Work-let Name: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Worklet Details

1. Worklet ID:
   xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
2. College Name:
   xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

KPIs achieved till now

[xxxxxx]

Your paragraph text

Any Challenges/ Issues faced

[xxxxxx]

Next Steps

[xxxxxx]

Key Achievements/ Outcome till now

[xxxxxx]

Date: xxxxxxx

Selective encryption for H.264/AVC Video streams

WORKLET ID: 24VI59IITJ

Literature Review

Prediction:

Video frames are divided into 16x16 pixel macroblocks, and predictions are made using inter-prediction (leveraging previous frames) or intra-prediction (using data within the same frame).

Transform:

Prediction residuals are transformed from spatial to frequency domain using a simplified Discrete Cosine Transform (DCT). The data is then quantized to discard less significant details, trading off between file size and quality.

Encoding:

The quantized data, along with prediction and sequence metadata, is organized into a compressed bitstream for efficient storage and transmission. This includes motion vectors, quantized coefficients, and structure details.

Decompression:

The compressed file is decompressed by reversing the process: decoding the bitstream, applying inverse transformation and quantization, and reconstructing frames using prediction data and residuals.

Challenges

Cost and Licensing:

Elecard StreamEye is a paid tool, many specific analysis tools can not be accessed without purchasing a License

Visualizing Block Movement:

We are unable to see the details of each block and how they are moving frame by frame in a textual representation.

Codec Compatibility:

The project involves using H.264 video, but the encryption scheme needs to be codec-agnostic to potentially work with other video codecs (e.g., HEVC, VP9)

Maintaining Video Quality:

Selective encryption might cause degradation in video quality if not done carefully. Encrypting parts of the video stream, such as specific frames or regions, could introduce artifacts or distortion, especially if the encryption interferes with the decoding process.

Next Steps

Selecting Encryption Scheme:

Choose efficient algorithms (e.g., AES-CTR) and target specific H.264 components (I-frames, motion vectors) to balance security and performance.

Visualizing the Stream:

Compare encrypted and original frames to identify scrambled regions and analyze encrypted NAL units via tools like FFmpeg.

Codec-Agnostic Approach:

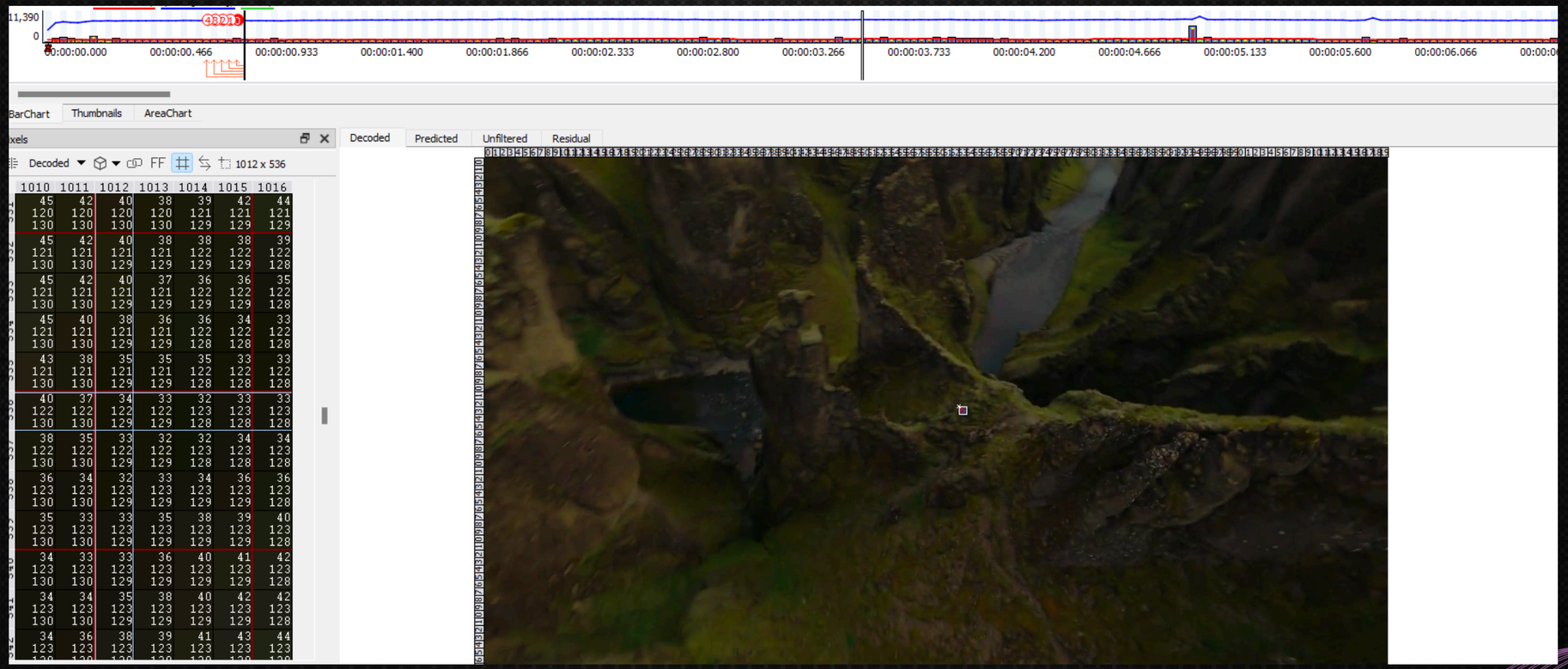
Design an abstraction layer to encrypt common structures (e.g., NAL units) across different codecs (H.264, HEVC, VP9).

Key Factors in Video Compression:

Key issues include maintaining compression efficiency, low-latency processing, hardware compatibility, and robust key management.

Outcomes Till Now

Visualisation on Elecard StreamEye



1

The software allows frame-level inspection with pixel-level granularity, displaying exact pixel values in a grid format. It supports viewing Decoded, Predicted, Unfiltered, and Residual data, which is crucial for understanding how video frames are processed and reconstructed

2

Elecard StreamEye provides comprehensive bitstream analysis, as shown by the graph at the top displaying bit allocation, quantization levels, and metrics over time

stream type	AVC/H.264
profile	High
level	5.0
chroma format	4:2:0
bitdepth	8
resolution	1920 x 1080
frame rate	30.00
coding mode	CABAC
interlace	no
declared bitrate	Undefined
duration	00:01:07:166
mux duration	00:01:07:166
epsnr	44.47
frames	2 015
I	12 (0.59%)
P	1 857 (92.15%)
B	146 (7.24%)
size (byte) / encode ratio (a...	22 810 / 136
I	127 215 / 24
P	23 207 / 134
B	9 171 / 339
bit allocation avg	5 474 417
max	8 320 840 [265]
min	3 239 784 [1 082]
instant bitrate	
max	8 320 840
min	9 240
qp	21.18
max	27
min	12
colour description	
colour primaries	ITU-R BT.709-5
transfer characteristics	ITU-R BT.709-5
matrix coeffs	ITU-R BT.709-5
distribution (avg bits per frame)	
total_size	174 511 (100.00%)
mb_skip	2 630 (1.51%)
mb_field_decoding_flag	0 (0.00%)
mb_type	8 426 (4.83%)
sub_mb_type	69 (0.04%)
prediction	
pred_mode_flag	2 945 (1.69%)
pred_mode	7 709 (4.42%)
inter_prediction	52 893 (30.31%)
transform	
coded_block_pattern	15 434 (8.84%)
transform_size_8x8_flag	219 (0.13%)
mb_qp_delta	9 589 (5.49%)
transform	74 598 (42.75%)

Visualisation on Elecard StreamEye

1

The software allows frame-level inspection with pixel-level granularity, displaying exact pixel values in a grid format. It supports viewing Decoded, Predicted, Unfiltered, and Residual data, which is crucial for understanding how video frames are processed and reconstructed

2

Elecard StreamEye provides comprehensive bitstream analysis, as shown by the graph at the top displaying bit allocation, quantization levels, and metrics over time

FFMPEG

Comprehensive Video Handling:

FFmpeg supports nearly every video codec (including H.264, HEVC, VP9) and container format, making it ideal for managing a variety of video types in a single pipeline.

Flexible Encoding and Decoding:

FFmpeg can be used to decode videos to individual frames or packets, modify them (e.g., applying encryption), and then re-encode the processed video, allowing for seamless playback after encryption.

Cross-Platform Support:

FFmpeg works on various operating systems, including Linux, Windows, and macOS, making it an accessible solution for development and deployment across different environments.

Open Source and Highly Customizable:

Being open-source, FFmpeg is free to use, modify, and integrate into your project. This allows for customization and flexibility in implementing specific features or optimizations tailored to selective encryption.

FFProbe

1 Extract Video Metadata:

FFprobe can be used to extract detailed video information such as codec type, frame rate, resolution, bitstream structure, keyframe positions, and GOP (Group of Pictures) structure.

2 Bitrate and Quality Analysis:

FFprobe helps in measuring the bitrate and video quality, which is crucial for assessing how much encryption impacts video performance and quality.

3 Identify Keyframes and Frames for Encryption:

FFprobe can be used to identify the keyframes (I-frames) and the type of frames (P-frames, B-frames), which can guide decisions on which frames or portions of the video stream to selectively encrypt.

FFProbe

Motion Vectors

```
frame= 3122 fps=286 q=-1.0 Lsize=     802KiB time=00:00:52.05 bitrate= 126.2kbits/s speed=4.78x
[libx264 @ 000001ee5ee900c0] frame I:20    Avg QP:15.05  size:  1444
[libx264 @ 000001ee5ee900c0] frame P:1066  Avg QP:19.87  size:   565
[libx264 @ 000001ee5ee900c0] frame B:2036  Avg QP:19.81  size:    94
[libx264 @ 000001ee5ee900c0] consecutive B-frames:  8.0% 12.9%  6.8% 72.3%
[libx264 @ 000001ee5ee900c0] mb I  I16..4:  1.5% 94.0%  4.4%
[libx264 @ 000001ee5ee900c0] mb P  I16..4:  0.4%  2.0%  0.5%  P16..4:  3.9%  2.4%  1.2%  0.0%  0.0%    skip:89.7%
[libx264 @ 000001ee5ee900c0] mb B  I16..4:  0.1%  0.1%  0.0%  B16..8:  3.3%  0.5%  0.1%  direct: 0.1%  skip:95.8%  L0:45
.3% L1:45.4% BI: 9.3%
[libx264 @ 000001ee5ee900c0] 8x8 transform intra:77.6% inter:68.8%
[libx264 @ 000001ee5ee900c0] coded y,u,v intra: 16.4% 7.7% 8.6% inter: 1.1% 0.3% 0.3%
[libx264 @ 000001ee5ee900c0] i16 v,h,dc,p: 55% 29% 10%  5%
[libx264 @ 000001ee5ee900c0] i8 v,h,dc,ddl,ddr,vr,hd,vl,hu: 24% 12% 59%  1%  1%  1%  1%  1%  1%
[libx264 @ 000001ee5ee900c0] i4 v,h,dc,ddl,ddr,vr,hd,vl,hu: 42% 19% 16%  3%  4%  5%  4%  4%  4%
[libx264 @ 000001ee5ee900c0] Weighted P-Frames: Y:31.4% UV:14.2%
[libx264 @ 000001ee5ee900c0] ref P L0: 69.3% 15.2% 11.1%  3.9%  0.5%
[libx264 @ 000001ee5ee900c0] ref B L0: 90.4%  8.0%  1.6%
[libx264 @ 000001ee5ee900c0] ref B L1: 98.6%  1.4%
[libx264 @ 000001ee5ee900c0] kb/s:126.13
```

Video Metadata