

Bildunterschriften zu Skalierbarer Videokodierung

- **Folie 2:** Ein skalierbarer Datenstrom kann – je nach Leistungsfähigkeit des Decoders – in unterschiedlichen zeitlichen und örtlichen Auflösungen sowie unterschiedlichen Qualitätsstufen abgespielt werden. Vor der Dekodierung wird dabei eine Teilmenge der vom Encoder erzeugten Daten extrahiert.
- **Folie 3:** Örtliche Skalierbarkeit beschreibt verschiedene Bildauflösungen, zeitliche Skalierbarkeit verschiedene Bildwiederholraten und qualitative Skalierbarkeit verschiedene Qualitätsstufen.
- **Folie 4:** Skalierbare Datenströme (rot und dunkelblau) können in breitbandigen Netzwerken vollständig übertragen werden und durch Adapter vor der Einspeisung in schmalbandigere Netzwerke, z.B. Mobilfunknetze, in ihrer Bandbreite reduziert werden. Je nach Komplexität des Adapters ist eine Datenratenreduktion pro Empfänger möglich, sodass jeder Empfänger einen von der Bandbreite her individuell zugeschnittenen Datenstrom (hellblaue Linien im rechten Bildteil; die Liniendicke gibt die Bandbreite an) erhält.
- **Folie 5:** Die Pixel eines Originalbildes (grau) sollen nach der Herunterskalierung weniger dicht platziert sein. Dadurch beinhaltet ein Pixel des herunterskalierten Bildes (weiß) mehrere Pixel des Originalbildes – das Einzugsgebiet jedes Pixels mit Fläche $4a^2$ ist mit einer dicken schwarzen Linie markiert. Das linke Bild zeigt einen ganzzahligen Skalierungsfaktor in beiden Richtungen (die Position der weißen Pixel relativ zu den grauen ist konstant), das rechte Bild einen nicht ganzzahligen (die Position der weißen Pixel relativ zu den grauen ändert sich mit der Pixelposition im Bild).
- **Folie 7:** Die bilineare Interpolation des Pixels $P = \begin{pmatrix} x \\ y \end{pmatrix}$ (grün) aus den Original(-nachbar-)pixeln $Q_{ij} = \begin{pmatrix} x_i \\ y_j \end{pmatrix}$, $i, j \in \{1, 2\}$ (rot) erfolgt durch (links) zweimalige lineare Interpolation in x-Richtung (blau, $R_j = \begin{pmatrix} x \\ y_j \end{pmatrix}$, $j \in \{1, 2\}$ aus $\overline{Q_{1j}Q_{2j}}$) und anschließende lineare Interpolation in y-Richtung ($P = \begin{pmatrix} x \\ y \end{pmatrix}$ aus $\overline{R_1R_2}$). Alternativ (rechts) kann das zu interpolierende Pixel (schwarz) aus der gewichteten Summe der zwischen dem Pixel und dem jeweils gegenüberliegenden Nachbarpixel eingeschlossenen Flächen berechnet werden.
- **Folie 8:** Je nach Pixelabstand (x-Achsen-Wert) vom zu interpolierenden Pixel werden die Original(-nachbar-)pixel mit dem Funktionswert der roten Kurve an dieser Stelle gewichtet. Verschiedene Werte von b (links

$b = 2$, rechts $b = 3$) führen zu unterschiedlichen Gewichtungen sowie einer veränderten Anzahl berücksichtigter Nachbarnpixel (vier bei Lanczos-2, sechs bei Lanczos-3).

- **Folie 11:** Bei SVC erfolgt die Kodierung der niedrigsten örtlichen Auflösung (auf Basis eines verkleinerten Originalbildes) regulär nach H.264 (gelbe Schattierung und rote Elemente). In den höheren örtlichen Auflösungen (hellblaue Schattierung) wird das kodierte Bild des jeweils darunter liegenden Layers heraufskaliert und als Prädiktionsbasis verwendet (grün). Pro örtlichem Layer wird die Differenz zwischen der Prädiktion und dem (eventuell heraufskalierten) Originalbild kodiert. Innerhalb jedes Layers wird qualitative Skalierbarkeit über progressive Kodierung (dunkelblau) realisiert. Die kodierten Daten aller Layer werden abschließend gemuxt und bilden den skalierbaren Bitstrom.
- **Folie 12:** Zeitliche Skalierbarkeit wird über eine hierarchische Prädiktionsstruktur realisiert. Die roten I- und P-Frames bilden den untersten Layer T_0 , die dazwischen liegenden blauen Frames über bidirektionale Prädiktion den nächsthöheren T_1 . Weitere zeitliche Layer $T_i, i \in \mathbb{N} \setminus \{0\}$ können analog durch das Einfügen weiterer B-Frames zwischen den Frames des darunter liegenden Layers T_{i-1} erzeugt werden, wodurch die Framerate verdoppelt wird.
- **Folie 14:** Bei der Inter-Layer-Intraprediktion (links) werden Blöcke des darunter liegenden Layers heraufskaliert und als Prädiktionsbasis verwendet. Bei der Inter-Layer-Modi- und Bewegungsprädiktion (Mitte) werden die Blockgrößen, Bewegungsvektoren und andere Daten heraufskaliert und als Prädiktionsbasis verwendet. Bei der Inter-Layer-Differenz-Prädiktion werden Motion-Compensation-Daten (Differenzen) heraufskaliert und als Prädiktionsbasis verwendet. Inter-Layer-Prädiktion in die Gegenrichtung, d.h. von höheren zu niedrigeren Layern, ist nicht erlaubt.
- **Folie 15:** Qualitative Skalierbarkeit erlaubt das zeitlich variable Verwerfen einzelner Teile der Verbesserungslayer (rot), um auf schwankende Bandbreiten reagieren zu können. Der Basislayer (blau) muss dabei immer erhalten bleiben, da ohne ihn keine Wiedergabe möglich ist.
- **Folie 16:** Örtliche Skalierbarkeit lässt sich mit zeitlicher kombinieren, wenn die Prädiktionsstruktur (strichlierte Linien) des Basislayers (unten) der des Verbesserungslayers (oben) entspricht und der Verbesserungslayer für die höchste zeitliche Auflösung (hellgrau) nicht vom Basislayer prädiziert.
- **Folie 17:** Grobe qualitative Skalierbarkeit lässt sich als Spezialfall von örtlicher Skalierbarkeit modellieren, bei dem die Auflösung dieselbe bleibt. Im Beispiel wird der mittlere Layer mit Faktor 1 heraufskaliert, d.h. unverändert übernommen, und progressiv verbessert, wodurch der obere Verbesserungslayer entsteht. Zeitliche Skalierbarkeit lässt sich mit örtlicher

wie bereits gezeigt kombinieren, wodurch insgesamt zeitliche, örtliche und qualitative Skalierbarkeit kombiniert werden können.

- **Folie 18:** Im Vergleich zu Simulcast (schwarz) ist SVC (hellrot) deutlich effizienter. Zwar kommt seine Performance nicht ganz an jene Konfiguration heran, bei der ausschließlich der höchste Layer regulär nach H.264 kodiert wird (hellgrau), doch ist der Overhead relativ gering.
- **Folie 19:** Der Overhead von SVC im Vergleich zur regulären Kodierung des höchsten Layers nach H.264 (schwarz) beträgt ca. 10% (rote bzw. graue Kurve). Dieser Overhead ist praktisch unabhängig von der Auflösung (CIF vs. 4CIF) und erreicht in praktischen Szenarien nie den Overhead von Simulcast (blau).