = 30 degree
- Highest = 656 x 656
- $656 \times 360/30 = 7872$
- 8K is enough

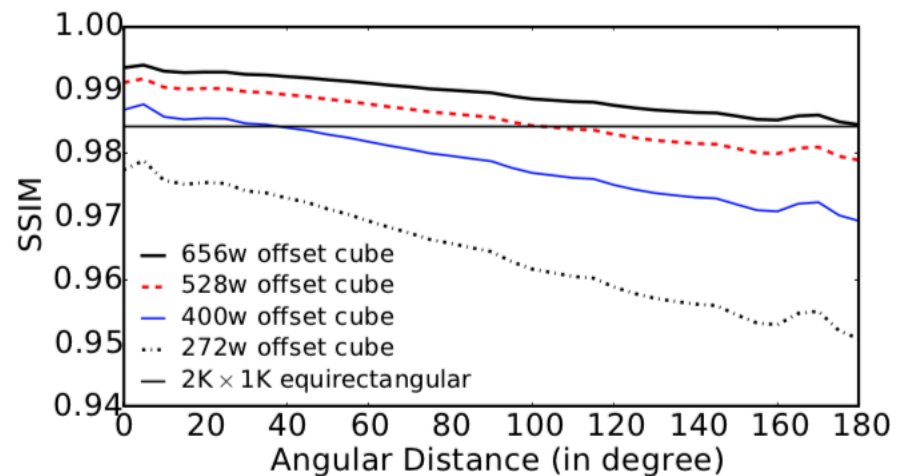
- In our comparison against reference images, we compute the visual quality between the displayed frames generated in the orientation θ_{view} from the offset cube (θ_{offset}, q) against reference images generated in the θ_{view} orientations from an original high quality equirectangular image
- View: 96 x 96 degree, 2000 x 2000 resolution
- PSNR
- SSIM

- Pitch $[-90,90]$,every five degree
- Yaw $[0,360)$,every five degree
- -> total 2664 view orientation

- the angular distance between θ_{offset} and θ_{view} (0~180 degree)



(a) PSNR



(b) SSIM

Adaptive Streaming in Oculus

- Segment, representation, MPD(URL, bitrate, resolution)
- Oculus extends the time-centric adaptation to allow streaming algorithms to not only select differing bitrates over time, as in standard DASH, but also to select higher or lower bitrates for different areas of the 360 degree view.
- (1) Which of the 22 different orientated offset cubes will perform best view for user?
- (2) Which quality level should be chosen?

- one Period, two Adaptation Sets (video+audio)
- The audio Adaptation Set contains only one Representation. The video Adaptation Set, on the other hand, contains 88 Representations

```

1 <ns0:MPD xmlns:ns0="urn:mpeg:dash:schema:mpd:2011" maxSegmentDuration="PT0H0M4.992S" mediaPresentationDuration="PT0H4M43.115S"
  minBufferTime="PT1.500S" profiles="urn:mpeg:dash:profile:isoff-on-demand:2011,http://dashif.org/guidelines/dash264" type
  ="static">
2 <ns0:Period duration="PT0H4M43.115S">
3   <ns0:AdaptationSet FBProjection="offset_cubemap" lang="und" maxFrameRate="30" maxHeight="1968" maxWidth="2624" par="4:3"
  segmentAlignment="true" subsegmentAlignment="true" subsegmentStartsWithSAP="1">
4     <ns0:Representation FBExpand_coef="1.025" FBIs_stereoscopic="true" FBOffcenter_x="0" FBOffcenter_y="0" FBOffcenter_z="
  -0.7" FBPitch="0" FBQualityClass="uhd" FBQualityLabel="2160p" FBRoll="0" FBYaw="30" bandwidth="20592721" codecs="
  avc1.640033" frameRate="30" height="1968" id="dash_sve360_qf_656w_crf_18_high_5.1_p13_30yaw_0pitch_frag_1_videod"
  mimeType="video/mp4" sar="1:1" startWithSAP="1" width="2624">
5       <ns0:BaseURL>https://video.xx.fbcdn.net/...</ns0:BaseURL>
6       <ns0:SegmentBase FBFirstSegmentRange="4338-10514" indexRange="922-4337" indexRangeExact="true">
7         <ns0:Initialization range="0-921" />
8       </ns0:SegmentBase>
9     </ns0:Representation>
10    ...
11  </ns0:AdaptationSet>
12 </ns0:Period>
13 </ns0:MPD>

```

Figure 7: MPD document of an Oculus 360-degree video on Facebook.

- 88 .mp4 files
- Sidx box at the beginning of each .mp4 file
- To request a segment for a specific Representation, the streaming player can analyze the sidx box to determine byte range of segment within the .mp4 file
- One second per segment
- =27~31 frames each segment

Experiment Setup

- the S7 is installed inside the GearVR
- The Oculus application comes pre-installed on S7

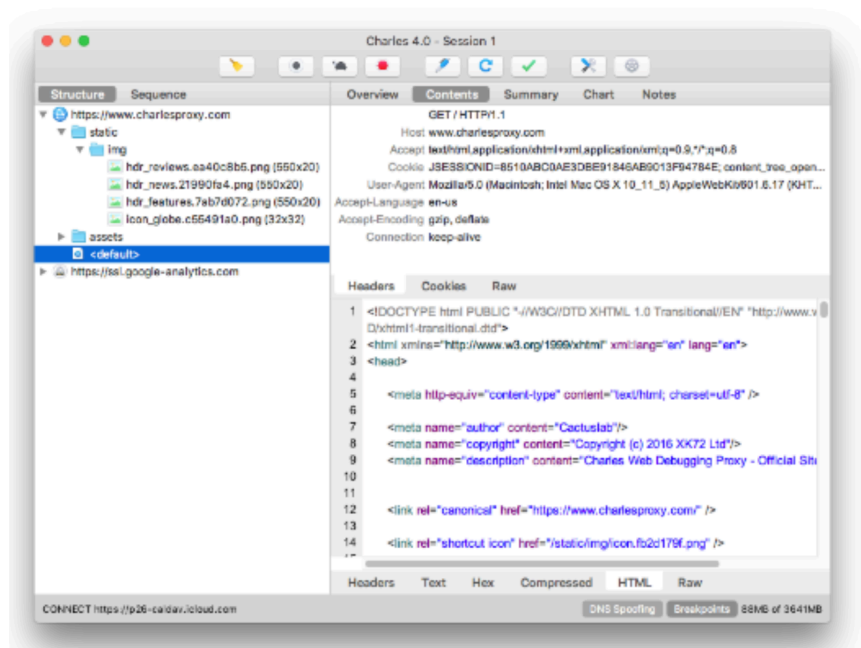


Figure 8: Our VR testbed consists of a GearVR, an S7, a Pan/Tilt mount, a 3D-printed holder for the mount, a Raspberry Pi to accurately control the mount's motion, and a tripod for stabilizing the test instruments.

- **VR testbed**
- ->
- two separate Hitec HS-422 servo motors (yaw, pitch)
- an Adafruit 16-channel PWM servo board

• Network measurement

- The Oculus application transmits all traffic through HTTPS
- -> hard to inspect
- Set up a man-in-the-middle proxy, Charles proxy



Oculus Streaming

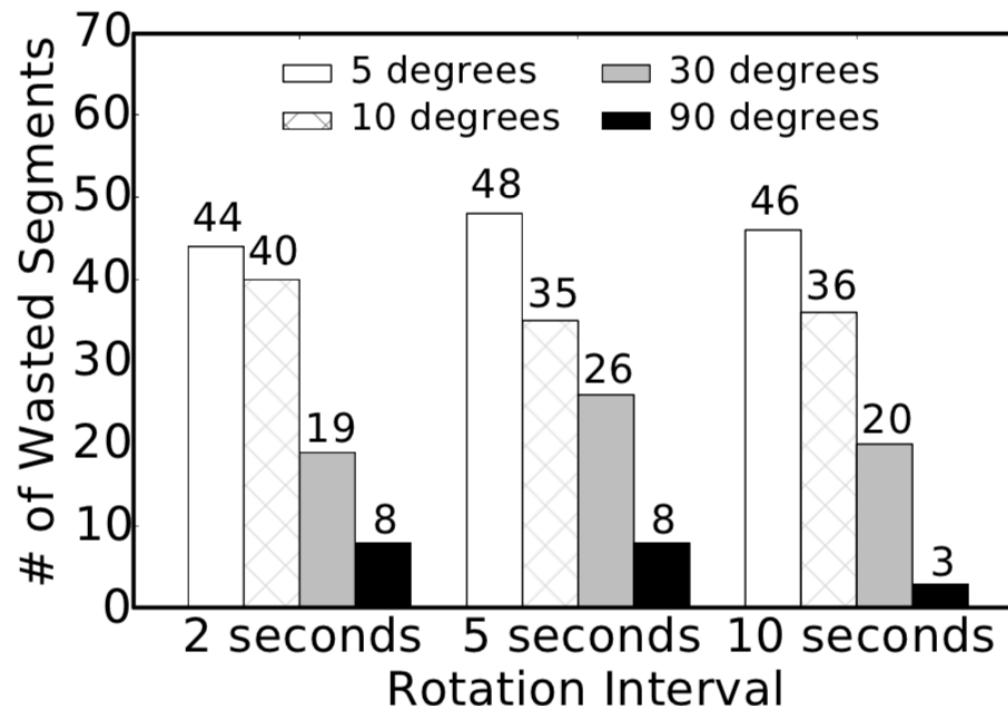
- Metadata downloading
 - The initial segment(required by DASH to initiate decoder) and the sidx segment
 - Using byte range in MPD to recognize these two segment
 - Download metadata of “all” representation (88) beforehand
 - 88 HTTPS requests -> 88 TCP connections
 - 1.345 sec -> startup delay increases
- Video segment downloading
 - Single TCP connection
- Playback buffer filling
 - Powered-off
 - Another five seconds

Streaming Adaptation and Wasted Segments

- Segments in the buffer never showed to user -> wasted segments
- (1) fixed quality level
- (2) fixed orientation
- (3) real user experiments.

- Fixed quality level
- 4.5 Mbps (>272w, <400w)
- Fix one servo motor-> pitch is always 0

- Every 2, 5, 10 sec
- Rotate 5, 10, 30, 90 degrees (0~180)



- Fixed orientation & real user
- Yaw=0, pitch=0 - >so that it will download offset cubes in only one single orientation

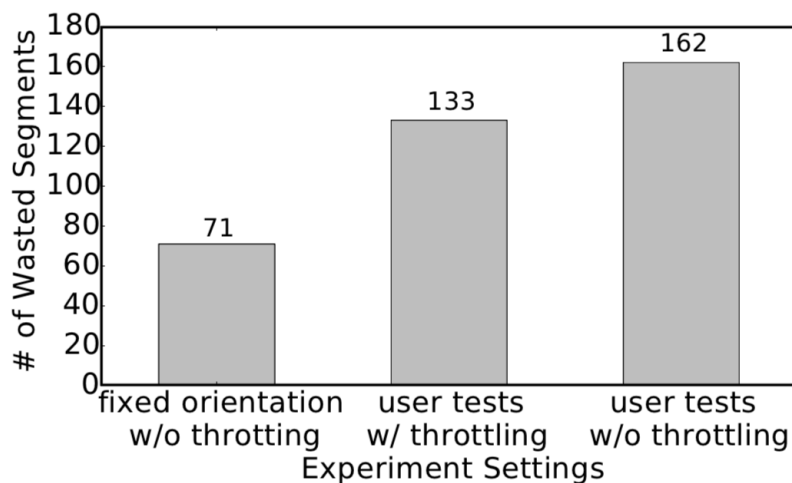


Figure 11: Number of wasted segments under fixed orientation tests and real user tests. The total number of segments in our test video is 282.

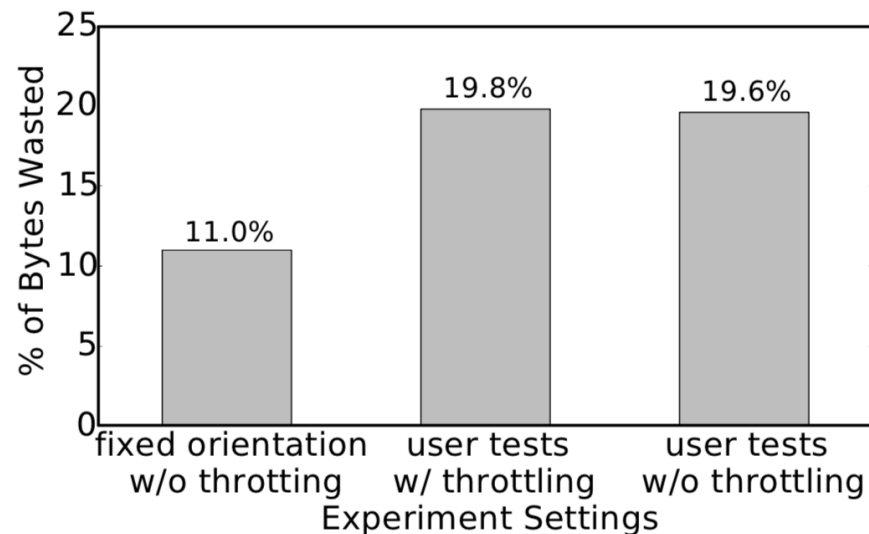


Figure 12: Percentage of downloaded bytes wasted under fixed orientation tests and real user tests.

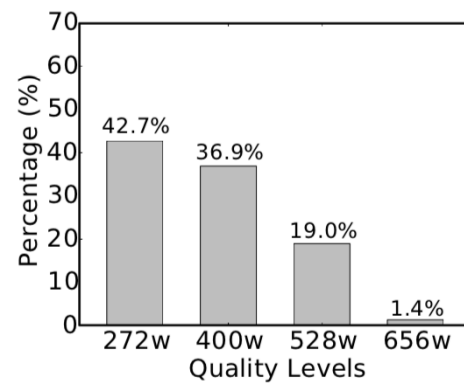


Figure 13: Percentage of segments downloaded at each quality level in fixed orientation tests.

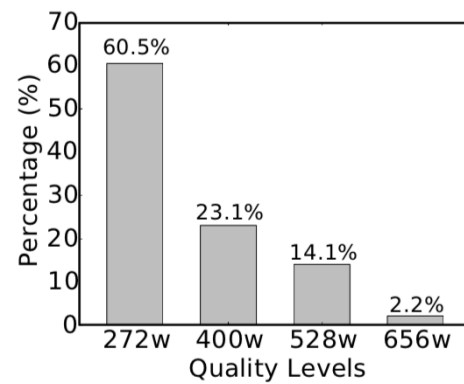


Figure 14: Percentage of segments downloaded at each quality level in real user tests.