APPENDIX

GPU-Accelerated MPEG-5 LCEVC Video Encoding with NVIDIA Ada Generation

Command Lines for Visual Quality (VQ) Tests

The table below reports the ffmpeg command lines used for the visual quality experiments, for both tune low latency (LL) and ultra-high-quality (UHQ).

NVENC HEVC, P4	
LL	ffmpeg -y -f rawvideo -pix_fmt yuv420p -s {source resolution} -r {source fps} -i {source.yuv} -g -1 -rc cbr -b:v {target bitrate}k -c:v hevc_nvenc -tune ll -preset p4 -bufsize {4xtarget bitrate/fps}k -profile main -fps_mode passthrough -unidir_b 1 -multipass qres -no-scenecut true -split_encode_mode 0 -s {encoded resolution} output.ts
UHQ	ffmpeg -y -f rawvideo -pix_fmt yuv420p -s {source resolution} -r {source fps} -i {source.yuv} -b:v {target bitrate}k -c:v hevc_nvenc -tune uhq -preset p4 -bufsize {target bitrate}k -fps_mode passthrough -s {encoded resolution} output.ts
LCEVC NVENC HEVC, P4	
LL	ffmpeg -y -f rawvideo -init_hw_device 'vulkan=vulkan:NVIDIA L40S,drm_modifiers=0' -pix_fmt yuv420p -s {source resolution} -r {source fps} -i {source.yuv} -filter_complex '[0]hwupload[input_in_gpu_memory]' -map '[input_in_gpu_memory]' -g -1 -b:v {target bitrate}k -c:v lcevc_hevc_vulkan -fps_mode passthrough -base_encoder nvenc_hevc -eil_params "preset=4;gpu_device=NVIDIA;lcevc_tune=low_latency;multi_pass=two-pass-quarter;tuning_info=low-latency;lcevc_preset=3;m_hf_modulation=0.35;encoding_upsample=modifiedcubic;split_encode_mode=disabled;uni_directional_b=1;rc_base_buffer_size={150%*(4xtarget bitrate/fps)};rc_pcrf_base_rc_mode=cbr" output.ts
UHQ	ffmpeg -y -f rawvideo -init_hw_device 'vulkan=vulkan:NVIDIA L40S,drm_modifiers=0' -pix_fmt yuv420p -s {source resolution} -r {source fps} -i {source.yuv} -filter_complex '[0]hwupload[input_in_gpu_memory]' -map '[input_in_gpu_memory]' -b:v {target bitrate}k -c:v lcevc_hevc_vulkan -fps_mode passthrough -base_encoder nvenc_hevc -eil_params "preset=4;gpu_device=NVIDIA;lcevc_tune=vq;tuning_info=ultra-high-quality;lcevc_preset=3;enable_badapt=0;enable_iadapt=0;m_hf_modulation=0.35;encoding_upsample=modifiedcubic;uni_directional_b=0;temporal_dynamic_refresh=0;rc_pcrf_base_reconfig_mode=1;rc_pcrf_sharpness_strength=0.65" output.ts
x265 medium	
LL	ffmpeg -y -f rawvideo -pix_fmt yuv420p -s {source resolution} -r {source fps} -i {source.yuv} -g -1 -b:v {target bitrate}k -c:v libx265 -tune zerolatency -preset fast -s {encoded resolution} -x265-params aq-mode=3:profile=main:scenecut=0:aq-strength=1:vbv-bufsize={4xtarget bitrate/fps}:vbv-maxrate={target bitrate}:rc-lookahead=0 output.ts
UHQ	ffmpeg -y -f rawvideo -pix_fmt yuv420p -s {source resolution} -r {source fps} -i {source.yuv} -b:v {target bitrate}k -c:v libx265 -vsync 0 -preset medium -s {encoded resolution} -x265-params vbv-bufsize={target bitrate} output.ts
LCEVC x264 Medium	
LL	ffmpeg -y -f rawvideo -pix_fmt yuv420p -s 3840x2160 -r {source fps} -i {source.yuv} -g -1 -b:v {target bitrate}k -c:v lcevc_hevc -base_encoder x265 -s 3840x2160 -eil_params "tune=zerolatency;preset=fast;aq-mode=3;bframes=0;profile=main;scenecut=0;lcevc_tune=low_latency;aq-strength=1;lcevc_preset=3;rc-lookahead=0;m_hf_modulation=0.35;encoding_upsample=modifiedcubic;rc_base_buffer_size={150%(4x target bitrate)};rc_pcrf_base_rc_mode=cbr" output.ts
UHQ	ffmpeg -y -f rawvideo -pix_fmt yuv420p -s 3840x2160 -r 60 -i input.yuv -b:v {target bitrate}k -c:v lcevc_hevc -vsync 0 -base_encoder x265 -s 3840x2160 -eil_params "preset=medium;lcevc_tune=vq;lcevc_preset=3;m_hf_modulation=0.35;encoding_upsample=modifiedcubic;rc_pcrf_base_reconfig_mode=0;rc_pcrf_sharpness_strength=0.15" output.ts

Command Lines for Performance Tests

The performance command lines used here are consistent with those from the visual quality tests. For CPU-based tests with x265/LCEVC x265, the default threading was applied, while for LCEVC x265, threading was explicitly set to `frame-threads=0` to ensure the default threading behavior.

Below are sample performance command lines for both x265 and LCEVC x265.

x265 Medium, 2160p	
LL	ffmpeg -readrate 0 -i {source.mp4} -filter_complex "[0]split=2[out0][out1]" -map "[out0]" -b:v {target bitrate}K -c:v libx265 -preset medium -profile main -rc-lookahead 0 -tune zerolatency -aq-strength 1 -aq-mode 3 -x265-params scenecut=0:vbv-bufsize=1333:vbv-maxrate={target bitrate} {output.ts} -vstats -vstats_file vstats.log -stats_period 0.5 -map "[out1]" -b:v {target bitrate}K -c:v libx265 -preset medium -profile main -rc-lookahead 0 -tune zerolatency -aq-strength 1 -aq-mode 3 -x265-params scenecut=0:vbv-bufsize=1333:vbv-maxrate={target bitrate} {output.ts} -vstats -vstats_file vstats.log -stats_period 0.5 -hide_banner -y -vsync 0
UHQ	ffmpeg -readrate 0 -i {source.mp4} -filter_complex "[0]split=2[out0][out1]" -map "[out0]" -b:v {target bitrate}K -c:v libx265 -preset medium -profile main -x265-params vbv-bufsize={target bitrate} {output.ts} -vstats -vstats_file vstats.log -stats_period 0.5 -map "[out1]" -b:v {target bitrate}K -c:v libx265 -preset medium -profile main -x265-params vbv-bufsize={target bitrate} {output.ts} -vstats -vstats_file vstats.log -stats_period 0.5 -hide_banner -y -vsync 0
LCEVC x264 Medium, 2160p	
LL	ffmpeg -readrate 0 -i {source.mp4} -filter_complex "[0]split=2[out0][out1]" -map "[out0]" -b:v {target bitrate}K -c:v lcevc_hevc -base_encoder x265 -g -1 -eil_params "tune=zerolatency;aq-mode=3;scenecut=0;lcevc_tune=low_latency;lcevc_preset=3;preset=medium;profile=main;aq-strength=1;rc-lookahead=0;bframes=0;m_hf_modulation=0.35;encoding_upsample=modifiedcubic;api_mode=synchronous;rc_base_buffer_size={150%(4x target bitrate)};frame-threads=0;rc_pcrf_base_rc_mode=cbr" {output.ts} -vstats -vstats_file vstats.log -stats_period 0.5 -map "[out1]" -b:v {target bitrate}K -c:v lcevc_hevc -base_encoder x265 -g -1 -eil_params "tune=zerolatency;aq-mode=3;scenecut=0;lcevc_tune=low_latency;lcevc_preset=3;preset=medium;profile=main;aq-strength=1;rc-lookahead=0;bframes=0;m_hf_modulation=0.35;encoding_upsample=modifiedcubic;api_mode=synchronous;rc_base_buffer_size={150%(4x target bitrate)};frame-threads=0;rc_pcrf_base_rc_mode=cbr" {output.ts} -vstats -vstats_file vstats.log -stats_period 0.5 -hide_banner -y -vsync 0
UHQ	ffmpeg -readrate 0 -i {source.mp4} -filter_complex "[0]split=2[out0][out1]" -map "[out0]" -b:v {target bitrate}K -c:v lcevc_hevc -base_encoder x265 -g -1 -eil_params "lcevc_encoder_type=cpu;lcevc_tune=vq;lcevc_preset=3;preset=medium;m_hf_modulation=0.35;encoding_upsample=modifiedcubic;api_mode=synchronous;frame-threads=0;rc_pcrf_base_reconfig_mode=0;rc_pcrf_sharpness_strength=0.15" {output.ts} -vstats -vstats_file vstats.log -stats_period 0.5 -map "[out1]" -b:v {target bitrate}K -c:v lcevc_hevc -base_encoder x265 -g -1 -eil_params "lcevc_encoder_type=cpu;lcevc_tune=vq;lcevc_preset=3;preset=medium;m_hf_modulation=0.35;encoding_upsample=modifiedcubic;api_mode=synchronous;frame-threads=0;rc_pcrf_base_reconfig_mode=0;rc_pcrf_sharpness_strength=0.15" {output.ts} -vstats -vstats_file vstats.log -stats_period 0.5 -hide_banner -y -vsync 0

How to access the libraries

The software required to replicate the experiments described in this document can be downloaded from V-Nova's download portal: download.v-nova.com. Registration onto the portal will give access to the generally available "LCEVC Trial SDK". These packages are ready-to-use with support for LCEVC-enhancement of x264 and x265. To access additional software as used in this blog, e.g. LCEVC GPU encoder for NVENC please contact V-Nova for further support via the Contact link on the Download Portal.