

1. Video Clips Properties

Clip Name	Code	File Size	Resolution	Length	Data Rate
Water.mp4	H.264	2.94 MB	1920 * 1080	2 s	11915 kbps
Blood.mp4	H.264	662 KB	1920 * 1080	1 s	5332 kbps
Escalator.mp4	H.264	1.24 MB	1920 * 1080	2 s	3434kbps
Spin.mp4	H.264	275 MB	1920 * 1080	<1 s	2163 kbps
Cara.mp4	H.264	4.00 MB	1280 * 720	10 s	3049 kbps
Spin1.mp4	H.264	278 MB	1280 * 720	<1 s	2277 kbps
Spin5.mp4	H.264	1.35 MB	1280 * 720	2 s	3795 kbps
Water1.mp4	H.264	957 KB	1280 * 720	2 s	3903 kbps

2. Testing on Computer

Computer Properties

System	64-bit Windows 10 Enterprise
Processor	Intel(R) Xeon(R) CPU E5-1650 0 @ 3.20GHz 3.20GHz
Installed RAM	32.0 GB
Graphics	AMD FirePro W5000 (FireGL V) Graphics Adapter

Test Results

Ind ex	Test Mode	Mode	Manually Prepare	FPS	Delay
1	1* Water + 12* Cara	Object	F	75~95	Slight
2	1* Water + 12* Cara	Object	T	80~95	Obvious
3	1* Water + 12* Spin	Object	F	5~10	Slight
4	1* Water + 12* Escalator	Object	F	5~15	Obvious
5	1* Water + 12* Spin1	Object	F	20~30	Sometimes
6	1* Water + 12* Spin5	Object	F	80~90	Slight
7	1* Water + 12* Water	Object	F	Almost 0	Unplayable
8	1* Water + 12* Water1	Object	F	10~15	Obvious
9	1* Water + 12* Blood	Object	F	5~10	Serious
10	1* Water + 6* Water	Object	F	80~90	Slight
11	1* Water + 6* Water	Object	T	80~90	Obvious
12	1* Water + 9* Water	Object	F	10~50	Obvious
13	1* Water + 12* Spin1	Image	T	20~30	Obvious
14	1* Water + 12*Water	Image	T	Almost 0	Unplayable
15	1* Water + 6* Water	Image	T	80~90	Obvious
16	1* Water + 1* Water + 1* Water1 + 1* Spin + 1* Spin1 + 1* Cara + 1* Escalator	Object	F	90	Invisible
17	1* Water + 2* Water + 2* Water1 + 2* Spin + 2* Spin1 + 2* Cara + 2* Escalator	Object	F	20~40	Slight

3. Testing on Laptop

Computer Properties

System	64-bit Windows 10 Pro
Processor	Intel(R) Core(™) i5-3337U CPU @ 1.80GHz 1.80GHz
Installed RAM	4.0 GB
Graphics	Intel(R) HD Graphics 4000

Test Results

In de x	Test Mode	FPS	CPU	Memory	Observation
1	1* Water + 3* Cara	-	>80%	500 M	Unplayable. Computer caton. Hard to exit. Stuck for about 10 seconds then play a few frames.
2	3* Cara	80~100	80%	325 M	Smooth. No caton or delay observed.
3	1* Spin+ 2* Cara	50~80	77%	395 M	Cara smooth with slight caton. Spin sometimes caton and delay
4	1* Water + 2* Cara	50~90	80%	395 M	Cara smooth with some caton. Water seriously caton
5	1* Water + 1* Spin	50~80	75%	395 M	Spin smooth with some delay. Water obvious delay.
6	1* Water	75~85	61%	270 M	Very smooth. No caton or delay observed.
7	1* Water + 1* Cara	70~90	78%	323 M	Cara smooth, no caton or delay observed. Water obvious delay.
8	2* Cara	70~80	65%	263 M	Smooth.

4. Observation

a. Videos have to be of MP4 format in H.264 coding

Unity doesn't have full support for all video formats and/or codings. H.264 or VP8 codings are suggested for Windows. For Android and IOS, the resolution is limited below 640*360 and 1280*720.

There must be other limitations that are not included in the official documentation. When I tried .mov videos, they cause serious crash bug.

b. Unity video player has (unresolvable) delay. Auto loop has shorter delay

Playing videos has unexpected delay even if preparing videos ahead of time. My test also shows it.

Video can be either prepared automatically with the "loop" mode, or prepared manually by scripts. According to the testing results above, the FPS of both ways are almost the same. But the delay before every playing is obviously longer for manually control.

It is not normal, because "prepare" function should get the video player ready for playing next video.

It should be a bug of Unity. Information on Unity forum shows that this bug has not been fixed at least until 2017.3.

c. Streaming to texture vs. override material: tie

In the testing results, "Image" means the video is rendered as a video texture, while "Object" means the video overrides the material of object. There's no visible difference in FPS for both methods.

d. Video length influences FPS (and delay): longer is better

Spin5.mp4 is a video contains 5 rounds of Spin1.mp4. The testing results 4 and 5 shows that longer video has higher FPS (and shorter delay. But the influence in delay is not obvious and could be observation error or caused by other factors.)

e. Resolution influences FPS

The testing result 2 and 4 shows that lower resolution can achieve higher FPS. Here notice that the Data Rate of Spin.mp4 and Spin1.mp4 are almost the same, so it should be a result of lower resolution (fewer pixels) than less data.

f. Data rate influences FPS (and delay)

The testing result on Water.mp4 and Blood.mp4 clearly show that data rate largely influences both FPS and delay.

g. The number of video players vs. video clips

Testing result 14 and 15 shows that, the number of video clips (how many clips are used) has little influence in FPS or delay. While more video players (more video playing at the same time) influences FPS.

Let me know if more performance testings are needed.

- Lotus