Motion, Flow of Optics, and Motion Vector

1. Define motion estimation in computer vision and discuss its importance in various applications.

Ans: Motion estimation is a process in computer vision that involves determining the movement of objects or pixels between consecutive frames in a video sequence. It quantifies how much and in what direction objects have moved, typically represented by motion vectors.

Importance in Applications:

Video Compression (e.g., MPEG, H.264): Motion estimation reduces redundancy by encoding only the changes (motion vectors) between frames, significantly lowering file sizes.

Object Tracking: Used in surveillance, autonomous vehicles, and robotics to follow objects across frames.

Action Recognition: Helps in analyzing human activities or gestures for applications like sports analytics or security.

Video Stabilization: Corrects shaky footage by estimating and compensating for unwanted camera motion.

Augmented Reality (AR): Aligns virtual objects with real-world motion for seamless integration.

2. Discuss the challenges faced in motion estimation, particularly in the presence of occlusions and complex scene dynamics. Propose potential solutions to address these challenges.

Ans: Challenges:

Occlusions: When objects are hidden or revealed (e.g., a person walking behind a tree), motion estimation fails for the obscured regions.

Complex Dynamics: Rapid motion, non-rigid deformations (e.g., cloth fluttering), or lighting changes complicate motion tracking.

Textureless Regions: Lack of features (e.g., blank walls) makes it hard to estimate motion.

Computational Cost: Dense motion estimation is resource-intensive.

Potential Solutions:

Occlusion Handling: Use bidirectional motion estimation (forward/backward tracking) or mask occluded areas explicitly.

Hierarchical Methods: Coarse-to-fine approaches (e.g., pyramidal optical flow) handle large motions.

Feature-Based Tracking: SIFT or ORB can robustly track keypoints in dynamic scenes.

Deep Learning: CNNs or transformers (e.g., FlowNet, RAFT) learn to predict motion from data, improving robustness.

3. Explain the concept of optical flow and its role in motion estimation. Discuss common optical flow algorithms and their applications.

Ans: Optical Flow is the pattern of apparent motion of objects between frames caused by relative movement between the observer (camera) and the scene. It outputs a vector field where each vector represents the displacement of a pixel.

Role in Motion Estimation:

Provides dense motion information (per-pixel vectors) for tasks like segmentation or 3D reconstruction.

Acts as input for higher-level processes (e.g., tracking, SLAM).

Common Algorithms:

Lucas-Kanade (Sparse): Tracks feature points assuming small, constant motion in local neighborhoods. Used in robotics and AR.

Horn-Schunck (Dense): Global smoothness constraint; suitable for slow, coherent motion (e.g., fluid dynamics).

Farnebäck (Dense): Polynomial expansion for robust flow; used in video stabilization.

Deep Learning (RAFT): State-of-the-art accuracy for complex scenes; applied in autonomous driving.

4. Define optical flow and explain its significance in computer vision applications.

Ans: Definition: Optical flow is the distribution of apparent velocities of brightness patterns in an image, representing pixel-wise motion between consecutive frames.

Significance:

Motion Analysis: Enables understanding of object/camera movement.

Real-Time Tracking: Critical for drones or self-driving cars to navigate dynamically.

Human-Computer Interaction: Powers gesture recognition in VR/AR.

Video Enhancement: Used in frame interpolation (e.g., slow-motion generation).

5. Describe the concept of motion vectors in video compression and discuss their role in reducing redundancy.

Ans: Motion Vectors are 2D vectors that indicate the displacement of a block of pixels between frames in compressed video (e.g., MPEG).

Role in Redundancy Reduction:

Inter-Frame Prediction: Instead of storing all pixels, codecs (e.g., H.265) predict frames using motion vectors and residual errors.

Block-Based Compression: Macroblocks in "P-frames" reference past/future frames via vectors, avoiding repeated data.

Efficiency: Motion vectors are compact (few bits), drastically reducing bitrate compared to raw frame storage.

Example: In a video with a moving ball, only the ball’s motion vector and small residual updates are encoded, not the entire frame