Motion Estimation & Compensation

313553014 Yi-Cheng Liao

1 Introduction

In this assignment, we focus on **motion estimation (ME)** and **motion compensation (MC)**, which are the fundamental techniques used in modern video compression standards such as MPEG and H.264. Motion estimation attempts to find the movement of image blocks between two consecutive frames, while motion compensation reconstructs the current frame based on the estimated motion vectors. The objective is to exploit temporal redundancy between frames to reduce the overall data required for video transmission.

1.1 Full Search

The **Full Search Block Matching Algorithm (FS-BMA)** exhaustively searches every possible displacement within a predefined search range $\pm R$ for each block (in this assignment, 8×8). For each candidate, the cost function—typically the Sum of Absolute Differences (SAD)—is computed as

$$\mathrm{SAD}(dx, dy) = \sum_{i,j} |B_t(i, j) - B_{t-1}(i + dy, j + dx)|.$$

The motion vector (dx, dy) with the lowest SAD is chosen as the best match. Although full search guarantees the optimal match, it is computationally expensive since the complexity grows with $(2R + 1)^2$.

1.2 Three-step Search

The **Three-Step Search (TSS)** algorithm aims to reduce computation while maintaining good accuracy. It starts from a large step size (usually $R/2$) and evaluates nine candidate positions (the center and its eight neighbors). The position with the smallest SAD becomes the new center, and the step size is halved in each iteration. After three iterations, the final motion vector is obtained. This hierarchical search greatly reduces runtime, making it suitable for real-time applications.

1.3 Motion Compensation

After motion estimation, motion compensation reconstructs the current frame by shifting the reference blocks according to the estimated motion vectors. The residual image is calculated as

$$\mathrm{Residual}(x, y) = I_{\mathrm{current}}(x, y) - I_{\mathrm{reconstructed}}(x, y),$$

which represents the difference that cannot be explained by motion alone (e.g., new objects or illumination changes). In this assignment, both the reconstructed frame and residual image are saved for analysis, and **PSNR** (Peak Signal-to-Noise Ratio) is used to evaluate reconstruction quality.

2 Experimental Results and Discussion

In this section, we compare the performance of the **Full Search (FS)** and **Three-Step Search (TSS)** algorithms under different search ranges (± 8, ± 16, and ± 32). The performance is evaluated in terms of **Peak Signal-to-Noise Ratio (PSNR)** and **Runtime**. All experiments use 8×8 block size and integer-pixel accuracy. The reference frame is `one_gray.png`, and the target frame is `two_gray.png`.

2.1 Quantitative Results

Table 1 summarizes the quantitative results. As the search range increases, the PSNR of Full Search improves significantly, but the runtime also grows rapidly due to its quadratic computational complexity. In contrast, TSS achieves much faster processing with only a small number of search evaluations, but its PSNR remains low because the search space is coarsely sampled.

Table 1: Comparison of PSNR and Runtime between Full Search and Three-Step Search

Search Range (R)	**Full Search**		**Three-Step Search**	
	PSNR (dB)	Runtime (s)	PSNR (dB)	Runtime (s)
± 8	22.950	4.361	16.667	0.512
± 16	25.508	14.271	16.913	0.546
± 32	29.103	54.165	16.678	0.619

2.2 Qualitative Observations

Figure 1 and Figure 2 show the reconstructed frames and residuals for both methods. For Full Search, increasing the search range from ± 8 to ± 32 results in visually smoother reconstructed frames and weaker residual energy, indicating more accurate motion estimation. In contrast, the TSS-reconstructed images exhibit more block artifacts and larger residuals, especially around moving edges such as the car boundaries, due to its coarse hierarchical search.

Figure 1: Reconstructed frames using Full Search with $R = \{8, 16, 32\}$.

Figure 2: Reconstructed frames using Three-Step Search with $R = \{8, 16, 32\}$.

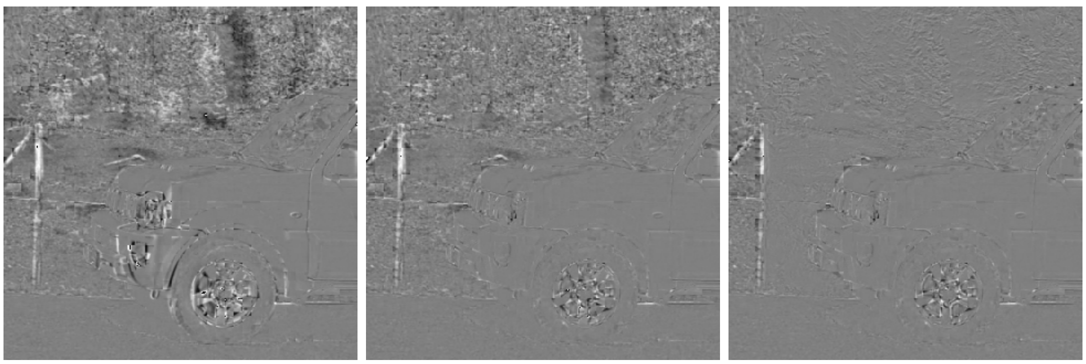

Figure 3: Residual images of Full Search. Higher search range yields smaller residuals.

Figure 4: Residual images of TSS. Notice larger differences and stronger edge artifacts.

2.3 Discussion

The results demonstrate the trade-off between accuracy and computational cost:

- **Full Search** achieves the highest PSNR (up to 29.1 dB) but requires over 50 seconds when $R = \pm 32$.
- **TSS** is approximately **8–10× faster**, with runtime below 1 second even for $R = \pm 32$, but the PSNR remains around 16–17 dB.
- The residual images confirm that Full Search can better predict motion in highly dynamic regions, while TSS tends to produce blocky residuals due to limited search granularity.

Overall, Full Search serves as a ground truth baseline for motion estimation accuracy, whereas Three-Step Search represents a practical balance between speed and acceptable visual quality, commonly used in real-time video coding systems.