

Facharbeit Informatik

Vergleich zwischen den verlustbehafteten
Videokomprimierungsalgorithmen
AVC, HEVC, VP9 und AV1 –
Kann AV1 zum neuen Standard werden?

Gymnasium Pelizaeus Paderborn
Facharbeit im Thema: Videokomprimierungsalgorithmen

Verfasser: Langrock, Lukas
Kurs: IF-LK-1
Fachlehrerin: Frau Löseke
Schuljahr: 2020/2021
Bearbeitungszeit: 6 Wochen
Abgabetermin: 12. April 2021

Inhaltsverzeichnis

| | |
|---|----|
| Einleitung..... | 1 |
| Hardwarebeschleunigung..... | 2 |
| Core Coding Tools..... | 2 |
| Partitions..... | 3 |
| Intra Prediction (Spatial Redundancy)..... | 4 |
| Inter Prediction (Temporal Redundancy)..... | 5 |
| Encoding Stages..... | 6 |
| Transformation & Quantization..... | 7 |
| Entropy Coding..... | 7 |
| In-loop Filtering..... | 8 |
| Film Grain Synthesis..... | 8 |
| Vergleich zwischen den Videocodern..... | 9 |
| Effizienz..... | 9 |
| Geschwindigkeit..... | 9 |
| Lizenzen & Patente..... | 10 |
| Adoption & Verfügbarkeit..... | 10 |
| Qualitätsvergleich..... | 11 |
| BD-Rate..... | 11 |
| Ergebnisse..... | 11 |
| Zusammenfassung..... | 12 |
| Quellenverzeichnis..... | 13 |

Einleitung

In dieser Facharbeit werde ich die Videokomprimierungsalgorithmen AVC (Advanced Video Codec), HEVC (High Efficiency Video Codec) und VP9, welche derzeit die am weit verbreitetsten und effizientesten sind, mit AV1 (AOMedia Video 1) vergleichen.

Videokomprimierungstechnologien sind ein wesentlicher Baustein der nicht nur im Internet, sondern auch im Fernsehen oder in physischen Medien essentiell ist. Ein konventionelles 60 Sekunden langes 1080p RGB-Video mit 8bit (pro Kanal) und 25 Bildern pro Sekunde würde unkomprimiert $\sim 9,33 \text{ GB}^{[1]}$ an Speicher benötigen und, über das Internet gesendet, eine Geschwindigkeit von mindestens $\sim 1,3 \text{ Gigabit/s}$ ($\sim 1300 \text{ Megabit/s}$) voraussetzen. In diesem Speicherformat ist es nach heutigem Standard nahezu unmöglich ein Video in Echtzeit über das Internet zu transferieren, da der weltweit durchschnittliche Internet-Breitband-Anschluss eine Download-Geschwindigkeit von $\sim 97 \text{ Megabit/s}^{[2]}$ besitzt.

Moderne Videokomprimierungstechnologien, welche beispielsweise bei der Videostreaming-Plattform YouTube Anwendung finden, erlauben es das gleiche Video bei einer subjektiv leicht reduzierten Videoqualität auf $\sim 60 \text{ MB}^{[3]}$ zu reduzieren und erfordern nur lediglich $\sim 8 \text{ Megabit/s}$ an Internet Bandbreite. Dies liegt deutlich unter dem $\sim 97 \text{ Megabit/s}^{[2]}$ Durchschnitt und kann daher selbst von sehr schwachen DSL-Anschlüssen, die weit unter diesem Durchschnitt liegen, aufgebracht werden können, um eine Echtzeit Videoübertragung zu ermöglichen.

Durch neuere Codecs, wie AV1, kann es nun ermöglicht werden bei gleicher Bandbreite eine höhere Qualität oder Auflösung zu übertragen oder die gleiche Qualität und Auflösung bei einer kleineren Datenrate zu ermöglichen. Dadurch wäre es möglich ein HD Video in Echtzeit bei noch schlechteren Internetbedingungen zu liefern, bei denen der Nutzer zuvor auf eine schlechtere Auflösung oder Qualität umschalten musste.

AV1 steht in dieser Facharbeit in einem besonderen Fokus, da er nicht nur der neuste und effizienteste Videocodec ist, sondern auch jedem frei zur Verfügung steht. Er ist das Produkt der Alliance for Open Media, einem Zusammenschluss vieler führender Technologiekonzerne um offene und lizenzfreie Lösungen für die nächste Generation der Medienverteilung zu entwickeln. Zu AOMedia gehören unter anderem Google, Netflix, Nvidia, Intel, AMD, Facebook, Cisco, Amazon, VideoLAN, Broadcom und Samsung, sowie viele mehr.^[4]

Aspekte eines Videocodecs

Der zentrale Fokus eines Videocodecs liegt darin ein Video beliebiger Art in möglichst guter Qualität bei einer hohen Effizienz und Geschwindigkeit in eine kleinere Datei zu codieren, welche auf möglichst vielen Geräten abspielbar ist.

Hardwarebeschleunigung

Dazu wird nun entweder reine Software genutzt oder Software mit Hardwarebeschleunigung, welche durch einen ASIC (Application-specific integrated circuit), der speziell entwickelt dafür wurde ein Video zu decodieren oder encodieren, beschleunigt wird. Hardware Beschleunigung durch einen ASIC hat den Vorteil, dass er im Gegensatz zur CPU, auf der die reine Verarbeitung durch Software stattfindet, deutlich schneller und effizienter in der Aufgabe ist, für die er entwickelt wurde.

Dies liegt daran, dass Software die Anweisungen erst in Maschinsprache umwandeln muss und auf Hardware, optimiert für Berechnungen aller Art, ausgeführt wird. Ein ASIC dagegen wird optimiert für einen einzelnen Fall und muss lediglich eine Art bzw. eine vorgegebene Folge von Berechnungen durchführen und kann daher für hohe Geschwindigkeiten dieser einzelnen Berechnungen optimiert werden.

Dazu hat Hardware Beschleunigung den Vorteil, dass ein System mit der Fähigkeit ein Video mittels Hardware zu decodieren deutlich weniger Energie benötigt als, wenn es Software Decodierung nutzen würde. Besonders für mobile Geräte wie Smartphones oder Laptops ist diese Fähigkeit von großem Wert. Ein Videokomprimierungsalgorithmus, welcher nur schwer in Hardware eingearbeitet werden kann, ist wesentlich unwahrscheinlicher von vielen mobilen Geräten genutzt zu werden und wird daher auch weniger Adoption von Diensten und Firmen finden.

Alle der in dieser Facharbeit zum Vergleich stehenden Videocodecs wurden daher mit einem Fokus auf Hardware Implementationen entwickelt.

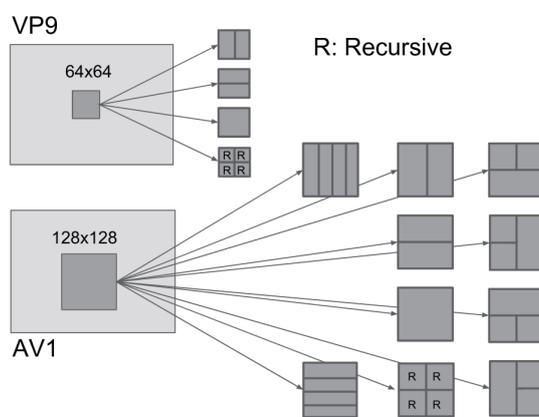
Core Coding Tools

Um nun ein Video zu codieren werden komplexe mathematische Funktionen genutzt um zum einen ein die einzelnen Bilder (Frames) eines Videos so effizient wie möglich zu speichern (Intracoding) und auch die Unterschiede zwischen den nächsten und der vorherigen Frames effizient zu sichern (Intercoding).^[C8] Dabei werden aufgrund vieler Faktoren nur die

wesentlichen Informationen eines Bildes und deren Bewegung und Veränderung im zeitlichen Geschehen des Videos genutzt. Alle übrigen Faktoren werden dann durch Prognosen und Schätzungen sowie Informationen aus zuvor decodierten Frames errechnet.

Die Core Coding Tools^[A1] von AV1, VP9 und AVC werden hier als Beispiel genutzt um sowohl die bestehenden Codierungstechniken zu erläutern als auch deren Weiterentwicklung:

Partitions



Zuerst partitioniert der Videocodier ein Frame eines Videos in *Coding Blocks* verschiedener Größen^[A1]. Diese Blocks werden an den Inhalt des Videos angepasst, um Teile, die der Codec als wichtig beurteilt, eine genauere Partitionierung zu geben und somit mehr Details erhalten, gegenüber anderen

Teilen, wie dem Hintergrund, die der Codec als weniger wichtig beurteilt, welche dann in größere Blocks eingeteilt werden.

AV1 nutzt *Superblocks* bis zu 128x128 Pixel die in 10 verschiedenen Möglichkeiten unterteilt werden können, sowie auch rekursiv selbst weitere Block enthalten können (mit „R“ markierte Teile in der Grafik).^[A1] Ein in Blocks unterteilter Frame nennt man *Partition tree*, da, wie auch bei einem Baum in der Informatik, ein Knoten mit weiteren Unterknoten verbunden ist, ein Block weitere Blocks enthält. Je nachdem welcher Detailgrad von dem Codec erhalten werden soll, sind *Partition trees* entweder knapp und grob oder tief und ausführlich.

Wenn der Codec nicht genügend Zeit zur Verfügung hat oder die ausgewählte Qualität oder Bitrate es nicht zulässt müssen zwangsläufig größere Partitionen genutzt werden, was zu visuell sichtbaren Blockartefakten sorgt.



AVC-typische sichtbare Block-Artefakte

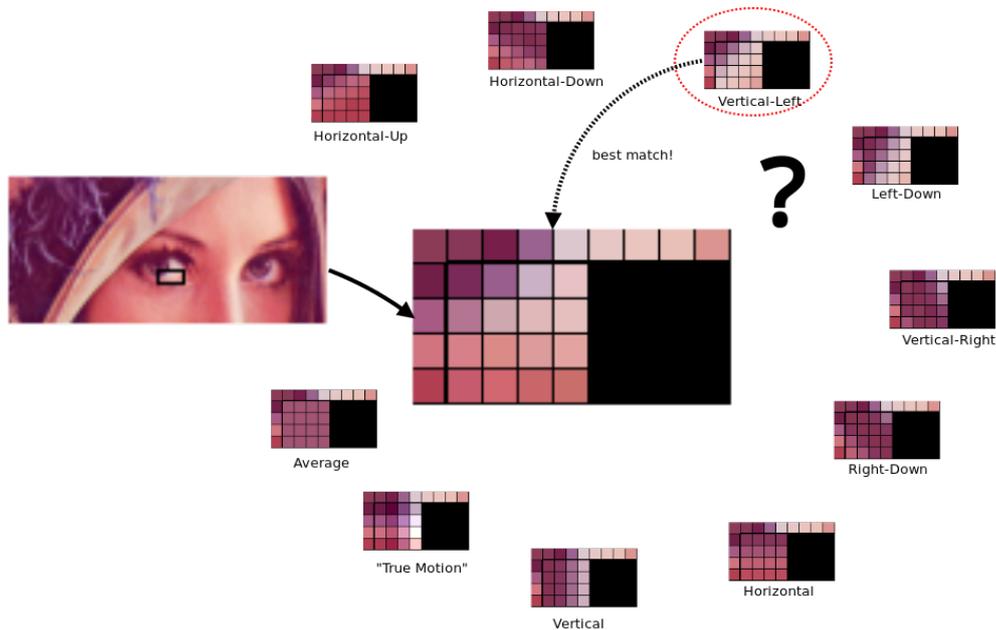
Man erkennt deutlich die durch eine begrenzte Qualität sichtbaren Block-Artefakte. In diesem Fall fokussiert der Codec sich auf die Brücke, insbesondere das Steinmuster im vorderen Teil, sowie auf die Landschaft und Gebäude, währenddessen Details der Wolken und des Wassers als auch die Texturen der Umgebung als weniger wichtig eingestuft wurden und eine größere Partitionierung erhalten.

Intra Prediction (Spatial Redundancy)

Bei Intra Prediction sucht der Codec nach Mustern und Spatial Redundancy innerhalb des gleichen Frames. Die mit Intra Prediction codierten Frames nennt man auch I-frame. Ein Codec besitzt meist mehrere Prediction Modi für sowohl Luma (Helligkeit) als auch Chroma (Farbe), wobei mehr Fokus auf Luma gelegt wird, da das menschliche Auge eher Unterschiede in der Helligkeit als in der Farbe erkennen kann. Auch in anderen Aspekten der Videocodierung findet oft eine Separierung zwischen Luma und Chroma statt um eine effizientere Codierung bei subjektiv gleicher Qualität zu ermöglichen.^[C5]

Das bedeutet der Codec kann beispielsweise in einem 4x4 Block basierend auf verschiedenen Prognose Methoden Pixel Werte generieren, anstatt die Farbinformationen für alle Pixel abzuspeichern. Der Encoder testet die verschiedenen Prediction Modi und wählt den aus, der dem Original am

ähnlichsten sieht. Der Decoder muss dann lediglich diese Pixel basierend auf dem gespeichertem Prediction Modus generieren.^[C6]



Darstellendes Diagramm der Prediction Modi in verlustbehaftetem WebP

Diese Komprimierungstechnik wird auch in Imagecodecs wie *PNG*, *JPEG* oder *WebP* verwendet. Das mittlerweile sehr bekannte *WebP* Format stammt von dem Intra Prediction Aspekt von *VP9*'s Vorgänger *VP8*.^[C5] Ebenso entstammen *HEIC/HEIF* aus *HEVC* und *AVIF* aus *AV1*.

Inter Prediction (Temporal Redundancy)

Bei der Inter Prediction nutzt der Codec bereits codierte I-frames, auch Reference Frame genannt, und codiert ein Bild mithilfe von vorherigen Reference Frames (P-frame) oder sowohl auch mit zukünftigen Reference Frames (B-frame).^[C7]

Da sich I-frames am wenigsten komprimieren lassen werden sie seltener Genutzt. P-frames sind doppelt so effizient wie I-frames und werden auch als Referenz genutzt. B-frames sind am effizientesten, aber es entstehen auch mehr Fehler, weswegen sie weniger als Referenz genutzt werden und mehr als Anhaltspunkte und Unterstützung für eine besser Ausnutzung der zeitlichen Redundanz.^[C7]

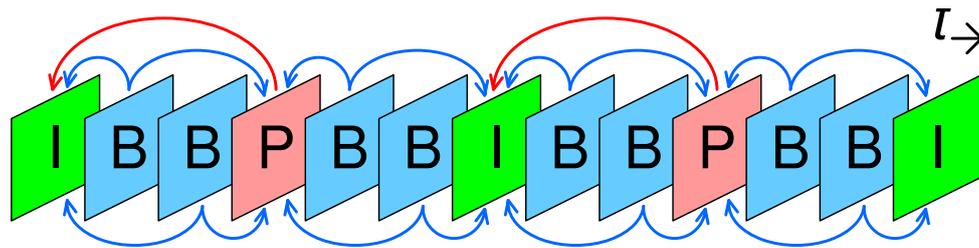


Illustration der Beziehungen der Frames in einer traditionellen Group of pictures Struktur (In modernen Codecs wird eine flexiblere Struktur genutzt)

Dabei werden nach der Partitionierung ähnliche Blocks in den I-frames, auch Reference Frame genannt, gesucht und der nächste Nachbar genutzt, den der Codec finden kann. Dieser wird dann mittels eines Motion Vectors^[A3], welcher die relative Positionsänderung angibt, referenziert.



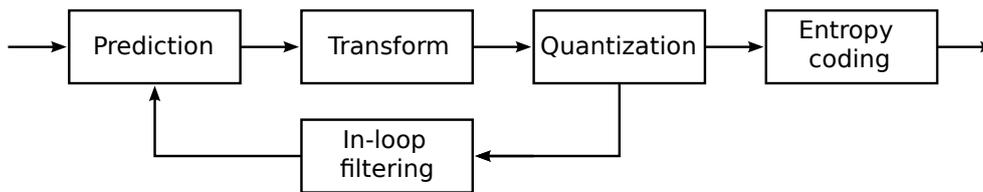
Visualisierte Motion Vectors die die Bewegung der Charaktere zeigen

Da sich dieser oft von dem Original unterscheidet werden die Unterschiede zwischen der Referenz und dem Original, bekannt als Prediction Error^[A3], zusätzlich codiert.

Mithilfe des Motion Vector und dem Prediction Error kann der Decoder die Blöcke eines P- oder B-frame aus den Reference Frames zusammensetzen. Dieser Vorgang spart an Daten und ist wesentlich effizienter als die Bilddaten eines jeden Frames allein mittels Intra Prediction zu bestimmen. Ein B-frame ist etwa nur 25 % so groß wie ein I-frame.

Encoding Stages

Nachdem die Frames mittels Partitioning und Prediction verarbeitet wurden, findet der Mathematisch-orientierte Teil der Datenkomprimierung statt.^[C8]



Encoder Verarbeitungsschritte

Dieser Teil von Videocodecs ist bei allen aktuellen Codecs identisch und es finden sich nur wenige Unterschiede in der Vorgehensweise.

Lediglich beim In-loop Filtering finden sich substantielle Unterschiede zwischen modernen Videocodecs.^[A1, A2]

Transformation & Quantization

Transformationskodierung wird hauptsächlich genutzt, um eine bessere Quantisierung zu ermöglichen. Der Vorgang besteht darin das Ergebnis der Prediction in Koeffizienten umzuwandeln, welche dann durch Quantization verkleinert werden.

Quantization ist der Teil der verlustbehafteten Videocodierung, bei dem die meisten Informationen verloren gehen.^[B2, C9]

Das Konzept basiert darauf, dass wir zwar einfache Helligkeitsunterschiede feststellen können, aber die genaue Stärke einer großen Veränderung dabei eher unwichtig ist. Mithilfe von einer Diskreten Kosinustransformation^[C9] kann man Partitionierte Blöcke in Frequenz Komponenten umwandeln und durch eine Quantization Matrix teilen, welche dafür sorgt, dass visuell mehr sichtbare Frequenzen eine höhere Auflösung erhalten als beispielsweise hohe oder kleine Frequenzen, die das menschliche Auge schwieriger Unterscheiden kann. Die Größe der Quantization Matrix kann basierend auf der angestrebten Datenrate oder Qualität angepasst werden. Eine größere Matrix sorgt für höhere Auflösung, muss aber ebenfalls in den Bitstream des Codecs eingebettet werden und erhöht somit die Datenrate.^[A3]

Entropy Coding

Entropiekodierung beschreibt das Finden und Auslagern von gleichen Sequenzen in Daten, in dem man sie durch einen Code, der kürzer ist als die eigentliche Sequenz, ersetzt. Später werden dann alle Vorkommnisse dieses Codes durch die eigentliche Sequenz ersetzt, um die Daten wiederherzustellen.^[A3]

In Videocodern findet meist die arithmetische Kodierung Anschluss und wird auf die Quantisierten Daten des Bitstreams angewandt, um diese zusätzlich verlustfrei zu komprimieren.^[C8]

In-loop Filtering

Ein In-loop Filter bezeichnet das Anwenden eines Bildes bzw. Videofilters innerhalb des Coding Loop, sowohl im Encoder als auch im Decoder.^[A1]

Zum Beispiel wird ein Deblocking Filter auf jeden 8x8 Makroblock im AVC Videocodern angewandt, um beim Rekonstruktionsprozess die Kompressions-Artefakte zu minimieren. Gleichzeitig werden gefilterte Frames als Referenz Frames genutzt, um die Bewegungskompensation der nachfolgend codierten Frames zu vereinfachen.^[A2]

Das Ergebnis eines In-loop Filters, wie dem Deblocking Filter, ist eine subjektiv bessere Qualität und/oder eine effizientere Codierung durch Anwendung des Filters auf den Referenz Frame.^[A1]

Neuere Coders wie AV1 haben viele In-loop Filter zur Verfügung und wenden sie entweder situationsbedingt automatisch oder als Ergebnis bestimmter Einstellungen an. Beispielsweise nutzt AV1 einen *Frame Super-resolution* Filter der es erlaubt einen Frame mit einer niedrigeren Auflösung zu speichern und ihn in der Rekonstruktion wieder hochzuskalieren, ohne zu komplex für Hardware basiertes In-loop Filtering zu sein. Diese Art von Methoden sind bekannt dafür besonders in sehr Bandbreite-Restriktiven Umgebungen die subjektive visuelle Qualität zu verbessern, waren jedoch bisher oft zu komplex als das sie in Videocodern praktische Anwendung finden konnten.^[A1]

Film Grain Synthesis

Neuere Coders wie AV1 besitzen die Fähigkeit Bildrauschen zu synthetisieren. Dies ist vorteilhaft, da Bildrauschen aufgrund seiner der willkürlichen Natur schwierig zu komprimieren ist und entweder von dem Codec aufgrund der Qualitätseinstellung als unwichtig eingestuft wird und lediglich als Artefakt über bleibt oder heutzutage meist komplett herausfiltert wird. Alternativ kann der Codec auch das Rauschen mit codieren, wobei er die gesamten temporalen Veränderungen verfolgen und codieren muss, was erheblich mehr Speicher und Encoding Ressourcen benötigt.

AV1 kann den Grad an Bildrauschen einfach als Wert notieren und ihn bei dem Kodierungsprozess herausfiltern. Ohne Bildrauschen kann der Codec

dann wesentlich effizienter arbeiten. Im finalen Rekonstruktionsschritt vom Decoder wird dann das Bildrauschen im Nachhinein basierend auf dem zuvor notiertem Wert einfach künstlich hinzugefügt. Da Bildrauschen einfach zufällig ist, ist es nicht wichtig die genaue Positionsänderungen vom Rauschen wiederzugeben, sondern lediglich den gleichen Grad an Bildrauschen herzustellen. Für das menschliche Auge ist das Rauschen im fertig konstruiertem Video identisch zu dem Original, vorausgesetzt man vergleicht nicht einzelne Frames, sondern das Video als bewegtes Bild.^[A1]

Vergleich zwischen den Videocodecs

Effizienz

Bei der Effizienz gilt üblicherweise neuer ist besser. Der Grund dafür sind das neuere Codecs besonders in den Bereichen des Partitioning, Intra- und Inter Prediction sowie dem Filtering fortgeschrittener als ältere Codecs sind und mehr Coding Techniken besitzen.

Beispielsweise besitzt VP9 10 Intra Prediction Modi, davon können 8 die Bewegung hinsichtlich einer direktionalen Bewegung interpolieren und 2 basieren auf nicht-direktionalen Mustern. AV1 hingegen fügt zu den 8 direktionalen Modi ein modifizierbares Delta von -3 bis +3 hinzu, die 8 direktionalen Modi werden damit zu effektiv 56 direktionalen Modi. Zudem werden noch 3 Smooth Predicators hinzugefügt, bei denen ein Block mittels quadratischer Interpolation in vertikale, horizontale oder in beide Richtungen gefüllt werden kann.^[A1]

Dazu kommen noch viele weitere Tricks und Encoder Optimierungen die für eine wesentlich effizientere Kodierung eines Frames im Intra Prediction Modus kommen.

Geschwindigkeit

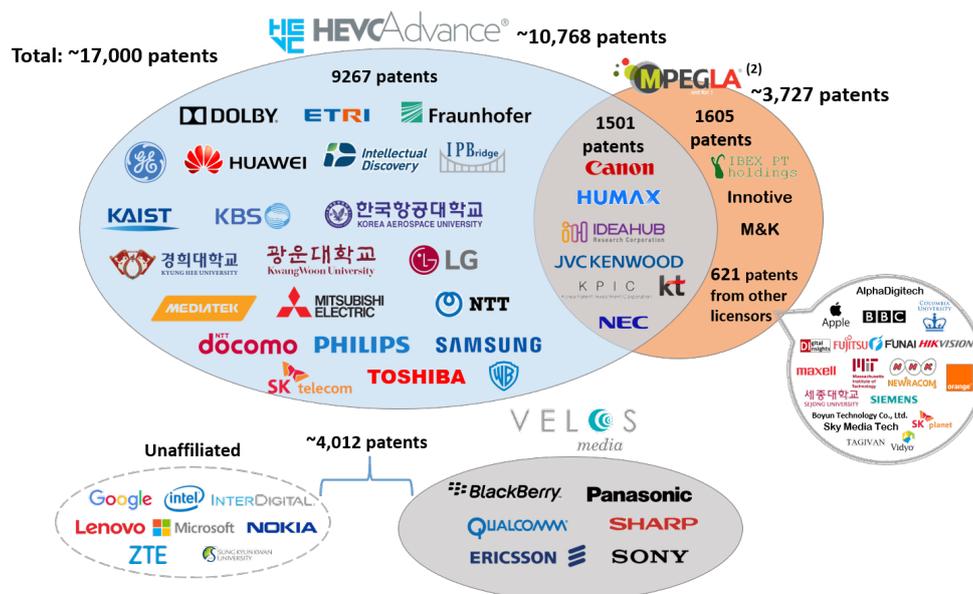
Während bei der Effizienz die neueren Codecs auch eine höhere Effizienz bieten ist es bei der Geschwindigkeit das genaue Gegenteil. Dadurch, dass neuere Codecs genauere Partitions nutzen können, mehr Prediction Modi besitzen und einfach deutlich mehr Coding Techniken nutzen als ältere Codecs, die wesentlich mehr Ressourcen erfordern, dauert das Codieren eines Videos bei neueren Codecs erheblich länger als bei älteren.

Codiert man ein Video auf den bestmöglichen Einstellungen so läuft das Encoding mit AVC in ¼x Echtzeit, mit VP9 in 1/45x Echtzeit, mit HEVC in 1/58x Echtzeit und mit AV1 in 1/45.2216x Echtzeit.^[B1]

Lizenzen & Patente

Damit Codecs wie AVC oder HEVC kommerziell genutzt werden, dürfen muss erst in mehrere Patent-Pools investiert werden, damit eine Plattform tatsächlich Videos mit dem entsprechendem Codec veröffentlichen darf.

Beispielsweise hat HEVC nur eine sehr limitierte Adoption von Anbietern gefunden da nur sehr große Firmen überhaupt einen praktischen Nutzen von den Vorteilen von HEVC gegenüber AVC im Kontext der Kosten der Patente ziehen konnten. Es gibt in etwa 17.000 Patente verteilt über mehrere Patent-Pools, wobei sich einige Verwalter der Patente in gleich 2 Patent Pools befinden und somit mehrfach bezahlt werden. Aufgrund dieses Chaos und den hohen Kosten ist HEVC für viele einfach unpraktisch.^[C10]



Übersicht über die Patent-Pools von HEVC

Adoption & Verfügbarkeit

AVC ist zurzeit der Standard bei so gut wie allen Webvideos und wird wegen seiner guten Effizienz, schnellen Coding Geschwindigkeit und Verfügbarkeit in allen aktuellen Geräten und Plattformen gerne genutzt.^[C11]

Während HEVC mittlerweile zwar bei einigen Diensten und Medien Anwendung gefunden hat, ist die Adoption dennoch sehr träge, obwohl der Codec bereits 2016 verfügbar wurde^[C10]. Streaming Dienste mit vielen

kleineren Videos und jene, die meistens über einen Webbrowser genutzt werden, finden weniger gefallen an HEVC. Beispielsweise nutzt YouTube AVC, VP9 und AV1^[C13], jedoch nicht HEVC. Das liegt vor allem an den Lizenzkosten als auch daran, dass YouTube dann für alle Videos ebenfalls HEVC Encodes erstellen müsste, die jedoch nicht in Webbrowsern abspielbar seien, würden, da, bis auf Apple's Safari, kein aktueller Browser die Wiedergabe von HEVC Videos unterstützt.

AV1 und VP9 hingegen wurden für das Web entwickelt. Während VP9 bereits von allen aktuellen Browsern (mit Ausnahme von Safari auf älteren Plattformen) unterstützt wird, wird AV1 aktuell implementiert und ist bereits in Firefox und Google Chrome, sowie in vielen Chromium-basierten Browsern, seit ~2019 verfügbar. Lediglich Apple's Safari und diverse kleinere oder mobile Browser unterstützten AV1 noch nicht.^[C11]

Zudem starten einige Streaming-Dienste AV1 zu testen. YouTube testen AV1 mit einigen Videos bereits seit 2018^[C13] und Netflix hat seit Anfang 2020^[C12] ebenfalls das experimentelle Testen von AV1 auf Android gestartet.

AV1, im Gegenzug zu AVC, HEVC und VP9, erhielt erst mit den neusten Generationen von Grafikeinheiten Hardwareunterstützung in Konsumer-Geräten, wie beispielsweise Intel's integrierten Grafikeinheiten.^[C14] Daher, dass AV1 mit Software Dekodierung einen erblichen Stromverbrauch hat, wird es auf mobilen Geräten noch nicht bzw. nur in sehr Bandbreite-Restriktiven Umgebungen eingesetzt.^[C12]

Qualitätsvergleich

Vergleich der modernen Codecs VP9 und HEVC mit AV1. AVC wurde in diesem Test nicht genutzt, da dieser Codec nicht bereits 2003 erschien und daher nicht als „modern“ gilt.

BD-Rate

Bjontegaard rate difference, auch als BD-rate bekannt, erlaubt es die Reduktion in Daten bei gleichbleibender visueller Qualität objektiv zu messen und wird besonders bei der Entwicklung und dem Testen von Videocodecs genutzt. Die BD-rate wird aus dem Peak Signal-to-Noise Ratio (PSNR) ermittelt und.^[C15]

Ergebnisse

AV1 übertrifft die VP9 und HEVC Encoder mit durchschnittlich ~30 % in allen Gebieten.^[A1]

TABLE I
BDRATE(%) OF AV1 IN COMPARISON WITH LIBVPX VP9 ENCODER

| Metric \ Set | 1080p | 1080p-screen | 720p | 360p | Average |
|--------------|--------|--------------|--------|--------|---------|
| PSNR-Y | -26.81 | -34.99 | -28.19 | -26.15 | -28.07 |
| PSNR-Cb | -31.27 | -45.86 | -25.42 | -23.77 | -30.10 |
| PSNR-Cr | -31.07 | -42.18 | -27.89 | -31.04 | -31.80 |

TABLE II
BDRATE(%) OF AV1 IN COMPARISON WITH X265 HEVC ENCODER

| Metric \ Set | 1080p | 1080p-screen | 720p | 360p | Average |
|--------------|--------|--------------|--------|--------|---------|
| PSNR-Y | -21.39 | -25.97 | -25.99 | -20.00 | -22.75 |
| PSNR-Cb | -40.23 | -47.57 | -36.87 | -34.89 | -39.18 |
| PSNR-Cr | -40.13 | -41.87 | -38.27 | -41.16 | -40.17 |

Die AV1 Coding Funktionen und Erweiterungen zeigen sich bei allen Auflösungen klar als vorteilhaft. Es ist eindeutig, dass die Effizienz von AV1 alle anderen aktuellen Videocodes übersteigt.

Zusammenfassung

Es stellt sich heraus, dass AOMedia Video 1 nicht nur AVC und HEVC im Thema Lizenzen überragen ist, sondern AVC, HEVC und auch VP9 in Effizienz und Vielseitigkeit schlägt. Nicht nur bietet AV1 moderne Funktionen wie Film Grain Synthesis an und kann in der Theorie mit erheblich mehr Coding Varianten überzeugen, auch bei visueller Qualität schlägt AV1 die anderen Codecs.

Kann also AV1 zum neuen Standard werden und die bereits etablierten Codecs ersetzen? **Noch nicht ganz.**

Wenn es um Effizienz, Vielseitigkeit und Lizenzen geht, hat AV1 den klaren Vorteil. Jedoch ist das Kodieren mit AV1 in Echtzeit bei einer hohen Auflösung nicht möglich. AV1 kann in Echtzeit nur bei niedriger Effizienz und geringen Auflösungen eingesetzt werden.

Wer aktuell sehr viel von AV1 hat, sind Streaming Dienste, die ihre Videos mit AV1 nur einmal optimieren müssen und dann fortan Daten sparen. Mit den neuen Generationen von mobilen Geräten kann AV1 auch effizient abgespielt werden und in den nächsten Jahren wird AV1 wohl zum Standard bei Netflix, YouTube und Co. auf neueren Geräten.

Für schnelles Kodieren in Echtzeitumgebungen und besonders bei reduzierten Ressourcen ist AVC immer noch deutlich besser und bei nicht allzulangen Encodierungen mit hoher Effizienz ist HEVC eine gute Wahl.

Heute kann AV1 noch nicht gegen die Geschwindigkeit der anderen Codecs ankommen, jedoch wird es nicht mehr allzu lange dauern bis die Encoder so effizient werden, dass der Effizienzvorteil der Geschwindigkeit gerecht wird und auch Konsument auf AV1 umsteigen.

Quellenverzeichnis

Alle Quellen wurden in der „Wayback Machine“ archiviert und sind unter <https://web.archive.org> abrufbar. Eine Kopie der Facharbeit ist unter <https://langrock.info/facharbeit-2021> verfügbar.

A: Wissenschaftliche Dokumente

A1. Google, An Overview of Core Coding Tools in the AV1 Video Codec

<https://research.google/pubs/pub47585/>

https://jmvalin.ca/papers/AV1_tools.pdf

A2. Gulistan Raja & Muhammad Javed Mirza,
In-loop Deblocking Filter for H.264/AVC Video

https://huyunf.github.io/blogs/2017/11/20/h264_deblocking_algorithm/cr1261.pdf

A3. Thomas Wiegand & Heiko Schwarz,
Part I of Fundamentals of Sourceand Video Coding

<http://iphome.hhi.de/wiegand/assets/pdfs/VBpart1.pdf>

B: Vorträge und Präsentationen

B1. Timothy B. Terriberry, The AV1 Video Codec

<https://2019.linux.conf.au/schedule/presentation/136/>

<https://www.youtube.com/watch?v=qubPzBcYCTw>

C: Webseiten

C1. Video filesize calculator (unkomprimiert, RGB, 3x8bit, 1080p25, 60s)

[https://toolstud.io/video/filesize.php?](https://toolstud.io/video/filesize.php?dimensions_w=1920&dimensions_h=1080&framerate=25&timeduration=60&timeduration_unit=seconds&compression=3000000&specificbitrate=100&specificbitrate_unit=1000000)

[dimensions_w=1920&dimensions_h=1080&framerate=25&timeduration=60&timeduration_unit=seconds&compression=3000000&specificbitrate=100&specificbitrate_unit=1000000](https://toolstud.io/video/filesize.php?dimensions_w=1920&dimensions_h=1080&framerate=25&timeduration=60&timeduration_unit=seconds&compression=3000000&specificbitrate=100&specificbitrate_unit=1000000) (Aufgerufen am 21.03.2021)

C2. Ookla, Global Speeds February 2021

<https://www.speedtest.net/global-index> (Aufgerufen am 21.03.2021)

C3. Video filesize calculator (YouTube HD, 8bmit, 1080p25, 60s)

[https://toolstud.io/video/filesize.php?](https://toolstud.io/video/filesize.php?dimensions_w=1920&dimensions_h=1080&framerate=25&timeduration=60&timeduration_unit=seconds&compression=19290&specificbitrate=100&specificbitrate_unit=1000000)

[dimensions_w=1920&dimensions_h=1080&framerate=25&timeduration=60&timeduration_unit=seconds&compression=19290&specificbitrate=100&specificbitrate_unit=1000000](https://toolstud.io/video/filesize.php?dimensions_w=1920&dimensions_h=1080&framerate=25&timeduration=60&timeduration_unit=seconds&compression=19290&specificbitrate=100&specificbitrate_unit=1000000) (Aufgerufen am 21.03.2021)

C4. Alliance for Open Media

<https://aomedia.org/about/> (Aufgerufen am 11.04.2021)

- C5. Google, WebP Compression Techniques
<https://developers.google.com/speed/webp/docs/compression>
(Aufgerufen am 09.04.2021)
- C6. Dave Marshall, Intra-frame coding
<http://users.cs.cf.ac.uk/Dave.Marshall/Multimedia/node248.html>
(Aufgerufen am 09.04.2021)
- C7. Krishna Rao Vijayanagar, Inter-frame coding
<https://ottverse.com/i-p-b-frames-idr-keyframes-differences-usecases/>
(Aufgerufen am 09.04.2021)
- C8. Wikimedia, Data compression
https://en.wikipedia.org/wiki/Data_compression#Video
(Aufgerufen am 09.04.2021)
- C9. John Wiseman, Quantization
https://web.archive.org/web/20111115004238/http://www.john-wiseman.com/technical/MPEG_tutorial.htm
(Aufgerufen am 09.04.2021)
- C10. Jonathan Easton, HEVC Patent Pool
<https://www.digitaltveurope.com/2020/07/02/hevc-advance-freezes-royalty-rates-on-patent-pool/> (Aufgerufen am 10.04.2021)
- C11. Can I use, AVC, HEVC, VP9 & AV1 Statistiken
<https://caniuse.com/mpeg4>, <https://caniuse.com/hevc>,
<https://caniuse.com/webm>, <https://caniuse.com/av1>
(Aufgerufen am 11.04.2021)
- C12. Netflix Now Streaming AV1 on Android
<https://netflixtechblog.com/netflix-now-streaming-av1-on-android-d5264a515202> (Aufgerufen am 11.04.2021)
- C13. YouTube, AV1 Beta Launch Playlist
<https://www.youtube.com/playlist?list=PLyqf6gJt7KuHBmeVzZteZUINUQAVLwrZS>
(Aufgerufen am 11.04.2021)
- C14. Intel, VA-API Codecs
<https://github.com/intel/intel-vaapi-driver/blob/master/README>
(Aufgerufen am 11.04.2021)
- C15. IETF, Video Codec Testing and Quality Measurement
<https://tools.ietf.org/id/draft-ietf-netvc-testing-06.html#rfc.section.4.2>
(Aufgerufen am 11.04.2021)