Lab Report: Text, Audio, and Image Data Manipulation

107474-Joseane Pereira
109050-Gabriel Costa
108538-Francisco Gonçalves
Universidade de Aveiro, DETI

December 28, 2024

Contents

1 Introduction

This project implements a video codec system using both intra-frame and inter-frame compression techniques. The implementation focuses on efficient compression while maintaining video quality through predictive coding, motion estimation, and Golomb encoding.

2 System Architecture

2.1 Core Components

The system consists of four main components:

- **BitStream**: Handles bit-level I/O operations for binary file manipulation
- **Golomb Codec**: Implements Golomb-Rice coding for entropy encoding
- **Image Codec**: Manages image compression using predictive coding
- **Video Codecs**: Implements both intra-frame and inter-frame compression

2.2 Implementation Details

2.2.1 BitStream Class

Provides low-level bit manipulation:

- Bit-level read/write operations
- Buffer management for efficient I/O
- Support for variable-length integer encoding

2.2.2 Golomb Encoding

Implements efficient entropy coding:

- Parameter 'm' optimization for data characteristics
- Support for both signed and unsigned integers
- Zigzag encoding for efficient signed number representation

2.2.3 Image Codec

Features predictive coding techniques:

- A-type predictor for spatial redundancy reduction
- Multi-channel support for color images
- Residual calculation and reconstruction

3 Video Compression Techniques

3.1 Intra-Frame Coding

Implements frame-independent compression:

- Channel separation for RGB frames
- Predictive coding using spatial correlations
- Single-file storage optimization for all frames
- Metadata management for frame properties

3.2 Inter-Frame Coding

Utilizes temporal redundancy:

- Motion estimation using block matching
- Configurable block size and search range
- I-frame and P-frame management
- Motion vector encoding and residual compression

4 Performance Analysis

4.1 Compression Efficiency

Method	Original Size	Compressed Size	Ratio	PSNR
Intra-Frame	X MB	Y MB	Z:1	W dB
Inter-Frame	X MB	Y MB	Z:1	W dB

Table 1: Compression Performance Comparison

4.2 Processing Time

- **Encoding Time**: Analysis of encoding speed per frame
- **Decoding Time**: Performance metrics for video playback
- **Motion Estimation**: Impact of block size and search range

4.3 Quality Assessment

Evaluation metrics include:

- PSNR (Peak Signal-to-Noise Ratio)
- MSE (Mean Squared Error)
- Visual quality comparison

5 Technical Innovations

5.1 Storage Optimization

- Single-file approach for all frame data
- Efficient metadata management
- Optimized binary format for frame storage

5.2 Motion Estimation

- Block-based search algorithm
- Adaptive motion vector encoding
- Efficient residual calculation

6 Future Improvements

Potential enhancements include:

- Advanced prediction modes
- Parallel processing support
- Adaptive Golomb parameter selection
- B-frame implementation
- Rate control mechanisms

7 Conclusion

The implemented video codec system demonstrates effective compression through:

- Efficient entropy coding using Golomb encoding
- Effective motion estimation and compensation
- Optimized storage mechanisms
- Balance between compression ratio and quality

A Implementation Details

Key implementation highlights and code snippets:

A.1 Golomb Encoding Example

```
void encode(int value) {
    if (mode == 0) {
        bs.writeBit(value < 0);
        value = abs(value);
    } else {
        value = zigzagEncode(value);
    }
    // ... encoding implementation
}
```

A.2 Motion Estimation Example

```
Mat calculateResidual(const Mat &current,
                      const Mat &reference,
                      vector<Point2i> &motionVectors) {
    // ... motion estimation implementation
}
```