

Interframe-Codierung

Englisch: interframe coding

VFX

Kompressionsverfahren, das Unterschiede zwischen aufeinanderfolgenden Frames speichert statt ganzer Bilder — spart Speicher und Bandbreite massiv. H.264, H.265 arbeiten damit; unkomprimierte Proxies brauchen es nicht.

Wer mit Videomaterial arbeitet, stößt unweigerlich auf das Konzept der Interframe-Codierung — und zwar dort, wo es weh tut: bei der Datenmenge. Statt jeden Frame als vollständiges Bild zu speichern, merkt sich der Codec nur die Unterschiede zwischen aufeinanderfolgenden Frames auf. Das klingt theoretisch, ist aber praktisch das Einzige, das modernen Workflows Sinn macht. Ein 10-Sekunden-Shot in 4K — unkomprimiert sind das 80 GB pro Sekunde. Mit Interframe-Codierung? Plötzlich passen es auf eine SSD. Die Mechanik funktioniert so: Der Codec identifiziert einen Keyframe — ein vollständiges Bild — und speichert diesen. Die nächsten Frames danach werden als Differenzen kodiert: nur die Pixel-Veränderungen. In einer statischen Szene mit bewegtem Talent braucht man also nicht die ganze Hintergrund-Information jedes Mal neu zu speichern. H.264 und H.265 (HEVC) arbeiten nach diesem Prinzip — deshalb sind Dateien im ProRes oder DNxHD Material so viel kleiner als unkomprimiertes Zeug. Der Catch: Je volatiler der Motion, je mehr Schnitte oder Szenenwechsel, desto häufiger muss ein neuer Keyframe gesetzt werden, und die Ersparnis schrumpft. Am Set oder in der Post macht das einen Unterschied, den man nicht ignorieren kann. Bei der Offline-Editierung arbeitet man bewusst mit verdichteter Interframe-Kompression — ProxyGen mit H.264 spart Speicher und macht den Schnitt flüssig. Für Color Grading oder VFX-intensive Shots schaltet man dann auf Intraframe-Codierung oder gar unkomprimiertes Material um — hier ist Latenzbandbreite und Frame-Accuracy wichtiger als Speicherökonomie. Ein Motion-Graphics-Shot mit schnellen Übergängen? Interframe-Codec kann problematisch sein, weil Keyframe-Abstände Artefakte erzeugen. Hier lieber unten im Schnitt auf I-Frame-only gehen oder ProRes 422 HQ nutzen. Das Tückische: Wenn man später mit verdichteter Interframe-Kompression schneidet, reißen die Differenzdaten an den Schnittpunkten ab. Deshalb exportiert man zum Finalcut in höherer Qualität. Bei Live-Streaming oder Real-Time-Transmission ist Interframe-Codierung unverzichtbar — die Kompression macht's möglich. Für Archivierung langfristig: eher nicht, zu anfällig für Dekodierungs-Fehler über Jahrzehnte.