NATURE OF SERVICE:
PRIOR ART SEARCH

SCOPE OF SERVICE:
FREEDOM TO OPERATE

TITLE OF SEARCH:

"A Non-Destructive Virtualisation and Programmable Assembly of Rendered Video"

Contents

1. PROJECT SCOPE

TITLE:	**Non-destructive virtualisation and programmable assembly of rendered video**
PRIORITY DATE:	Patent applications granted and/or published in "Provided Jurisdiction" in the last 25 years & PCT applications filed in last 31 Months.
CLIENT REQUEST :	Perform an FTO search on "Non-destructive virtualisation and programmable assembly of rendered video".
TAXONOMY:	**Primary Features:** A. A computer-implemented method for non-destructive virtualisation and programmable assembly of rendered video, comprising: 1. Separating, by one or more processors, a digital video file compliant with an ISO Base Media File Format into (i) structural metadata describing temporal and spatial organisation of the video and (ii) compressed media samples containing audiovisual data 2. Storing the compressed media samples without modification, duplication, or re-encoding, and maintaining addressable byte-range access to said samples 3. Generating a virtualised structural representation of the video that references the compressed media samples via byte offsets and time-aligned indices 4. Creating a machine-readable semantic manifest associated with the virtualised structural representation, the semantic manifest comprising time-aligned descriptors, conceptual labels, and mappings between semantic meaning, temporal ranges, and corresponding byte ranges of the compressed media samples

	5. Receiving an input instruction comprising a natural language prompt or programmatic query 6. Interpreting, by an artificial intelligence system, the input instruction to determine an intended video output 7. Mapping the intended video output to one or more temporal segments defined within the semantic manifest 8. Resolving the temporal segments into corresponding byte ranges of the compressed media samples via the virtualised structural representation 9. Dynamically generating, at runtime and without modifying the compressed media samples, a standards-compliant video stream by constructing container structural metadata that references the resolved byte ranges; and delivering the generated video stream for playback 10. The video stream is assembled deterministically in response to the input instruction without transcoding, re-rendering, or duplicating the compressed media samples, thereby enabling programmable, queryable, and non-destructive interaction with rendered video content.

2. EXECUTIVE SUMMARY

This report provides a Freedom to Operate (FTO) analysis for "Non-destructive virtualisation and programmable assembly of rendered video". The purpose of this analysis is to determine whether the production, sale, and use of "Non-destructive virtualisation and programmable assembly of rendered video" infringe upon any existing patents. Based on our analysis, we have identified several patents that may pose a risk to the commercialization of "Non-destructive virtualisation and programmable assembly of rendered video". The search is focused on patent applications published or granted in "provided jurisdiction" in the last 25 years or PCT applications filed in last 31 months.

Multiple searches were performed on different databases, such as Patseer, Patentscope, USPTO, ESPACENET, AusPat and Google Patents to extract relevant references. The identified references were categorized in 2 sub categories - Relevant References and Closely Related References based on the relevancy criteria discussed further in the report.

We have considered the following 2 patents and 1 patent application as relevant references which may hinder free operations of the subject product in concerned market(s):
1. Published Patent **US 11,308,159 B2**, entitled, "**Dynamic detection of custom linear video clip boundaries**".
2. Published Patent **US 12,149,773 B2**, entitled, "**Voice-based scene selection for video content on a computing device**".
3. Published Patent application **US20250291845 A1,** entitled, "**Artificial intelligence assisted streaming video scene selection**".

3. RISK ASSESSMENT MATRIX (UTILITY):

3.1 RELEVANT REFERENCES:

TAXONOMY	US 11,308,159 B2	US 12,149,773 B2	US20250291845 A1
A. **A computer-implemented method for non-destructive virtualisation and programmable assembly of rendered video, comprising:**	YES	YES	YES
1. **Separating, by one or more processors, a digital video file compliant with an ISO Base Media File Format into (i) structural metadata describing temporal and spatial organisation of the video and (ii) compressed media samples containing audiovisual data**	YES	YES	NO
2. **Storing the compressed media samples without modification, duplication, or re-encoding, and maintaining addressable byte-range access to said samples**	NO	NO	NO
3. **Generating a virtualised structural representation of the video that**	YES	YES	NO

references the compressed media samples via byte offsets and time-aligned indices			
4. **Creating a machine-readable semantic manifest associated with the virtualised structural representation, the semantic manifest comprising time-aligned descriptors, conceptual labels, and mappings between semantic meaning, temporal ranges, and corresponding byte ranges of the compressed media samples**	YES	YES	NO
5. **Receiving an input instruction comprising a natural language prompt or programmatic query**	YES	YES	YES
6. **Interpreting, by an artificial intelligence system, the input instruction to determine an intended video output**	YES	YES	YES
7. **Mapping the intended video output to one or more temporal segments defined within the semantic manifest**	YES	YES	NO

8. **Resolving the temporal segments into corresponding byte ranges of the compressed media samples via the virtualised structural representation**	NO	NO	NO
9. **Dynamically generating, at runtime and without modifying the compressed media samples, a standards-compliant video stream by constructing container structural metadata that references the resolved byte ranges; and delivering the generated video stream for playback**	YES	YES	YES
10. **The video stream is assembled deterministically in response to the input instruction without transcoding, re-rendering, or duplicating the compressed media samples, thereby enabling programmable, queryable, and non-destructive interaction with rendered video content.**	YES	NO	NO

3.2 CLOSELY RELATED REFERENCES:

TAXONOMY	US20250047939A1	AU2011352094B2	US20250390533A1	US11853370B2	US12225269B2
A. **A computer-implemented method for non-destructive virtualisation and programmable assembly of rendered video, comprising:**	YES	YES	YES	YES	YES
1. **Separating, by one or more processors, a digital video file compliant with an ISO Base Media File Format into (i) structural metadata describing temporal and spatial organisation of the video and (ii) compressed media samples containing audiovisual data**	YES	YES	YES	NO	NO
2. **Storing the compressed media samples without modification, duplication, or re-encoding, and maintaining addressable byte-range access to said samples**	NO	NO	NO	NO	NO
3. **Generating a virtualised structural representation of the video that**	NO	NO	NO	NO	NO

references the compressed media samples via byte offsets and time-aligned indices					
4. **Creating a machine-readable semantic manifest associated with the virtualised structural representation, the semantic manifest comprising time-aligned descriptors, conceptual labels, and mappings between semantic meaning, temporal ranges, and corresponding byte ranges of the compressed media samples**	YES	NO	NO	NO	NO
5. **Receiving an input instruction comprising a natural language prompt or programmatic query**	YES	YES	YES	YES	YES
6. **Interpreting, by an artificial intelligence system, the input instruction to determine an intended video output**	YES	YES	YES	YES	YES
7. **Mapping the intended video output to one or more temporal segments defined within the semantic manifest**	NO	NO	YES	NO	NO

8. **Resolving the temporal segments into corresponding byte ranges of the compressed media samples via the virtualised structural representation**	NO	NO	NO	NO	NO
9. **Dynamically generating, at runtime and without modifying the compressed media samples, a standards-compliant video stream by constructing container structural metadata that references the resolved byte ranges; and delivering the generated video stream for playback**	YES	YES	YES	YES	YES
10. **The video stream is assembled deterministically in response to the input instruction without transcoding, re-rendering, or duplicating the compressed media samples, thereby enabling programmable, queryable, and non-destructive interaction with rendered video content.**	NO	YES	NO	NO	NO

4. RELEVANT REFERENCES (UTILITY):

REFERENCE – 1 | **US 11,308,159 B2** | **TITLE:** Dynamic detection of custom linear video clip boundaries
FILING DATE: JUL 27, 2018 | **PUBLICATION DATE:** APR 19, 2022 | **PRIORITY DATE:** JUL 28, 2017
CURRENT ASSIGNEE: COMCAST CABLE COMMUNICATIONS LLC
STATUS: GRANTED | **INFRINGEMENT RISK**: HIGH

RELEVANT TEXT:

TAXONOMY	US 11,308,159 B2
A. **A computer-implemented method for non-destructive virtual representation and programmable assembly of rendered video, comprising:**	**A method comprising:** **receiving a query associated with content, the query comprising a first portion and a second portion;** **determining a first match in content metadata for the first portion;** **determining, based on the first match, a start boundary preceding a time associated with the first match;** **determining a second match in the content metadata for the second portion;** **determining, based on the second match, an end boundary following a time associated with the second match; and** **generating, based on the start boundary and the end boundary, a portion of the content. (Refer: Claim 1)** The method of claim 1, wherein **generating, based on the start boundary and the end boundary, the portion of the content**

	comprises extracting the portion of the content as a video clip or storing a content identifier, the start boundary, and the end boundary. (Refer: Claim 8) **Remark:** *Prior art claim discloses a method of generating a content portion (i.e., assembling and rendering video segments) based on predefined start and end boundaries. In contrast, the present method determines these boundaries dynamically by identifying specific matches within content metadata (virtual representation of content) corresponding to query portions, thereby enabling targeted extraction of content segments/clips based on query-driven boundary identification.*
1. **Separating, by one or more processors, a digital video file compliant with an ISO Base Media File Format into (i) structural metadata describing temporal and spatial organisation of the video and (ii) compressed media samples containing audiovisual data**	**A method comprising:** **receiving a query associated with content, the query comprising a first portion and a second portion;** **determining a first match in content metadata for the first portion;** determining, based on the first match, a start boundary preceding a time associated with the first match; **determining a second match in the content metadata for the second portion;** determining, based on the second match, an end boundary following a time associated with the second match; and **generating,** based on the start boundary and the end boundary, **a portion of the content. (Refer: Claim 1)** **Remark:** *Prior art claim discloses a method of generating a content portion (i.e., media samples containing audiovisual data) by comparing the input query with the structural metadata of the content. This implies a clear*

	separation between the content metadata—representing temporal or spatial organisation of content—and the original content (audiovisual media samples).
2. **Storing the compressed media samples without modification, duplication, or re-encoding, and maintaining addressable byte-range access to said samples**	N/A
3. **Generating a virtualised structural representation of the video that references the compressed media samples via byte offsets and time-aligned indices**	A method comprising: receiving a query associated with content, the query comprising a first portion and a second portion; **determining a first match in content metadata for the first portion;** **determining, based on the first match, a start boundary preceding a time associated with the first match;** **determining a second match in the content metadata for the second portion;** **determining, based on the second match, an end boundary following a time associated with the second match;** and generating, based on the start boundary and the end boundary, a portion of the content. **(Refer: Claim 1)** The method of claim 1, wherein **the content metadata comprises linear content metadata. (Refer: Claim 2)**

	The method of claim 1, wherein generating, based on the start boundary and the end boundary, **the portion of the content comprises extracting the portion of the content as a video clip** or storing a content identifier, the start boundary, and the end boundary. **(Refer: Claim 8)** The method of claim 16, **wherein the first portion of the query comprises a word or phrase and wherein determining the first part of the content metadata that is analogous to the first portion comprises determining, based on the first portion of the query, a second word or phrase in the first part of the content metadata that matches the word or phrase in the first portion of the query. (Refer: Claim 20)** **Remark:** *Prior art claim discloses a method of generating a content portion (i.e., video segment) by comparing the input query with the linear content metadata (virtualised structural representation) of the content/video. The linear metadata could be a transcript aligned with timestamps or markers that indicate when certain words or events occur in the content. Therefore, metadata here is structured, time-aligned descriptive information (like transcripts, words, phrases or markers) that allows the system to locate and extract relevant portions of the content efficiently.*
4. **Creating a machine-readable semantic manifest associated with the virtualised structural representation, the semantic manifest comprising time-aligned**	A method comprising: receiving a query associated with content, the query comprising a first portion and a second portion; **determining a first match in content metadata for the first portion;**

descriptors, conceptual labels, and mappings between semantic meaning, temporal ranges, and corresponding byte ranges of the compressed media samples	**determining, based on the first match, a start boundary preceding a time associated with the first match;** **determining a second match in the content metadata for the second portion;** **determining, based on the second match, an end boundary following a time associated with the second match;** and generating, based on the start boundary and the end boundary, a portion of the content. **(Refer: Claim 1)** The method of claim 1, wherein **the content metadata comprises linear content metadata. (Refer: Claim 2)** The method of claim 16, **wherein the first portion of the query comprises a word or phrase and wherein determining the first part of the content metadata that is analogous to the first portion comprises determining, based on the first portion of the query, a second word or phrase in the first part of the content metadata that matches the word or phrase in the first portion of the query. (Refer: Claim 20)** **Remark:** *Prior art claim discloses a method of generating a content portion by comparing the input query with the linear content metadata (virtualised structural representation) of the content. The linear metadata could be a transcript aligned with timestamps or markers that indicate when certain words or events occur. Therefore, metadata here is structured, time-aligned descriptors (like transcripts, words, phrases or markers) that allows the*

	system to locate and extract relevant portions of the content efficiently.
5. **Receiving an input instruction comprising a natural language prompt or programmatic query**	**A method comprising:** **receiving a query associated with content, the query comprising a first portion and a second portion;** determining a first match in content metadata for the first portion; determining, based on the first match, a start boundary preceding a time associated with the first match; determining a second match in the content metadata for the second portion; determining, based on the second match, an end boundary following a time associated with the second match; and generating, based on the start boundary and the end boundary, a portion of the content. **(Refer: Claim 1)**
6. **Interpreting, by an artificial intelligence system, the input instruction to determine an intended video output**	A method comprising: receiving a query associated with content, the query comprising a first portion and a second portion; **determining a first match in content metadata for the first portion;** **determining, based on the first match, a start boundary preceding a time associated with the first match;** **determining a second match in the content metadata for the second portion;** **determining, based on the second match, an end boundary following**

	a time associated with the second match; and **generating, based on the start boundary and the end boundary, a portion of the content. (Refer: Claim 1)** **Remark:** *Prior art claim discloses a system that interprets a query comprising a first portion and a second portion to identify a requested content portion (intended video output). Specifically, the system determines matches in the content metadata corresponding to the first and second portions of the query, establishes a start point based on the first match and an end point based on the second match, and generates the intended content portion between these boundaries.*
7. **Mapping the intended video output to one or more temporal segments defined within the semantic manifest**	A method comprising: receiving a query associated with content, the query comprising a first portion and a second portion; determining a first match in content metadata for the first portion; **determining, based on the first match, a start boundary preceding a time associated with the first match;** determining a second match in the content metadata for the second portion; **determining, based on the second match, an end boundary following a time associated with the second match;** and **generating, based on the start boundary and the end boundary, a portion of the content. (Refer: Claim 1)** The method of claim 1, wherein **the content metadata comprises linear**

	content metadata. (Refer: Claim 2) The method of claim 16, **wherein the first portion of the query comprises a word or phrase and wherein determining the first part of the content metadata that is analogous to the first portion comprises determining, based on the first portion of the query, a second word or phrase in the first part of the content metadata that matches the word or phrase in the first portion of the query. (Refer: Claim 20)** **Remark:** *Prior art claim discloses a system that interprets a query comprising a first portion and a second portion to identify a requested content portion (intended video output). Specifically, the system determines matches in the content metadata corresponding to the first and second portions of the query, establishes a start point based on the first match and an end point based on the second match, and generates the intended content portion between these boundaries. The system takes the desired content portion by finding the specific time segments (associated with the metadata) in the video that match the query.*
8. **Resolving the temporal segments into corresponding byte ranges of the compressed media samples via the virtualised structural representation**	N/A
9. **Dynamically generating, at runtime and without modifying**	A method comprising: receiving a query associated with content, the query comprising a first

the compressed media samples, a standards-compliant video stream by constructing container structural metadata that references the resolved byte ranges; and delivering the generated video stream for playback	portion and a second portion; determining a first match in content metadata for the first portion; determining, based on the first match, a start boundary preceding a time associated with the first match; determining a second match in the content metadata for the second portion; determining, based on the second match, an end boundary following a time associated with the second match; and **generating, based on the start boundary and the end boundary, a portion of the content. (Refer: Claim 1)** The method of claim 1, wherein **generating, based on the start boundary and the end boundary, the portion of the content comprises extracting the portion of the content as a video clip or storing a content identifier, the start boundary, and the end boundary. (Refer: Claim 8)** **Remark:** *Prior art claim discloses the system dynamically generates a content portion (video stream/clip) based on the start boundary and the end boundary (metadata associated with time).*
10. **The video stream is assembled deterministically in response to the input instruction without transcoding, re-rendering, or duplicating the compressed**	A method compriSing: **receiving a query associated with content, the query comprising a first portion and a second portion;** **determining a first match in content metadata for the first portion;**

media samples, thereby enabling programmable, queryable, and non-destructive interaction with rendered video content.	**determining, based on the first match, a start boundary preceding a time associated with the first match;** **determining a second match in the content metadata for the second portion;** **determining, based on the second match, an end boundary following a time associated with the second match; and** **generating, based on the start boundary and the end boundary, a portion of the content. (Refer: Claim 1)** The method of claim 1, wherein **generating, based on the start boundary and the end boundary, the portion of the content comprises extracting the portion of the content as a video clip or storing a content identifier, the start boundary, and the end boundary. (Refer: Claim 8)** **Remark:** *Prior art claim discloses the system generates a content portion (video stream/clip is assembled) in response to the query (input instruction) thereby enabling programmable and queryable interaction with rendered video content. The portion of the content comprises extracting the portion of the content as a video clip (made up of video streams).*

REFERENCE – 2 | **US 12,149,773 B2** | **TITLE:** Voice-based scene selection for video content on a computing device
FILING DATE: SEP 02, 2022 | **PUBLICATION DATE:** NOV 19, 2024 | **PRIORITY DATE:** AUG 22, 2022
CURRENT ASSIGNEE: GOOGLE LLC
STATUS: GRANTED | **INFRINGEMENT RISK**: HIGH

RELEVANT TEXT:

TAXONOMY	US 12,149,773 B2
A. **A computer-implemented method for non-destructive virtualisation and programmable assembly of rendered video, comprising:**	**A method implemented by one or more processors comprising:** receiving, from a user and via a computing device, a spoken utterance that includes a query; **identifying video content being presented in a vicinity of the user by a media player application** when the spoken utterance is received from the user; **accessing scene metadata associated with the identified video content,** wherein the scene metadata includes, for each of one or more respective scenes in the identified video content, semantic scene description data describing the respective scene and timestamp data identifying one or more locations in the identified video content corresponding to the respective scene; **determining,** based on the query and the scene metadata associated with the identified video content, **whether the query in the spoken utterance is a scene playback request directed to the media player application to play a requested scene in the identified video content;** in response to determining that the query in the spoken utterance is a

	scene playback request, **causing a media control command to be issued to the media player application to cause the media player application to seek to a predetermined location in the identified video content corresponding to the requested scene and identified in the timestamp data of the scene metadata for the identified video content;** and in response to determining that the query in the spoken utterance is not a scene playback request directed to the media player application, causing a non-scene playback request operation to be executed for the query included in the spoken utterance. **(Refer: Claim 1)** **Remark:** *The prior art claim discloses a computer-implemented method for processing a scene playback request directed to a media player application, wherein the application seeks a specific location within the identified video content—corresponding to the requested scene and indicated by timestamp data in the scene metadata—allowing for non-destructive rendering of video content. This approach facilitates programmable assembly of the video by enabling dynamic, metadata-driven navigation to particular scenes without altering the original media.*
1. **Separating, by one or more processors, a digital video file compliant with an ISO Base Media File Format into (i) structural metadata describing temporal and spatial organisation of the video and (ii) compressed media samples containing**	**A method implemented by one or more processors comprising:** **receiving, from a user and via a computing device,** a spoken utterance that includes a query; identifying video content being presented in a vicinity of the user by a media player application when the spoken utterance is received from the user; **accessing scene metadata associated with the identified video content, wherein the scene metadata includes, for each of one or**

audiovisual data	**more respective scenes in the identified video content, semantic scene description data describing the respective scene and timestamp data identifying one or more locations in the identified video content corresponding to the respective scene;** determining, based on the query and the scene metadata associated with the identified video content, whether the query in the spoken utterance is a scene playback request directed to the media player application to play a requested scene in the identified video content; in response to determining that the query in the spoken utterance is a scene playback request, causing a media control command to be issued to the media player application to cause the media player application to seek to a predetermined location in the identified video content corresponding to the requested scene and identified in the timestamp data of the scene metadata for the identified video content; and in response to determining that the query in the spoken utterance is not a scene playback request directed to the media player application, causing a non-scene playback request operation to be executed for the query included in the spoken utterance. **(Refer: Claim 1)** **Remark:** *The prior art claim discloses one or more processors access and separate the scene metadata (structural metadata) from the digital video content (media sample containing audiovisual data). This scene metadata encompasses semantic descriptions, timestamps, and scene boundaries, collectively representing the temporal and spatial organization of the video content.*
2. **Storing the compressed media samples without modification,**	N/A

duplication, or re-encoding, and maintaining addressable byte-range access to said samples	
3. **Generating a virtualised structural representation of the video that references the compressed media samples via byte offsets and time-aligned indices**	A method implemented by one or more processors comprising: receiving, from a user and via a computing device, a spoken utterance that includes a query; identifying video content being presented in a vicinity of the user by a media player application when the spoken utterance is received from the user; **accessing scene metadata associated with the identified video content, wherein the scene metadata includes, for each of one or more respective scenes in the identified video content,** semantic scene description data describing the respective scene and **timestamp data identifying one or more locations in the identified video content corresponding to the respective scene;** determining, based on the query and the scene metadata associated with the identified video content, whether the query in the spoken utterance is a scene playback request directed to the media player application to play a requested scene in the identified video content; in response to determining that the query in the spoken utterance is a scene playback request, causing a media control command to be issued to the media player application to cause the media player application to seek to a predetermined location in the identified video content corresponding to the requested scene and identified in the timestamp data of the scene metadata for the identified video content; and

	in response to determining that the query in the spoken utterance is not a scene playback request directed to the media player application, causing a non-scene playback request operation to be executed for the query included in the spoken utterance. **(Refer: Claim 1)** **Remark:** *The prior art claim discloses one or more processors access and separate the scene metadata—serving as a virtualized structural representation—from the digital video content (media samples). This scene metadata encompasses semantic descriptions and timestamps (time-aligned indices) that delineate the organization of the video content into distinct scenes.*
4. **Creating a machine-readable semantic manifest associated with the virtualised structural representation, the semantic manifest comprising time-aligned descriptors, conceptual labels, and mappings between semantic meaning, temporal ranges, and corresponding byte ranges of the compressed media samples**	A method implemented by one or more processors comprising: receiving, from a user and via a computing device, a spoken utterance that includes a query; identifying video content being presented in a vicinity of the user by a media player application when the spoken utterance is received from the user; **accessing scene metadata associated with the identified video content, wherein the scene metadata includes, for each of one or more respective scenes in the identified video content, semantic scene description data describing the respective scene and timestamp data identifying one or more locations in the identified video content corresponding to the respective scene;** determining, based on the query and the scene metadata associated with the identified video content, whether the query in the spoken utterance is a scene playback request directed to the media player application to play a

	requested scene in the identified video content; in response to determining that the query in the spoken utterance is a scene playback request, causing a media control command to be issued to the media player application to cause the media player application to seek to a predetermined location in the identified video content corresponding to the requested scene and identified in the timestamp data of the scene metadata for the identified video content; and in response to determining that the query in the spoken utterance is not a scene playback request directed to the media player application, causing a non-scene playback request operation to be executed for the query included in the spoken utterance. **(Refer: Claim 1)** **Remark:** *The prior art claim discloses the use of the scene metadata (semantic manifest) which includes semantic descriptions—such as conceptual labels conveying the semantic meaning of each scene—and timestamps that are time-aligned descriptors indicating specific locations within the video content corresponding to each scene.*
5. **Receiving an input instruction comprising a natural language prompt or programmatic query**	**A method implemented by one or more processors comprising:** **receiving, from a user and via a computing device, a spoken utterance that includes a query;** identifying video content being presented in a vicinity of the user by a media player application **when the spoken utterance is received from the user;** accessing scene metadata associated with the identified video content, wherein the scene metadata includes, for each of one or more respective scenes in the identified video content, semantic scene description data

	describing the respective scene and timestamp data identifying one or more locations in the identified video content corresponding to the respective scene; **determining, based on the query** and the scene metadata associated with the identified video content, **whether the query in the spoken utterance is a scene playback request directed to the media player application to play a requested scene in the identified video content;** **in response to determining that the query in the spoken utterance is a scene playback request,** causing a media control command to be issued to the media player application to cause the media player application to seek to a predetermined location in the identified video content corresponding to the requested scene and identified in the timestamp data of the scene metadata for the identified video content; and in response to determining that the query in the spoken utterance is not a scene playback request directed to the media player application, causing a non-scene playback request operation to be executed for the query included in the spoken utterance. **(Refer: Claim 1)**
6. **Interpreting, by an artificial intelligence system, the input instruction to determine an intended video output**	**A method implemented by one or more processors comprising:** receiving, from a user and via a computing device, a spoken utterance that includes a query; **identifying video content being presented in a vicinity of the user by a media player application when the spoken utterance is received from the user;** accessing scene metadata associated with the identified video content, wherein the scene metadata includes, for each of one or more respective

	scenes in the identified video content, semantic scene description data describing the respective scene and timestamp data identifying one or more locations in the identified video content corresponding to the respective scene; **determining, based on the query and the scene metadata associated with the identified video content, whether the query in the spoken utterance is a scene playback request directed to the media player application to play a requested scene in the identified video content;** **in response to determining that the query in the spoken utterance is a scene playback request, causing a media control command to be issued to the media player application to cause the media player application to seek to a predetermined location in the identified video content corresponding to the requested scene and identified in the timestamp data of the scene metadata for the identified video content;** and **in response to determining that the query in the spoken utterance is not a scene playback request directed to the media player application, causing a non-scene playback request operation to be executed for the query included in the spoken utterance. (Refer: Claim 1)** **Remark:** *The prior art claim discloses a method for interpreting whether the query in the spoken utterance (input instruction) is a scene playback request directed to the media player application, to determine intended video output, and subsequently controlling the media player to seek to the corresponding scene (intended video output) in the identified video content, thereby enabling semantic-aware scene navigation and interaction with video content.*

7. **Mapping the intended video output to one or more temporal segments defined within the semantic manifest**	**A method implemented by one or more processors comprising:** receiving, from a user and via a computing device, a spoken utterance that includes a query; identifying video content being presented in a vicinity of the user by a media player application when the spoken utterance is received from the user; accessing scene metadata associated with the identified video content, wherein the scene metadata includes, for each of one or more respective scenes in the identified video content, semantic scene description data describing the respective scene and timestamp data identifying one or more locations in the identified video content corresponding to the respective scene; **determining, based on the query and the scene metadata associated with the identified video content, whether the query in the spoken utterance is a scene playback request directed to the media player application to play a requested scene in the identified video content;** in response to determining that the query in the spoken utterance is a scene playback request, causing a media control command to be issued to the media player application to cause the media player application to seek to a predetermined location in the identified video content corresponding to the requested scene and identified in the timestamp data of the scene metadata for the identified video content; and in response to determining that the query in the spoken utterance is not a scene playback request directed to the media player application, causing a non-scene playback request operation to be executed for the query included

	in the spoken utterance. **(Refer: Claim 1)** **Remark:** *The prior art claim discloses a method for interpreting whether the query in the spoken utterance (input instruction) is a scene playback request directed to the media player application, based on the the query and scene metadata (semantic manifest), means mapping the intended video output to the scene metadata. The system looks at what the user asked, acess metadata about the scenes in the video, including descriptions and timestamps and based on both the query and scene metadata, the system determines whether the user is requesting to directly jump to and play a specific scene (intended video output) in the video.*
8. **Resolving the temporal segments into corresponding byte ranges of the compressed media samples via the virtualised structural representation**	N/A
9. **Dynamically generating, at runtime and without modifying the compressed media samples, a standards-compliant video stream by constructing container structural metadata that references the resolved byte ranges; and delivering the generated video stream for playback**	**A method implemented by one or more processors comprising:** receiving, from a user and via a computing device, a spoken utterance that includes a query; identifying video content being presented in a vicinity of the user by a media player application when the spoken utterance is received from the user; accessing scene metadata associated with the identified video content, wherein the scene metadata includes, for each of one or more respective scenes in the identified video content, semantic scene description data describing the respective scene and timestamp data identifying one or more

	locations in the identified video content corresponding to the respective scene; determining, based on the query and the scene metadata associated with the identified video content, whether the query in the spoken utterance is a scene playback request directed to the media player application to play a requested scene in the identified video content; **in response to determining that the query in the spoken utterance is a scene playback request, causing a media control command to be issued to the media player application to cause the media player application to seek to a predetermined location in the identified video content corresponding to the requested scene and identified in the timestamp data of the scene metadata for the identified video content;** and in response to determining that the query in the spoken utterance is not a scene playback request directed to the media player application, causing a non-scene playback request operation to be executed for the query included in the spoken utterance. **(Refer: Claim 1)** **Remark:** *The prior art claim discloses when the spoken utterance is a scene playback request, the media player application seek to a predetermined location in the identified video content corresponding to the requested scene and identified in the timestamp data of the scene metadata for the identified video content. Therefore, it can be considered that final requested content is assembled and delivered to the user via media player application.*
10. **The video stream is assembled deterministically in response to the input instruction without**	N/A

transcoding, re-rendering, or duplicating the compressed media samples, thereby enabling programmable, queryable, and non-destructive interaction with rendered video content.	

REFERENCE – 3 | **US20250291845 A1** | **TITLE:** Artificial intelligence assisted streaming video scene selection
FILING DATE: MAR 18, 2024 | **PUBLICATION DATE:** MAR 18, 2024 | **PRIORITY DATE:** SEP 18, 2025
CURRENT ASSIGNEE: RISHI KUMAR
STATUS: GRANTED | **INFRINGEMENT RISK**: HIGH

RELEVANT TEXT:

TAXONOMY	US20250291845 A1
A. **A computer-implemented method for non-destructive virtualisation and programmable assembly of rendered video, comprising:**	**A computer-implemented method comprising:** **receiving, from a remote device, a user query describing a scene that is accessible via a streaming video application;** searching, via a machine learning (ML) model and based on the user query, a content database associated with the streaming video application; determining, via the ML model and the user query, **one or more relevant scenes within the content database;** and **displaying, via a display device, the one or more relevant scenes. (Refer: Claim 8)** **Remark:** *The prior art claim discloses a computer-implemented method of providing relevant scenes (programmable assembly of rendered video) from the content database to the user device. The method enables a user to search for and view specific scenes within streaming video content (film, TV episode).*
1. **Separating, by one or more processors, a digital video file**	N/A

compliant with an ISO Base Media File Format into (i) structural metadata describing temporal and spatial organisation of the video and (ii) compressed media samples containing audiovisual data	
2. **Storing the compressed media samples without modification, duplication, or re-encoding, and maintaining addressable byte-range access to said samples**	N/A
3. **Generating a virtualised structural representation of the video that references the compressed media samples via byte offsets and time-aligned indices**	N/A
4. **Creating a machine-readable semantic manifest associated with the virtualised structural representation, the semantic manifest comprising time-aligned descriptors, conceptual labels,**	N/A

and mappings between semantic meaning, temporal ranges, and corresponding byte ranges of the compressed media samples	
5. **Receiving an input instruction comprising a natural language prompt or programmatic query**	**A computer-implemented method comprising:** **receiving, from a remote device, a user query describing a scene that is accessible via a streaming video application;** searching, via a machine learning (ML) model and based on the user query, a content database associated with the streaming video application; determining, via the ML model and the user query, one or more relevant scenes within the content database; and **displaying, via a display device, the one or more relevant scenes. (Refer: Claim 8)** **The computer-implemented method of claim 8, further comprising:** **receiving a user selection from the one or more relevant scenes. (Refer: Claim 9)**
6. **Interpreting, by an artificial intelligence system, the input instruction to determine an intended video output**	A computer-implemented method comprising: receiving, from a remote device, a user query describing a scene that is accessible via a streaming video application; **searching, via a machine learning (ML) model and based on the user query, a content database associated with the streaming video application;**

	determining, via the ML model and the user query, one or more relevant scenes within the content database; and displaying, via a display device, the one or more relevant scenes. **(Refer: Claim 8)** **Remark:** *The prior art claim discloses machine learning model (artificial intelligence system) interprets one or more relevant scenes (intended video outputs) within the content database based on the query (input instruction).*
7. **Mapping the intended video output to one or more temporal segments defined within the ~~semantic manifest~~**	N/A
8. **Resolving the temporal segments into corresponding byte ranges of the compressed media samples via the virtualised structural ~~representation~~**	N/A
9. **Dynamically generating, at runtime and without modifying the compressed media samples, a standards-compliant video stream by constructing container structural metadata that references the resolved byte ranges; and delivering the**	A computer-implemented method comprising: receiving, from a remote device, a user query describing a scene that is accessible via a streaming video application; searching, via a machine learning (ML) model and based on the user query, a content database associated with the streaming video application; **determining,** via the ML model and the user query, **one or more ~~relevant scenes within the content database;~~** and

generated video stream for playback	**displaying, via a display device, the one or more relevant scenes. (Refer: Claim 8)** **Remark:** *The prior art claim discloses a system that identifies and displays one or more relevant scenes (video streams) present within streaming content, such as films or TV episodes, via a display device. Consequently, the system delivers the constituent video streams that make up a scene to display device for playback, thereby enabling programmable and queryable interaction with rendered video content.*
10. **The video stream is assembled deterministically in response to the input instruction without transcoding, re-rendering, or duplicating the compressed media samples, thereby enabling programmable, queryable, and non-destructive interaction with rendered video content.**	N/A

5. CLOSELY RELATED REFERENCES (UTILITY):

REFERENCE – 4 | **US20250047939A1*** | **TITLE:** Machine-Learning Assisted Personalized Real-Time Video Editing and Playback

FILING DATE: OCT 22, 2024 | **PUBLICATION DATE:** FEB 06, 2025 | **PRIORITY DATE:** OCT 22, 2024

CURRENT ASSIGNEE: JASON HENDERSON, JULIA GUZMAN-HENDERSON

STATUS: PENDING| **INFRINGEMENT RISK**: HIGH

Note: * The most recently amended claims are considered for mapping.

RELEVANT TEXT:

TAXONOMY	US20250047939 A1
A. **A computer-implemented method for non-destructive virtualisation and programmable assembly of rendered video, comprising:**	**A method for machine-learning assisted personalized real-time video editing and playback, comprising:** Receiving user preferences via a user input module as voice or text commands during video playback, wherein the user preferences include requests relating to storylines spanning multiple episodes; **Analyzing video content metadata using a machine learning engine, including generating inferred metadata where explicit metadata is insufficient to identify scenes belonging to a requested storyline;** **Filtering the video content to remove irrelevant segments based on the user's preferences;** **Delivering the edited content dynamically in response to real-time user commands** while the user is engaged in a viewing session; Storing the user preferences in a Concierge database and automatically applying the stored preferences when the user views subsequent episodes

	of the video content. **(Refer: Claim 4)** **Remark:** *Prior art claim discloses a system of machine-learning assisted personalized real-time video editing and playback (programmable assembly of rendered video) by receiving user preference or search terms. Upon receiving the search terms, the system analyses the user's preference and compare it to metadata, then delivering the video to the user's device providing the programmable assembly of video.*
1. **Separating, by one or more processors, a digital video file compliant with an ISO Base Media File Format into (i) structural metadata describing temporal and spatial organisation of the video and (ii) compressed media samples containing audiovisual data**	A method for machine-learning assisted personalized real-time video editing and playback, comprising: **Receiving user preferences via a user input module as voice or text commands during video playback, wherein the user preferences include requests relating to storylines spanning multiple episodes;** **Analyzing video content metadata using a machine learning engine, including generating inferred metadata where explicit metadata is insufficient to identify scenes belonging to a requested storyline;** Filtering the video content to remove irrelevant segments based on the user's preferences; Delivering the edited content dynamically in response to real-time user commands while the user is engaged in a viewing session; Storing the user preferences in a Concierge database and automatically applying the stored preferences when the user views subsequent episodes of the video content. **(Refer: Claim 4)** **Remark:** *Prior art claim discloses that the system receives the user's preference and compares them with the metadata that is associated with the video content (i.e. media sample) to identify scenes belonging to a*

	requested storyline. This implies a separation between the metadata and the original video content within the system.
2. **Storing the compressed media samples without modification, duplication, or re-encoding, and maintaining addressable byte-range access to said samples**	N/A
3. **Generating a virtualised structural representation of the video that references the compressed media samples via byte offsets and time-aligned indices**	N/A
4. **Creating a machine-readable semantic manifest associated with the virtualised structural representation, the semantic manifest comprising time-aligned descriptors, conceptual labels, and mappings between semantic meaning, temporal ranges, and corresponding byte ranges of the compressed media samples**	A method for machine-learning assisted personalized real-time video editing and playback, comprising: **Receiving user preferences via a user input module as voice or text commands during video playback, wherein the user preferences include requests relating to storylines spanning multiple episodes;** **Analyzing video content metadata using a machine learning engine, including generating inferred metadata where explicit metadata is insufficient to identify scenes belonging to a requested storyline;** Filtering the video content to remove irrelevant segments based on the user's preferences; Delivering the edited content dynamically in response to real-time user

	commands while the user is engaged in a viewing session; Storing the user preferences in a Concierge database and automatically applying the stored preferences when the user views subsequent episodes of the video content. **(Refer: Claim 4)** The system of claim 1, wherein **the machine the machine learning engine compiles a collection of metadata including tags related to characters, storylines, scenes, or other elements of the video content. (Refer: Claim 3)** **Remark:** *Prior art claim discloses that system uses a machine learning engine to analyze the metadata associated with the video content. If explicit metadata (like scene tags or storyline labels) is insufficient to identify specific scenes or storylines, the system generates inferred metadata—meaning it uses AI to interpret and predict scene classifications or storyline associations. The metadata includes tags or conceptual labels (semantic manifest) related to characters, storylines, scenes, or other elements of the video content.*
5. **Receiving an input instruction comprising a natural language prompt or programmatic query**	**A method for machine-learning assisted personalized real-time video editing and playback, comprising:** **Receiving user preferences via a user input module as voice or text commands during video playback, wherein the user preferences include requests relating to storylines spanning multiple episodes;** Analyzing video content metadata using a machine learning engine, including generating inferred metadata where explicit metadata is insufficient to identify scenes belonging to a requested storyline;

	Filtering the video content to remove irrelevant segments based on the user's preferences; Delivering the edited content dynamically in response to real-time user commands while the user is engaged in a viewing session; Storing the user preferences in a Concierge database and automatically applying the stored preferences when the user views subsequent episodes of the video content. **(Refer: Claim 4)** **Remark:** *Prior art claim discloses that the system contains user input module for receiving the user's preferences (i.e. input instructions) such as requests relating to storylines spanning multiple episodes.*
6. **Interpreting, by an artificial intelligence system, the input instruction to determine an intended video output**	**A method for machine-learning assisted personalized real-time video editing and playback, comprising:** Receiving user preferences via a user input module as voice or text commands during video playback, **wherein the user preferences include requests relating to storylines spanning multiple episodes;** **Analyzing video content metadata using a machine learning engine, including generating inferred metadata where explicit metadata is insufficient to identify scenes belonging to a requested storyline;** **Filtering the video content to remove irrelevant segments based on the user's preferences;** Delivering the edited content dynamically in response to real-time user commands while the user is engaged in a viewing session; Storing the user preferences in a Concierge database and automatically applying the stored preferences when the user views subsequent episodes

	of the video content. **(Refer: Claim 4)** **Remark:** *Prior art claim discloses a machine learning engine (AI system) that analyse (i.e. interpret) the user's preferences (input instructions) and compare them with the metadata to identify scenes (intended video output) belonging to a requested storyline. The system filters the video in real time in response to the user's preferences i.e., remove the undesired video segment from the video. Then the intended video output (edited video content) is delivered to the user's device.*
7. **Mapping the intended video output to one or more temporal segments defined within the semantic manifest**	N/A
8. **Resolving the temporal segments into corresponding byte ranges of the compressed media samples via the virtualised structural representation**	N/A
9. **Dynamically generating, at runtime and without modifying the compressed media samples, a standards-compliant video stream by constructing container structural metadata that references the resolved byte**	**A method for machine-learning assisted personalized real-time video editing and playback, comprising:** Receiving user preferences via a user input module as voice or text commands during video playback, wherein the user preferences include requests relating to storylines spanning multiple episodes; Analyzing video content metadata using a machine learning engine, including generating inferred metadata where explicit metadata is

ranges; and delivering the generated video stream for playback	insufficient to identify scenes belonging to a requested storyline; Filtering the video content to remove irrelevant segments based on the user's preferences; **Delivering the edited content dynamically in response to real-time user commands while the user is engaged in a viewing session;** Storing the user preferences in a Concierge database and automatically applying the stored preferences when the user views subsequent episodes of the video content. **(Refer: Claim 4)** **Remark:** *Prior art claim discloses that the machine assisted real-time (i.e. run time) video editing and playback system that deliver the edited video content (i.e., desired video stream) to the user's device for playback. The edited video content is generated by removing the undesired video segment from the video, based on the user preferences.*
10. **The video stream is assembled deterministically in response to the input instruction without transcoding, re-rendering, or duplicating the compressed media samples, thereby enabling programmable, queryable, and non-destructive interaction with rendered video content.**	N/A

REFERENCE – 5 | **AU2011352094 B2** | **TITLE:** Searching recorded video
FILING DATE: DEC 29, 2011 | **PUBLICATION DATE:** MAY 19, 2016 | **PRIORITY DATE:** DEC 30, 2010
CURRENT ASSIGNEE: PELCO INC
STATUS: GRANTED| **INFRINGEMENT RISK**: MEDIUM

RELEVANT TEXT:

TAXONOMY	AU2011352094 B2
A. **A computer-implemented method for non-destructive virtualisation and programmable assembly of rendered video, comprising:**	**A method for searching video data, the method comprising: receiving a search query from a user through a user interface, wherein the search query includes a plurality of query parameters** indicative of one or more query characteristics that are characteristics of at least one of a query object or a query event associated with the query object; **calculating a distance measure between the plurality of query parameters and each of a plurality of sets of metadata parameters,** each set of the metadata parameters being indicative of at least one of a candidate object or a candidate event, **corresponding to what the query parameters are indicative of, in video data; and providing an indication of one or more video segments through the user interface**, wherein each of the one or more video segments has a corresponding distance measure less than a threshold value. **(Refer: Claim 1)** **Remark:** *Prior art claim discloses a computer-implemented method for searching video data, wherein a user provides a search query that include query parameters, which are then compared with metadata parameters associated with video content to identify relevant video segments. Thus, making the method to be programmable on the basis of providing output video (assembly of rendered video) based upon user query.*

1. **Separating, by one or more processors, a digital video file compliant with an ISO Base Media File Format into (i) structural metadata describing temporal and spatial organisation of the video and (ii) compressed media samples containing audiovisual data**	**A method for searching video data,** the method comprising: receiving a search query from a user through a user interface, wherein the search query includes a plurality of query parameters indicative of one or more query characteristics that are characteristics of at least one of a query object or a query event associated with the query object; calculating a distance measure between the plurality of query parameters and each of **a plurality of sets of metadata parameters, each set of the metadata parameters being indicative of at least one of a candidate object or a candidate event, corresponding to what the query parameters are indicative of, in video data;** and providing an indication of one or more video segments through the user interface, wherein each of the one or more video segments has a corresponding distance measure less than a threshold value. **(Refer: Claim 1)** **Remark:** *Prior art claim discloses a method where a search query containing query parameters—representing characteristics of an object or event in the video—is received. These query parameters are compared to metadata parameters associated with the video data, which describe objects or events within the video (i.e., structural metadata). Based on this comparison, the system identifies and provides relevant video segments whose metadata closely matches the query parameters, using a distance measure threshold. This process effectively separates the video data into metadata (descriptive information about objects/events) and the actual media content, enabling targeted retrieval of specific segments based on the query.*
2. **Storing the compressed media samples without modification, duplication, or re-encoding, and maintaining addressable byte-**	N/A

range access to said samples	
3. **Generating a virtualised structural representation of the video that references the compressed media samples via byte offsets and time-aligned indices**	N/A
4. **Creating a machine-readable semantic manifest associated with the virtualised structural representation, the semantic manifest comprising time-aligned descriptors, conceptual labels, and mappings between semantic meaning, temporal ranges, and corresponding byte ranges of the compressed media samples**	N/A
5. **Receiving an input instruction comprising a natural language prompt or programmatic query**	**A method for searching video data, the method comprising: receiving a search query from a user through a user interface, wherein the search query includes a plurality of query parameters indicative of one or more query characteristics that are characteristics of at least one of a query object or a query event associated with the query object;** calculating a distance measure between the plurality of query parameters and each of a plurality of sets of metadata parameters, each set of the metadata parameters being indicative of at least one of a candidate

	object or a candidate event, corresponding to what the query parameters are indicative of, in video data; and providing an indication of one or more video segments through the user interface, wherein each of the one or more video segments has a corresponding distance measure less than a threshold value. **(Refer: Claim 1)** **Remark:** *Prior art claim disclosed a method that includes receiving a search query from a user (i.e. input instruction) through a user interface, wherein the search query comprises a plurality of query parameters indicative of one or more query characteristics associated with objects or events. This user-provided search query functions as an input instruction that directs the system to identify relevant video content.*
6. **Interpreting, by an artificial intelligence system, the input instruction to determine an intended video output**	**A method for searching video data, the method comprising: receiving a search query from a user through a user interface, wherein the search query includes a plurality of query parameters** indicative of one or more query characteristics that are characteristics of at least one of a query object or a query event associated with the query object; **calculating a distance measure between the plurality of query parameters and each of a plurality of sets of metadata parameters,** each set of the metadata parameters being indicative of at least one of a candidate object or a candidate event, **corresponding to what the query parameters are indicative of, in video data; and providing an indication of one or more video segments through the user interface,** wherein each of the one or more video segments has a corresponding distance measure less than a threshold value. **(Refer: Claim 1)** **Remark:** *Prior art claim discloses a method where a user submits a search query composed of multiple query parameters that represent specific characteristics of objects or events within the video. The system interprets*

	this query by calculating a distance measure between these query parameters and sets of metadata parameters associated with the video data, which describe the objects or events present. Video segments are then identified and presented if their metadata parameters are sufficiently similar to the query parameters—that is, if their calculated distance is below a certain threshold. This process indicates that the system first interprets and extracts meaningful query parameters from the user's input, then compares these parameters with the video's metadata to retrieve relevant segments.
7. **Mapping the intended video output to one or more temporal segments defined within the semantic manifest**	N/A
8. **Resolving the temporal segments into corresponding byte ranges of the compressed media samples via the virtualised structural representation**	N/A
9. **Dynamically generating, at runtime and without modifying the compressed media samples, a standards-compliant video stream by constructing container structural metadata that references the resolved byte ranges; and delivering the**	**A method for searching video data, the method comprising: receiving a search query from a user through a user interface, wherein the search query includes a plurality of query parameters** indicative of one or more query characteristics that are characteristics of at least one of a query object or a query event associated with the query object; calculating a distance measure between the plurality of query parameters and each of a plurality of sets of metadata parameters, each set of the metadata parameters being indicative of at least one of a candidate object or a

generated video stream for playback	candidate event, **corresponding to what the query parameters are indicative of, in video data; and providing an indication of one or more video segments through the user interface,** wherein each of the one or more video segments has a corresponding distance measure less than a threshold value. **(Refer: Claim 1)** The method according to any one of the preceding claims, further comprising: **receiving an indication from the user identifying an indicated video segment in the one or more video segments; retrieving the indicated video segment; and displaying the retrieved video segment to the user. (Refer: Claim 4)** **Remark:** *Prior art claim discloses that the method provides an indication of video segments on a user device upon understanding the user query and comparing it with the metadata parameters that contain the information related to object or event in the video and upon indicating the video segments, the video segments (video streams) are retrieved and displayed or delivered to the user for the playback.*
10. **The video stream is assembled deterministically in response to the input instruction without transcoding, re-rendering, or duplicating the compressed media samples, thereby enabling programmable, queryable, and non-destructive interaction with**	**A method for searching video data, the method comprising: receiving a search query from a user through a user interface,** wherein the search query includes a plurality of query parameters indicative of one or more query characteristics that are characteristics of at least one of a query object or a query event associated with the query object; calculating a distance measure between the plurality of query parameters and each of a plurality of sets of metadata parameters, each set of the metadata parameters being indicative of at least one of a candidate object or a candidate event, **corresponding to what the query parameters are indicative of, in video data; and providing an indication of one or more video**

rendered video content.	**segments through the user interface,** wherein each of the one or more video segments has a corresponding distance measure less than a threshold value. **(Refer: Claim 1)** The method according to any one of the preceding claims, further comprising: **receiving an indication from the user identifying an indicated video segment in the one or more video segments; retrieving the indicated video segment; and displaying the retrieved video segment to the user. (Refer: Claim 4)** **Remark:** *Prior art claim discloses a computer-implemented method for searching video data, wherein a user provides a search query that is processed that include query parameters, which are then compared with metadata parameters associated with video content to identify relevant video segments. Upon identifying the video segments the system provide indication to the user interface. When user selects a particular video segment, the system retrieve (assemble the video streams) and display the identified video segments to the user which makes the method to be programmable and queryable as it displays the video segments only after receiving and interpreting the search query from the user.*

REFERENCE – 6 | **US20250390533A1*** | **TITLE:** Building security system with artificial intelligence video analysis and natural language video searching
FILING DATE: AUG 22, 2025 | **PUBLICATION DATE:** DEC 25, 2025 | **PRIORITY DATE:** AUG 01, 2023
CURRENT ASSIGNEE: TYCO FIRE AND SECURITY GMBH
STATUS: PENDING | **INFRINGEMENT RISK**: MEDIUM
Note: * The most recently amended claims are considered for mapping.

RELEVANT TEXT:

TAXONOMY	US20250390533 A1*
A. **A computer-implemented method for non-destructive virtualisation and programmable assembly of rendered video, comprising:**	**A method for classifying and searching video files** in a building security system, the method comprising: applying classifications to video files using an artificial intelligence (AI) model, the classifications comprising one or more objects or events recognized in the video files by the Al model; **extracting one or more entities from a search query received via a user interface,** the entities comprising one or more objects or events indicated by the search query; **searching the video files using the classifications applied by the AI model and the one or more entities extracted from the search query**; and **presenting one or more of the video files identified as results of the search query as playable videos via the user interface. (Refer: Claim 1)** **Remark:** *Prior art claim discloses a method for classifying and searching video files in a building security system, wherein an artificial intelligence*

	model is used to apply classifications to video content and user queries are processed to extract entities for searching and retrieving relevant video files on the basis of user search query. The method further includes presenting the identified video files (assembly of rendered video) as playable outputs via a user interface making the method to be a programmable one.
1. **Separating, by one or more processors, a digital video file compliant with an ISO Base Media File Format into (i) structural metadata describing temporal and spatial organisation of the video and (ii) compressed media samples containing audiovisual data**	A method for classifying and searching video files in a building security system, the method comprising: **applying classifications to video files using an artificial intelligence (AI) model, the classifications comprising one or more objects or events recognized in the video files by the AI model;** extracting one or more entities from a search query received via a user interface, the entities comprising one or more objects or events indicated by the search query; searching the video files using the classifications applied by the AI model and the one or more entities extracted from the search query; and presenting one or more of the video files identified as results of the search query as playable videos via the user interface. **(Refer: Claim 1)** The method of claim 1, wherein **applying the classifications to the video files comprises:** **processing a timeseries of video frames of a video file recorded over a time period using the AI model to identify an event that begins at a start time during the time period and ends at an end time during the time period**; and **applying a classification to the video file that identifies the event,**

	the start time of the event, and the end time of the event. (Refer: Claim 6) **Remark:** *Prior art claim discloses a method for classifying and searching video files in a building security system which uses an AI model that applies classifications to identify objects or events within the video. These classifications, along with identified start and end times of events, can be inferred as metadata describing the start and end time of an event (i.e. temporal aspects) and identifying objects and events from the video (i.e. semantic aspects) of the video content.*
2. **Storing the compressed media samples without modification, duplication, or re-encoding, and maintaining addressable byte-range access to said samples**	N/A
3. **Generating a virtualised structural representation of the video that references the compressed media samples via byte offsets and time-aligned indices**	N/A
4. **Creating a machine-readable semantic manifest associated with the virtualised structural representation, the semantic**	N/A

manifest comprising time-aligned descriptors, conceptual labels, and mappings between semantic meaning, temporal ranges, and corresponding byte ranges of the compressed media samples	
5. **Receiving an input instruction comprising a natural language prompt or programmatic query**	A method for classifying and searching video files in a building security system, the method comprising: applying classifications to video files using an artificial intelligence (AI) model, the classifications comprising one or more objects or events recognized in the video files by the Al model; extracting **one or more entities from a search query received via a user interface,** the entities comprising one or more objects or events indicated by the search query; searching the video files using the classifications applied by the AI model and the one or more entities extracted from the search query; and presenting one or more of the video files identified as results of the search query as playable videos via the user interface. **(Refer: Claim 1)** The method of claim 1, **wherein the search query is a natural language search query comprising freeform text or verbal inputs provided by a user via the user interface;** and the method comprises **extracting the one or more entities from the natural language search query using natural language processing.**

	(Refer: Claim 3) **Remark:** *Prior art claim discloses* a method *for classifying and searching video files in a building security system, wherein the system receives a natural language search query (i.e. input instruction) via a user interface and extracts one or more entities using natural language processing.*
6. **Interpreting, by an artificial intelligence system, the input instruction to determine an intended video output**	**A method for classifying and searching video files** in a building security system, the method comprising: applying classifications to video files using an artificial intelligence (AI) model, the classifications comprising one or more objects or events recognized in the video files by the Al model; **extracting one or more entities from a search query received via a user interface, the entities comprising one or more objects or events indicated by the search query;** **searching the video files using the classifications applied by the AI model and the one or more entities extracted from the search query;** and presenting one or more of the video files identified as results of the search query as playable videos via the user interface. **(Refer: Claim 1)** **Remark:** *Prior art claim discloses a method for classifying and searching video files, where the system applies classifications to video files using an AI model and extracts one or more entities from a natural language search query and provide the output as a video file (intended video output). This AI-driven interpretation of user input on the basis of user search query leads to video retrieval as playable video.*

7. **Mapping the intended video output to one or more temporal segments defined within the semantic manifest**	**A method for classifying and searching video files** in a building security system, the method comprising: **applying classifications to video files using an artificial intelligence (AI) model,** the classifications comprising one or more objects or events recognized in the video files by the Al model; extracting one or more entities from a search query received via a user interface, the entities comprising one or more objects or events indicated by the search query; **searching the video files using the classifications applied by the AI model and the one or more entities extracted from the search query; and** **presenting one or more of the video files identified as results of the search query as playable videos via the user interface. (Refer: Claim 1)** The method of claim 1, **comprising cutting the video files to create one or more snippets of the video files based on an output of the AI model indicating one or more times at which the one or more entities extracted from the search query appear in the video files; and** **presenting the one or more snippets of the video files as the results of the search query via the user interface. (Refer: Claim 9)** The method of claim 1, wherein applying the classifications to the video files comprises: **processing a timeseries of video frames of a video file recorded over**

	a time period using the AI model to identify an event that begins at a start time during the time period and ends at an end time during the time period; and **applying a classification to the video file that identifies the event, the start time of the event, and the end time of the event. (Refer: Claim 6)** **Remark:** *Prior art claim discloses that the system applies the classifications to video files using the AI model, where the video files (intended video output) are searched on the basis of classification and the query of the user. The classifcations are used to identify the event in the video file, start time of the event and the end time of the event corresponding to the semantic manifest. The output video is cutted down into snippets (i.e. segments) corresponding to the times these entities appear which corresponds to the mapping of intended video output to temporal segments.*
8. **Resolving the temporal segments into corresponding byte ranges of the compressed media samples via the virtualised structural representation**	N/A
9. **Dynamically generating, at runtime and without modifying the compressed media samples, a standards-compliant video stream by constructing container structural metadata that**	**A method for classifying and searching video files** in a building security system, the method comprising: applying classifications to video files using an artificial intelligence (AI) model, the classifications comprising one or more objects or events recognized in the video files by the Al model; extracting one or more entities from a search query received via a user

references the resolved byte ranges; and delivering the generated video stream for playback	interface, the entities comprising one or more objects or events indicated by the search query; **searching the video files using the classifications applied by the AI model and the one or more entities extracted from the search query; and** **presenting one or more of the video files identified as results of the search query as playable videos via the user interface. (Refer: Claim 1)** **Remark:** *Prior art claim discloses a method for classifying and searching video files in a building security system, where the video files (video stream) is presented as an result for playback via user interface in response or as a result of the search query (i.e. input instruction) of the user.*
10. **The video stream is assembled deterministically in response to the input instruction without transcoding, re-rendering, or duplicating the compressed media samples, thereby enabling programmable, queryable, and non-destructive interaction with rendered video content.**	N/A

REFERENCE – 7 | **US 11,853,370 B2** | **TITLE:** Scene aware searching
FILING DATE: OCT 27, 2022 | **PUBLICATION DATE:** DEC 26, 2023 | **PRIORITY DATE:** JUN 07, 2017
CURRENT ASSIGNEE: ADEIA MEDIA HOLDINGS LLC
STATUS: GRANTED | **INFRINGEMENT RISK**: MEDIUM

RELEVANT TEXT:

TAXONOMY	US11853370 B2
A. **A computer-implemented method for non-destructive virtualisation and programmable assembly of rendered video, comprising:**	**A method, comprising:** **receiving, at an artificial intelligence (AI) engine, a first search query from a user;** determining whether the first search query is related to a video stream that is currently being consumed by the user; **in response to determining that the first search query is related to the video stream** that is currently being consumed by the user: **determining context related to the first search query, wherein the determination comprises analyzing one or more objects in a frame of the video stream currently being consumed by the user;** refining, by the AI engine, the context determined based on the first search query in response to receiving a second search query, wherein the second search query is related to the first search query; identifying, by the AI engine, one or more matches based on the refined context, wherein the one or more matches are entries in one or more data lakes of a database; and **displaying results of the identified matches. (Refer: Claim 1)**

	Remark: *Prior art claim discloses an AI-driven system that enables context-aware interaction with a video while it is being watched, where user queries are interpreted in relation to the current visual content. It works by analyzing frames of the ongoing video stream to detect objects and scene context, linking user queries to what is currently visible, and refining that understanding through follow-up queries before retrieving relevant results from data lakes which aligns with the functioning of programmable assembly of video data.*
1. **Separating, by one or more processors, a digital video file compliant with an ISO Base Media File Format into (i) structural metadata describing temporal and spatial organisation of the video and (ii) compressed media samples containing audiovisual data**	N/A
2. **Storing the compressed media samples without modification, duplication, or re-encoding, and maintaining addressable byte-range access to said samples**	N/A
3. **Generating a virtualised structural representation of the video that references the**	N/A

compressed media samples via byte offsets and time-aligned indices	
4. **Creating a machine-readable semantic manifest associated with the virtualised structural representation, the semantic manifest comprising time-aligned descriptors, conceptual labels, and mappings between semantic meaning, temporal ranges, and corresponding byte ranges of the compressed media samples**	N/A
5. **Receiving an input instruction comprising a natural language prompt or programmatic query**	**A method, comprising:** **receiving, at an artificial intelligence (AI) engine, a first search query from a user;** determining whether the first search query is related to a video stream that is currently being consumed by the user; in response to determining that the first search query is related to the video stream that is currently being consumed by the user: determining context related to the first search query, wherein the determination comprises analyzing one or more objects in a frame of the video stream currently being consumed by the user; ~~refining, by the AI engine, the context determined based on the first search~~

	query in response to receiving a second search query, wherein the second search query is related to the first search query; identifying, by the AI engine, one or more matches based on the refined context, wherein the one or more matches are entries in one or more data lakes of a database; and Displaying results of the identified matches. **(Refer: Claim 1)** **Remark:** *Prior art claim discloses that the artificial intelligence (AI) engine receives the user query (i.e. input instruction)*
6. **Interpreting, by an artificial intelligence system, the input instruction to determine an intended video output**	A method, comprising: receiving, **at an artificial intelligence (AI) engine**, a first search query from a user; determining whether the first search query is related to a video stream that is currently being consumed by the user; in response to determining that the first search query is related to the video stream that is currently being consumed by the user: **determining context related to the first search query, wherein the determination comprises analyzing one or more objects in a frame of the video stream currently being consumed by the user;** **refining, by the AI engine, the context determined based on the first search query in response to receiving a second search query, wherein the second search query is related to the first search query;** **identifying, by the AI engine, one or more matches based on the refined context, wherein the one or more matches are entries in one or more data lakes of a database; and**

	displaying results of the identified matches. (Refer: Claim 1) **Remark:** *Prior art claim discloses that the artificial intelligence (AI) engine receives the user query and determine the context related to that query by analysing the objects in the video frames and later the AI engine identify the matches based on the determined context with the matches that are present in the database. Upon identifying and matching the context from the database, it displays the results corresponding to the matches corresponding to determining the output on the basis of received query.*
7. **Mapping the intended video output to one or more temporal segments defined within the semantic manifest**	N/A
8. **Resolving the temporal segments into corresponding byte ranges of the compressed media samples via the virtualised structural representation**	N/A
9. **Dynamically generating, at runtime and without modifying the compressed media samples, a standards-compliant video stream by constructing container structural metadata that references the resolved byte**	A method, comprising: receiving, at an artificial intelligence (AI) engine, a first search query from a user; determining whether the first search query is related to a video stream that is currently being consumed by the user; in response to determining that the first search query is related to the video stream that is currently being consumed by the user: determining context

ranges; and delivering the generated video stream for playback	related to the first search query, wherein the determination comprises analyzing one or more objects in a frame of the video stream currently being consumed by the user; refining, by the AI engine, the context determined based on the first search query in response to receiving a second search query, wherein the second search query is related to the first search query; identifying, by the AI engine, **one or more matches based on the refined context, wherein the one or more matches are entries in one or more data lakes of a database; and** **displaying results of the identified matches. (Refer: Claim 1)** **Remark:** *Prior art claim discloses a method that where the artificial intelligence (AI) engine receives the user query. In response to the user query the context is being determined by analysing the objects in the frame of the video stream and the AI engine receives the second query where the second query is related to the first query. By matching the determined context with matches available in the database the system displays the result corresponding to the assembling the video or output in response to the query (i.e. input instruction) for playback.*
10. **The video stream is assembled deterministically in response to the input instruction without transcoding, re-rendering, or duplicating the compressed media samples, thereby enabling programmable, queryable, and non-destructive interaction with rendered video content.**	N/A

REFERENCE – 8 | **US12225269 B2** | **TITLE:** Methods, systems, and apparatuses to respond to voice requests to play desired video clips in streamed media based on matched close caption and sub-title text
FILING DATE: NOV 06, 2023 | **PUBLICATION DATE:** FEB 11, 2025 | **PRIORITY DATE:** FEB 14, 2020
CURRENT ASSIGNEE: DISH NETWORK TECH INDIA PRIVATE LIMITED
STATUS: GRANTED | **INFRINGEMENT RISK**: MEDIUM

RELEVANT TEXT:

TAXONOMY	US12225269B2
A. **A computer-implemented method for non-destructive virtualisation and programmable assembly of rendered video, comprising:**	**A method, comprising:** **converting a voice request received at a local device to text;** executing, by a server in communication with the local device, a search on a media database to identify media content that matches the text; and **Playing, on the local device, a video in response to the search identifying the video as matching the text. (Refer: Claim 1)** **Remark:** *Prior art claim discloses delivering a video stream for playback directly in response to the input instruction, which aligns with the assembly of rendered video.*
1. **Separating, by one or more processors, a digital video file compliant with an ISO Base Media File Format into (i) structural metadata describing temporal and spatial organisation of the video and (ii) compressed media**	N/A

samples containing audiovisual data	
2. **Storing the compressed media samples without modification, duplication, or re-encoding, and maintaining addressable byte-range access to said samples**	N/A
3. **Generating a virtualised structural representation of the video that references the compressed media samples via byte offsets and time-aligned indices**	N/A
4. **Creating a machine-readable semantic manifest associated with the virtualised structural representation, the semantic manifest comprising time-aligned descriptors, conceptual labels, and mappings between semantic meaning, temporal ranges, and corresponding byte ranges of the compressed media samples**	N/A
5. **Receiving an input instruction**	A method, comprising:

comprising a natural language prompt or programmatic query	**converting a voice request received at a local device to text;** executing, by a server in communication with the local device, **a search on a media database to identify media content that matches the text;** and Playing, on the local device, a video in response to the search identifying the video as matching the text. **(Refer: Claim 1)** The method of claim 1, **wherein converting the voice request to the text further comprises applying a natural language processing (NLP) to convert the voice request to the text. (Refer: Claim 2)** **Remark:** *Prior art claim discloses a method of receiving and converting a voice request to text, which directly corresponds to receiving a query, and further specifies applying NLP to convert the voice request to text, reinforcing the natural language processing aspect.*
6. **Interpreting, by an artificial intelligence system, the input instruction to determine an intended video output**	A method, comprising: **converting a voice request received at a local device to text;** **executing,** by a server in communication with the local device, **a search on a media database to identify media content that matches the text;** and Playing, on the local device, a video in response to the search identifying the video as matching the text. **(Refer: Claim 1)** **Remark:** *Prior art claim discloses the input voice request (input instruction), converts it to text, and then searches the media database. This process directly determines which video output is intended by the user.*
7. **Mapping the intended video output to one or more temporal**	N/A

segments defined within the semantic manifest	
8. **Resolving the temporal segments into corresponding byte ranges of the compressed media samples via the virtualised structural representation**	N/A
9. **Dynamically generating, at runtime and without modifying the compressed media samples, a standards-compliant video stream by constructing container structural metadata that references the resolved byte ranges; and delivering the generated video stream for playback**	A method, comprising: converting a voice request received at a local device to text; executing, by a server in communication with the local device, a search on a media database to identify media content that matches the text; and **Playing, on the local device, a video in response to the search identifying the video as matching the text. (Refer: Claim 1)** **Remark:** *Prior art claim discloses that the device plays the video directly in response to the user's request, which constitutes the deterministic assembly of a video stream from stored media database.*
10. **The video stream is assembled deterministically in response to the input instruction without transcoding, re-rendering, or duplicating the compressed media samples, thereby enabling programmable, queryable, and non-destructive interaction with rendered video content.**	N/A

6. KEY STRINGS

PATSEER:

Sr. No.	Key Strategies
1	C:((VIDEO W3 (PLAYBACK OR VIRTUALIZATION OR REPRESENTATION OR PLAYBACK OR RENDER*))) AND C:((VIDEO W3 (SEGMENT* OR PORTION* OR FRAME*))) AND C:((DYNAMIC W2 VIDEO W2 ASSEMBLY) OR (REAL-TIME W2 RENDERING))
2	TAC:((VIDEO W3 (VIRTUALISATION OR RENEDER* OR ASSEMBL*))) OR TAC:((VIDEO W3 (SEPARATE OR SEPARATION OR ABSTRACTION OR REPRESENTATION))) OR TACD:(((AI OR "ARTIFICIAL INTELLIGENCE" OR "MACHINE LEARNING" OR "NEURAL NETWORK") W5 VIDEO))
3	TAC:((VIDEO W3 (MEDIA OR VIRTUALISATION OR RENEDER* OR ASSEMBL* OR STRUCTURE OR METADATA OR REPRESENTATION))) AND TAC:((VIDEO WS (SEGMENT* OR PORTION* OR PARTS OR INDEX*))) AND TAC:(((VIDEO W5 (DUPLICATE OR RE-RENDER* OR STORAGE)) WS (REDUCE OR ELIMINATE OR AVOID))) AND TACD:((AI OR "ARTIFICIAL INTELLIGENCE" OR "MACHINE LEARNING"))
4	TAC:(((VIDEO W3 (MEDIA OR VIRTUALISATION OR RENEDER* OR ASSEMBL* OR STRUCTURE OR METADATA OR REPRESENTATION)))) AND TAC:(((VIDEO WS (SEGMENT* OR PORTION* OR PARTS OR INDEX*)))) AND TAC:((((VIDEO W5 (DUPLICATE OR RE-RENDER* OR STORAGE)) WS (REDUCE OR ELIMINATE OR AVOID)))) AND TAC:((MACHINE W2 LEARNING OR NEURAL W2 NETWORK OR AI OR CNN))
5	TAC:(((VIDEO W3 (MEDIA OR VIRTUALISATION OR RENEDER* OR ASSEMBL* OR STRUCTURE OR METADATA OR REPRESENTATION)))) AND TAC:(((VIDEO WS (SEGMENT* OR PORTION* OR PARTS OR INDEX*)))) AND TAC:(((USER W4 (INTENT OR INSTRUCTION OR QUARY OR INPUT OR COMMAND)) WS (AI OR CNN OR MACHINE-LEARNING OR MODEL)))

6	TAC:((VIDEO W3 (MEDIA OR VIRTUALISATION OR RENEDER* OR ASSEMBL* OR STRUCTURE OR METADATA OR REPRESENTATION))) AND TAC:((VIDEO WS (SEGMENT* OR PORTION* OR PARTS OR INDEX*))) AND AC:((H04N21/00 OR H04N21/234 OR H04N21/266)) AND TAC:(((VIDEO OR MEDIA) W5 (DUPLICATE OR DUPLICATION OR STORAGE)))
7	TAC:(((VIDEO W3 (MEDIA OR VIRTUALISATION OR RENEDER* OR ASSEMBL* OR STRUCTURE OR METADATA OR REPRESENTATION)))) AND TAC:(((VIDEO WS (SEGMENT* OR PORTION* OR PARTS OR INDEX*)))) AND TAC:((ASSEMBLY OR COMPOSITION OR RECOMPOSITION)) AND AC:((H04N21/00 OR H04N21/234 OR H04N21/266 OR G06N20/00 OR G06N3/08 OR G06N5/00))
8	C:(((VIDEO W3 (MEDIA OR VIRTUALISATION OR RENEDER* OR ASSEMBL* OR STRUCTURE OR METADATA OR REPRESENTATION)))) AND C:(((VIDEO WS (SEGMENT* OR PORTION* OR PARTS OR INDEX*)))) AND C:(((USER W4 (INTENT OR INSTRUCTION OR QUARY OR INPUT OR COMMAND)) WS (AI OR CNN OR MACHINE-LEARNING OR MODEL))) AND AC:((H04N21/00 OR H04N21/234 OR H04N21/266 OR H04N21/44 OR G06F16/00 OR G06F16/783 OR G06F16/735 OR G06N20/00 OR G06N3/08 OR G06N5/00))
9	ALLCTOF(PNC:US9918134B2 OR US11715497B2 OR US11350169B2 OR US12456179B2 OR US11948271B2 OR US2025322642A1 OR US2025047939A1 OR US2016142902A1 OR US2022159327A1 OR US11170819B2 OR EP4430488A1 OR US2025039519A1 OR US2025139861A1 OR US2023143389A1 OR US11853370B2 OR US2025291845A1 OR US11223838B2)
10	TAC:((VIDEO OR MULTIMEDIA OR AUDIOVISUAL OR MOVIE OR MEDIA OR VOD OR (VIDEO W3 DEMAND) OR (LIVE W2 STREAM*) OR MULTI_MEDIA) AND ((((LINK* OR TIME OR SEQUENCE) W3 (INFORMATION OR DATA)) OR METADATA OR TIME_STAMP OR (TIME W3 (STAMP OR LOCATION OR POSITION))) W5 (VIDEO OR MULTIMEDIA OR AUDIOVISUAL OR MOVIE OR MEDIA OR VOD OR (VIDEO W3 DEMAND) OR (LIVE W2 STREAM*) OR MULTI_MEDIA)) AND (QUERY OR PROMPT OR (INPUT W3 INSTRUCTION) OR (USER W2 INPUT)) AND (AI OR "ARTIFICIAL INTELLIGENCE" OR "MACHINE LEARNING") AND ((SEARCH OR FIND OR DETERMINE OR LOCAT*) W3 (SCENE OR (VIDEO W2 (ELEMENT OR SEGEMENT OR PART OR CLIP OR SECTION OR SHOT OR "TIME RANGE")))))

	AND PBD:[2000-01-01 TO 2026-03-24]
11	TAC:((VIDEO OR MULTIMEDIA OR AUDIOVISUAL OR MOVIE OR MEDIA OR VOD OR (VIDEO W3 DEMAND) OR (LIVE W2 STREAM*) OR MULTI_MEDIA) AND ((METADATA OR TIME_STAMP OR (TIM*3 W3 (STAMP OR LOCATION OR POSITION OR INFORMATION OR DATA))) W5 (VIDEO OR MULTIMEDIA OR AUDIOVISUAL OR MOVIE OR MEDIA OR VOD OR (VIDEO W3 DEMAND) OR (LIVE W2 STREAM*) OR MULTI_MEDIA)) AND (QUERY OR PROMPT OR (INPUT W3 INSTRUCTION) OR (USER W2 INPUT)) AND (AI OR "ARTIFICIAL INTELLIGENCE" OR "MACHINE LEARNING") AND (SCENE OR (VIDEO W2 (ELEMENT OR SEGMENT OR PART OR CLIP OR SECTION OR SHOT OR "TIME RANGE")))) AND PBD:[2000-01-01 TO 2026-03-24]
12	TAC:((VIDEO OR MULTIMEDIA OR AUDIOVISUAL OR MOVIE OR MEDIA OR VOD OR (VIDEO W3 DEMAND) OR (LIVE W2 STREAM*) OR MULTI_MEDIA) AND (METADATA OR TIME_STAMP OR (TIM*3 W3 (STAMP OR LOCATION OR POSITION OR INFORMATION OR DATA))) AND (AI OR "ARTIFICIAL INTELLIGENCE" OR "MACHINE LEARNING") AND ((QUERY OR PROMPT OR (INPUT W3 INSTRUCTION) OR (USER W2 INPUT) OR REQUEST* OR INQUIRY) WS (SCENE OR (VIDEO W2 (ELEMENT OR SEGMENT OR PART OR CLIP OR SECTION OR SHOT OR "TIME RANGE"))))) AND PBD:[2000-01-01 TO 2026-03-24] AND AC:(G06F16/* OR G06V20/49)
13	AC:(G06F16/732 OR G06F16/738) AND AC:(G06F16/783) AND TAC:((SCENE OR (VIDEO W2 (ELEMENT OR SEGMENT OR PART OR CLIP OR SECTION))) AND (QUERY OR PROMPT OR (INPUT W3 INSTRUCTION) OR (USER W2 INPUT) OR REQUEST* OR INQUIRY)) AND LSC:(ACTIVE - GRANTED)
14	TAC:((VIDEO OR MULTIMEDIA OR AUDIOVISUAL OR MOVIE OR MEDIA OR VOD OR (VIDEO W3 DEMAND) OR (LIVE W2 STREAM*) OR MULTI_MEDIA) AND (METADATA OR TIME_STAMP OR (TIM*3 W3 (STAMP OR LOCATION OR POSITION OR INFORMATION OR DATA))) AND ((QUERY OR PROMPT OR (INPUT W3 INSTRUCTION) OR (USER W2 INPUT) OR REQUEST* OR INQUIRY) W6 (SCENE OR (VIDEO W2 (ELEMENT OR SEGMENT OR PART OR CLIP OR SECTION OR SHOT OR "TIME RANGE" OR PORTION))))) AND PBD:[2000-01-01 TO 2026-03-24]
15	AASN:(((IDOMOO W2 LTD) OR (SONY W2 GROUP W2 CORP) OR (INTEL) OR (NETFLIX) OR APPLE OR (IMAGEPROOF) OR (IBM) OR (GOOGLE) OR (HCL W3 TECHNOLOGY) OR (GENETEC) OR (VIDAFAIR) OR (VERIZON)

	OR (AMAZON) OR (DIVX) OR (NAGRAVISION W3 SARL) OR (GENETEC) OR (ION W3 VIDEO) OR (MICROSOFT) OR ((WARNER W3 BROS) W3 ENTERTAINMENT) OR (SONY) OR (ERICSSON) OR (III W3 HOLDINGS) OR (SLING W3 MEDIA) OR (HANGZHOU W3 TAOPIAOPIAO W3 FILM))) AND TAC:((VIDEO W3 (MEDIA OR VIRTUALISATION OR RENEDER* OR ASSEMBL* OR STRUCTURE OR METADATA OR REPRESENTATION))) AND TAC:(((OPERAT* OR ANALY* OR CHECK*) WS (AI OR CNN OR MACHINE_LEARNING OR NEURAL)))
16	INV:(((CANDELORE W2 BRANT) OR (GOLDBERG W2 ADAM) OR (BLANCHARD W2 ROBERT) OR (CHRISTIAN W3 KAISER) OR (JEAN W3 MARIE W3 WHITE) OR (YUNG W3 HSIAO W3 LAI) OR (YANN W3 BIEBER) OR (YONGJUN W3 WU) OR (BALACHANDAR W3 SIVAKUMAR) OR (SHYAM W3 SADHWANI) OR (PADMANABHA W3 RAO) OR (RAN W3 LIU) OR (CHUANJI W3 TANG) OR (ZUOLONG W3 WANG) OR (YE W3 SUN) OR (KYONG W3 PARK) OR (MICHAEL W3 CHEN) OR (PIERRE W3 RACZ) OR (FREDERIC W3 RIOUX) OR (TING W3 TSENG) OR (PAWEL W3 JURCZYK) OR (SEAN W3 WATSON) OR (MATTHEW W3 DALCIN) OR (AARON W3 MARKING) OR (JEFFREY W3 LOTSPIECH) OR (KENNETH W3 GOELLER) OR (PATTERSON W2 GENEVIEV))) AND TAC:((VIDEO W3 (MEDIA OR VIRTUALISATION OR RENEDER* OR ASSEMBL* OR STRUCTURE OR METADATA OR REPRESENTATION))) AND AC:(((H04N21/00 OR H04N21/234 OR H04N21/266)))

AUSPAT:

Sr. No.	Key Strategies
1	CLAIMS = (VIDEO OR "VIDEO VIRTUALIZATION" OR "VIDEO RETRIEVAL" OR "VIDEO RENDER" OR "VIDEO ASSEMBLY") AND (SEGMENT* OR CLIP* OR FRAME*) AND ("AI" OR "ARTIFICIAL INTELLIGENCE") AND ("CONTEXT AWARE" OR "USER SPECIFIC" OR "SCENE BASED") AND ("USER INTENT" OR QUERY OR "NATURAL LANGUAGE") AND FILING DATE FROM 1/1/2006 TO 3/20/2026
2	CLAIMS= ("VIDEO PLAYBACK" OR MEDIA OR "MULTIMEDIA ABSTRACTION" OR "VIDEO ASSEMBLY" OR "VIDEO STREAM" OR VIDEO) AND (ARTIFICIAL INTELLIGENCE OR MACHINE LEARNING OR "AI" OR "LLM" OR "NATURAL LANGUAGE") AND (SEGMENT* OR CLIP* OR FRAME* OR PART*) AND (QUERY OR "USER INTENT" OR "USER PREFERENCE") AND (INTERPRET OR UNDERSTAND) AND FILING DATE FROM 1/1/2006 TO 3/20/2026 AND DESCRIPTION= ((RENDER OR "FILE DUPLICATION" OR TRANSCODE) AND (REASSEMBLE OR RESEQUENCE OR REARRANGE))
3	CLAIMS = (SYSTEM OR METHOD) AND (VIDEO OR "VIDEO VIRTUALIZATION" OR "VIDEO ABSTRACTION") AND (SEGMENT* OR CLIP* OR FRAME*) AND (INTERPRET OR UNDERSTAND) AND (QUERY OR "NATURAL LANGUAGE" OR PROMPT) AND (RETRIEVE OR REASSEMBLE OR RESEQUENCE OR REARRANGE) AND ("AI" OR "ARTIFICIAL INTELLIGENCE" OR "ML" OR "MACHINE LEARNING") AND (TRANCOD* OR "FILE DUPLICATION" OR RENDER) AND FILING DATE FROM 1/1/2006 TO 3/20/2026
4	CLAIMS= (("VIDEO /3/ ASSEMBLY") OR ("VIDEO /3/ VIRTUALIZATION") OR ("VIDEO /3/ RETRIEVE")) AND (SEGMENT* OR CLIP* OR FRAME* OR PORTION* OR PART* OR SAMPLE*) AND ("AI" OR "ARTIFICIAL INTELLIGENCE" OR "MACHINE LEARNING" OR "ML" OR "LLM") AND ("USER /3/ INTENT" OR QUERY OR PROMPT OR COMMAND) AND FILING DATE FROM 1/1/2006 TO 3/20/2026
5	CLAIMS = (MEDIA OR MULTIMEDIA OR VIDEO) AND (SEGMENT* OR CLIP* OR FRAME* OR PORTION* OR PART*) AND ("USER /3/ INTENT" OR "USER /3/ INSTRUCTION" OR QUERY OR PROMPT) AND (METADATA OR

	"TIME STAMPS" OR TIMESTAMPS) AND FILING DATE FROM 1/1/2006 TO 3/25/2026
6	CLAIMS = ((MEDIA OR MULTIMEDIA OR VIDEO OR AUDIOVISUAL OR VISUAL) AND (SCENE OR SEGMENT OR PIECE) AND ("USER /3/ INTENT" OR "USER /3/ INSTRUCTION" OR QUERY OR PROMPT OR INQUIRY OR REQUEST) AND (METADATA OR "DESCRIPTIVE /3/ DATA" OR DESCRIPTORS OR "TIME STAMPS" OR TIMESTAMPS OR "TIMING INFORMATION")) AND FILING DATE FROM 1/1/2006 TO 3/25/2026

GOOGLE PATENTS:

Sr. No.	Key Strategies
1	CL=("VIDEO" OR "MULTIMEDIA" OR "FILM") CL=("AI" OR "ARTIFICIAL INTELLIGENCE" OR "MACHINE LEARNING" OR "INTELLIGENT AGENT") CL=("ADDRESSABLE" OR "COMPOSABLE" OR "VIRTUALIZATION" OR "STRUCTURAL") COUNTRY:US,EP,AU STATUS:GRANT
2	CL=("VIDEO COMPOSITION" OR "MEDIA ASSEMBLY") AND (AI OR "MACHINE LEARNING") AND (DYNAMIC OR REAL-TIME) AND (NO RENDERING)
3	CL=(VIDEO ADJ3 (VIRTUALIZATION OR REPRESENTATION OR PLAYBACK OR RENDER*)) CL=(VIDEO ADJ/3 (SEGMENT* OR FRAME* OR SMAPLES OR SEQUENCE OR PORTION*)) CL=(OPERAT* ADJ/4 (AI OR "MACHINE LEARNING" OR "NEURAL NETWORK" OR CNN)) (G06V20/48)
4	CL=((VIDEO OR MEDIA) ADJ/4 (ASSEMBL* OR VIRTUALIZATION OR REPRESENTATION OR PLAYBACK OR RENDER*)) CL=((VIDEO OR MEDIA) ADJ/4 (SEGMENT OR FRAME* OR PART*)) CL=(VIDEO ADJ/4 (MACHINE_LEARNING OR NEURAL OR AI OR CNN)) CL=(USER ADJ/4 (PREFERENCE OR COMMAND OR INPUT OR INTENT)) ASSIGNEE:(NETFLIX INC.) ASSIGNEE:(INTEL CORP) ASSIGNEE:(AMAZON TECH INC) COUNTRY:US,EP,AU BEFORE:PRIORITY:20260323 AFTER:PRIORITY:20060101
5	CL=((VIDEO OR MEDIA) ADJ/4 (ASSEMBL* OR VIRTUALIZATION OR REPRESENTATION OR PLAYBACK OR RENDER*)) CL=((VIDEO OR MEDIA) ADJ/4 (SEGMENT OR FRAME* OR PART*)) CL=(VIDEO ADJ/4 (MACHINE_LEARNING OR NEURAL OR AI OR CNN)) CL=(USER ADJ/4 (PREFERENCE OR COMMAND OR INPUT OR INTENT)) CL=((NO OR AVOID OR STOP) ADJ/4 (RENDER* OR DUPLICAT* OR STOR*)) ASSIGNEE:(NETFLIX INC.) ASSIGNEE:(INTEL CORP) ASSIGNEE:(AMAZON TECH INC) COUNTRY:US,EP,AU BEFORE:PRIORITY:20260323 AFTER:PRIORITY:20060101
6	CL= (VIDEO ADJ3 SEGMENT*)AND (METADATA OR SEMANTIC OR EMBEDDING) AND ((QUERY OR "NATURAL

	LANGUAGE") NEAR3 (INTERPRET* OR PROCESS*))AND (INTENT NEAR3 (DETECT* OR INFER*)) AND ("AI" NEAR3 (MODEL OR LEARNING)) AND (SELECT* NEAR3 SEGMENT*) AND ((AVOID OR PREVENT) ADJ3 (RERENDER* OR REEDIT* OR TRANSCOD*))
7	CL= ((VIDEO OR MEDIA) ADJ3 (VIRTUALIZATION OR PROCESSING OR ASSEMBLY)) AND ((SEGMENT* OR CLIP* OR PORTION OR PART*) NEAR3 (REASSEMBLE OR RESTRUCTURE OR RESEQUENCE)) AND ("AI" OR "ARTIFICIAL INTELLIGENCE" OR "MACHINE LEARNING" OR "LLM") AND ("USER INTENT" OR "USER CHOICE" OR "USER PREFERENCE") AND ((AVOID OR PREVENT OR NO) ADJ3 (REENCODING OR "FILE DUPLICATION" OR "RE EDITING"))
8	CL= (VIDEO ADJ3 SEGMENT*)AND (METADATA OR SEMANTIC OR EMBEDDING) AND ((QUERY OR "NATURAL LANGUAGE") NEAR3 (INTERPRET* OR PROCESS*))AND (INTENT NEAR3 (DETECT* OR INFER*)) AND ("AI" NEAR3 (MODEL OR LEARNING)) AND (SELECT* NEAR3 SEGMENT*) AND ((AVOID OR PREVENT) ADJ3 (RERENDER* OR REEDIT* OR TRANSCOD*))

USPTO:

Sr. No.	Key Strategies
1	((VIDEO OR MULTIMEDIA) ADJ3 (VIRTUALIZATION OR ASSEMBLY OR RETRIEVAL)) AND (SEGMENT OR CLIP OR PORTION) AND ((OPERATE* OR ANALY*) NEAR3 (AI OR "NEURAL NETWORK" OR MACHINE LEARNING OR ALGO* OR MODEL*)) AND ((USER OR HUMAN*) NEAR4 (INTENT* OR PREFERENCE* OR INPUT OR COMMAND)) AND (H04N21/43072 OR G06N3/0464 OR G06V20/48).CPC.
2	((VIDEO NEAR3 SEGMENT*) NEAR5 (ASSEMBL* OR COMPOS* OR GENERAT*)) AND ((METADATA OR SEMANTIC) NEAR5 (INSTRUCTION* OR REFERENCE*)) AND ((INTENT OR QUERY) NEAR5 ("AI" OR "MACHINE LEARNING"))

3	((VIDEO OR MULTIMEDIA) ADJ3 (VIRTUALIZATION OR ASSEMBLY OR RETRIEVAL)) AND (SEGMENT OR CLIP OR PORTION)AND (("NATURAL LANGUAGE" OR QUERY) NEAR3 (PROCESS* OR UNDERSTAND* OR INTERPRET*)) AND (VIDEO ADJ3 SEGMENT*) AND ((RETRIEV* OR SELECT* OR LOCALIZ*) NEAR5 (SEMANTIC OR EMBEDDING OR FEATURE)) AND ((EXTRACT OR FETCH OR RETRIEVE) ADJ3 (METADATA OR "MOOV" OR "MEDIA DATA" OR "MDAT"))
4	((G06T2207/10016 AND G06T7/10).CPC.) AND ("AI" OR "ML" OR "MACHINE LEARNING") AND ((USER) ADJ3 (INTENT* OR PREFERENCE OR NEED OR CHOICE))
5	((VIDEO) ADJ3 (SEGMENT OR CLIP OR PORTION)) AND ("AI" OR "MACHINE LEARNING") AND ((EXTRACT OR RETRIEVE) NEAR3 (METADATA OR "MEDIA DATA")) AND ((USER NEAR3 (INTENT* OR PREFER*)) OR "NATURAL LANGUAGE" OR "USER QUERY") AND (TRANSCOD* OR RERENDER*)
6	((VIDEO OR MULTIMEDIA) ADJ3 (VIRTUALIZATION OR ASSEMBLY OR RETRIEVAL)) AND (SEGMENT OR CLIP OR PORTION) AND ((OPERATE* OR ANALY*) NEAR3 (AI OR "NEURAL NETWORK" OR MACHINE LEARNING OR ALGO* OR MODEL*)) AND (348/042 OR 348/E13.001 OR 715/719).CCLS.
7	(382/199 OR 725/087 OR 725/109348/042 OR 348/E13.001 OR 715/719).CCLS. AND ((VIDEO OR MULTIMEDIA) ADJ3 (VIRTUALIZATION OR ASSEMBLY OR RETRIEVAL)) AND (SEGMENT OR CLIP OR PORTION)AND ((OPERATE* OR ANALY*) NEAR3 (AI OR "NEURAL NETWORK" OR MACHINE LEARNING OR ALGO* OR MODEL*))
8	"((VIDEO OR MULTIMEDIA) ADJ3 (SEGMENT OR CLIP OR PORTION)) AND ((METADATA OR SEMANTIC OR FEATURE) NEAR5 (RETRIEV* OR SELECT* OR SEARCH)) AND ((INTENT OR QUERY OR "NATURAL LANGUAGE") NEAR5 (""AI"" OR "MACHINE LEARNING")) AND H04N21/8456.CPC."
9	("VIDEO VIRTUALIZATION" OR "VIDEO VIRTUALISATION" OR "MEDIA ABSTRACTION" OR "VIDEO DECOMPOSITION" OR "CONTENT DECOMPOSITION" OR "MEDIA SEGMENTATION" OR "STRUCTURAL VIDEO REPRESENTATION") AND (G06N3/0464 OR G06N20/10 OR G06V20/49).CPC.

10	(G06V20/48 OR G06V20/49 OR H04N21/845 OR G06V20/46).CPC. AND ((VIDEO OR MEDIA OR CONTENT) NEAR3 (VIRTUALISATION OR RENEDER* OR ASSEMBL* OR STRUCTURE OR METADATA OR REPRESENTATION)) AND ((ANALY* OR CHECK* OR OPERAT* OR VALIDAT*) NEAR3 VIDEO)
11	("VIDEO" AND ("VIRTUALIZATION" OR "VIRTUALISATION" OR "ABSTRACTION LAYER") AND ("STRUCTURE" OR "SCHEMA" OR "REPRESENTATION") AND ("SEPARATE" OR "DECOUPLE" OR "ISOLATE") AND ("PIXEL DATA" OR "MED$_{I}$A S$_{A}$MPL$_{E}$S" OR "ENCODED S$_{T}$RE$_{AM}$")) AND (("CON$_{TEN}$T-A^{DDR}$_{E}$SSABL$_{E}$" OR "INDEXED $_{M}$E$_{D}$IA") AND ("VIDEO" OR "MULTIMEDIA") AND ("GRANULAR" OR "FRAME-LEVEL" OR "OBJECT-LEVEL"))
12	("ARTIFICIAL INTELLIGENCE" OR "MACHINE LEARNING" OR "NEURAL NETWORK") AND ("VIDEO GENERATION" OR "MEDIA COMPOSITION") AND (("SEMANTIC RETRIEVAL" OR "CONTENT-BASED RETRIEVAL") AND ("VIDEO" OR "MEDIA") AND ("COMPOSITION" OR "ASSEMBLY"))

ESPACENET:

Sr. No.	Key Strategies
1	(CLAIMS=("VIDEO" PROX/DISTANCE<3 "RENDER*") OR CLAIMS=("VIDEO" PROX/DISTANCE<3 "PLAYBACK") OR CLAIMS=("VIDEO" PROX/DISTANCE<3 "REPRESENTATION*")) AND (CLAIMS ALL "SEGMENT*" OR CLAIMS ALL "PORTION*" OR CLAIMS ALL "FRAMES") AND (CLAIMS ANY "AI-SYSTEMS" OR CLAIMS ANY "CNN" OR CLAIMS=("MACHINE" PROX/DISTANCE<3 "LEARNING")) AND CL ANY "H04N21/845"
2	(CLAIMS=("VIDEO" PROX/DISTANCE<3 "RENDER*") OR CLAIMS=("VIDEO" PROX/DISTANCE<3 "PLAYBACK") OR CLAIMS=("VIDEO" PROX/DISTANCE<3 "REPRESENTATION*")) AND (CLAIMS ALL "SEGMENT*" OR CLAIMS ALL "PORTION*" OR CLAIMS ALL "FRAMES") AND (CLAIMS ANY "AI-SYSTEMS" OR CLAIMS ANY "CNN" OR CLAIMS=("MACHINE" PROX/DISTANCE<3 "LEARNING")) AND (CL ANY "H04N21/845" OR CL ANY "G06V20/46" OR CL ANY "G06N20/10")
3	(CLAIMS=("VIDEO" PROX/DISTANCE<3 "VIRTUALISATION") OR CLAIMS=("VIDEO" PROX/DISTANCE<3 "MEDIA") OR CLAIMS=("VIDEO" PROX/DISTANCE<3 "RENDER*") OR CLAIMS=("VIDEO" PROX/DISTANCE<3 "ASSEMBLY")) AND (CTXT ALL "SEGMENT*" OR CTXT ALL "PORTION" OR CTXT ALL "FRAME*") AND (CTXT ANY "AI" OR CTXT=("NEURAL" PROX/DISTANCE<3 "NETWORK") OR CTXT=("MACHINE" PROX/DISTANCE<3 "LEARNING") OR CTXT=("ARTIFICIAL" PROX/DISTANCE<3 "INTELLEGENCE")) AND (CTXT=("OPERAT*" PROX/DISTANCE<3 "VIDEO") OR CTXT=("VIDEO" PROX/DISTANCE<3 "ANALY*"))
4	(CLAIMS=("VIDEO" PROX/DISTANCE<3 "VIRTUALISATION") OR CLAIMS=("VIDEO" PROX/DISTANCE<3 "MEDIA") OR CLAIMS=("VIDEO" PROX/DISTANCE<3 "RENDER*") OR CLAIMS=("VIDEO" PROX/DISTANCE<3 "ASSEMBLY")) AND (CTXT ALL "SEGMENT*" OR CTXT ALL "PORTION" OR CTXT ALL "FRAME*") AND (CTXT ANY "AI" OR CTXT=("NEURAL" PROX/DISTANCE<3 "NETWORK") OR CTXT=("MACHINE" PROX/DISTANCE<3 "LEARNING") OR CTXT=("ARTIFICIAL" PROX/DISTANCE<3 "INTELLEGENCE")) AND (CTXT=("OPERAT*" PROX/DISTANCE<3 "VIDEO") OR CTXT=("VIDEO" PROX/DISTANCE<3 "ANALY*")) AND (CTXT=("USER" PROX/DISTANCE<3 "INETENT") OR CTXT ALL "PREFRENCE" OR CTXT ALL "INPUT" OR CTXT ALL "COMMAND")

5	(CTXT=("VIDEO" PROX/DISTANCE<3 "VIRTUALIZATION") OR CTXT=("VIDEO" PROX/DISTANCE<3 "RETRIEVAL") OR CTXT=("VIDEO" PROX/DISTANCE<3 "PLAYBACK")) AND (CTXT ANY "SEGMENT" OR CTXT ANY "CLIP") AND (CTXT=("RETRIEVE" PROX/UNIT=SENTENCE "SEGMENT") OR CTXT=("FETCH" PROX/UNIT=SENTENCE "CLIP")) AND (NFTXT = "AI" OR NFTXT = "MACHINE LEARNING")
6	(CTXT=("VIDEO" PROX/DISTANCE<3 "VIRTUALIZATION") OR CTXT=("VIDEO" PROX/DISTANCE<3 "RETRIEVAL") OR CTXT=("VIDEO" PROX/DISTANCE<3 "PLAYBACK")) AND (CTXT ANY "SEGMENT" OR CTXT ANY "CLIP") AND (CTXT=("RETRIEVE" PROX/UNIT=SENTENCE "SEGMENT") OR CTXT=("FETCH" PROX/UNIT=SENTENCE "CLIP")) AND (NFTXT = "AI" OR NFTXT = "MACHINE LEARNING")
7	(CTXT=("VIDEO " PROX/DISTANCE<3 "VIRTUALIZATION") OR CTXT=("VIDEO" PROX/DISTANCE<3 "ASSEMBLY") OR CTXT=("VIDEO" PROX/DISTANCE<3 "RENDER*")) AND (CTXT=("VIDEO" PROX/DISTANCE<3 "SEGMENTS") OR CTXT=("VIDEO" PROX/DISTANCE<3 "CHUNKS") OR CTXT=("VIDEO" PROX/DISTANCE<3 "FRAMES") OR CTXT=("VIDEO" PROX/DISTANCE<3 "SAMPLES")) AND (NFTXT = "AI" OR NFTXT=("ARTIFICIAL" PROX/DISTANCE<3 "INTELLIGENCE") OR NFTXT=("MACHINE" PROX/DISTANCE<3 "LEARNING") OR NFTXT=("AI" PROX/DISTANCE<3 "MODEL")) AND (CTXT ALL "QUERY" OR CTXT=("USER" PROX/DISTANCE<3 "PROMPT") OR CTXT=("USER " PROX/DISTANCE<3 "COMMAND") OR CTXT=("USER " PROX/DISTANCE<3 "INPUT"))
8	(CTXT=("VIDEO " PROX/DISTANCE<3 "VIRTUALIZATION") OR CTXT=("VIDEO" PROX/DISTANCE<3 "ASSEMBLY") OR CTXT=("VIDEO" PROX/DISTANCE<3 "RENDER*")) AND (CTXT=("VIDEO" PROX/DISTANCE<3 "SEGMENTS") OR CTXT=("VIDEO" PROX/DISTANCE<3 "CHUNKS") OR CTXT=("VIDEO" PROX/DISTANCE<3 "FRAMES") OR CTXT=("VIDEO" PROX/DISTANCE<3 "SAMPLES")) AND (NFTXT = "AI" OR NFTXT=("ARTIFICIAL" PROX/DISTANCE<3 "INTELLIGENCE") OR NFTXT=("MACHINE" PROX/DISTANCE<3 "LEARNING") OR NFTXT=("AI" PROX/DISTANCE<3 "MODEL")) AND (CTXT ALL "QUERY" OR CTXT=("USER" PROX/DISTANCE<3 "PROMPT") OR CTXT=("USER " PROX/DISTANCE<3 "COMMAND") OR CTXT=("USER " PROX/DISTANCE<3 "INPUT")) AND (NFTXT ANY "INTERPRET" OR NFTXT ANY "ANALYSE" OR NFTXT ANY "UNDERSTAND") AND (NFTXT ANY "REASSEMBLE" OR NFTXT ANY "RESEQUENCE" OR NFTXT ANY "RETRIEVE")

9	(CTXT=("VIRTUALIZATION" PROX/DISTANCE<3 "SYSTEM") OR CTXT=("RETRIEVAL" PROX/DISTANCE<3 "SYSTEM")) AND (CTXT=("VIDEO" PROX/DISTANCE<3 "SEGMENT") OR CTXT=("VIDEO" PROX/DISTANCE<3 "SAMPLE") OR CTXT=("VIDEO" PROX/DISTANCE<3 "FRAMES")) AND (CTXT = "AI" OR CTXT=("ARTIFICIAL " PROX/DISTANCE<3 "INTELLIGENCE") OR CTXT=("MACHINE" PROX/DISTANCE<3 "LEARNING")) AND (CTXT=("USER" PROX/DISTANCE<3 "INTENT") OR CTXT ANY "QUERY" OR CTXT=("USER " PROX/DISTANCE<3 "PROMPT") OR CTXT=("NATURAL " PROX/DISTANCE<3 "LANGUAGE"))
10	(CTXT=("VIDEO" PROX/DISTANCE<3 "SCENE") OR CTXT=("VIDEO" PROX/DISTANCE<3 "SEGMENT") OR CTXT=("VIDEO" PROX/DISTANCE<3 "SECTION")) AND CTXT=("SEARCH" PROX/DISTANCE<3 "SCENE") AND (CTXT ALL "METADATA" OR CTXT ALL "TIME")
11	(CTXT=("VIDEO" PROX/DISTANCE<3 "SCENE") OR CTXT=("VIDEO" PROX/DISTANCE<3 "SEGMENT") OR CTXT=("VIDEO" PROX/DISTANCE<3 "SECTION") OR CTXT=("VIDEO" PROX/DISTANCE<3 "CLIP") OR CTXT=("VIDEO" PROX/DISTANCE<3 "SHOT")) AND CTXT ALL "METADATA" AND (CTXT ANY "AI" OR CTXT=("ARTIFICIAL " PROX/DISTANCE<3 "INTELLIGENCE") OR CTXT=("MACHINE" PROX/DISTANCE<3 "LEARNING"))
12	(CTXT=("VIDEO" PROX/DISTANCE<3 "SCENE") OR CTXT=("VIDEO" PROX/DISTANCE<3 "SEGMENT") OR CTXT=("VIDEO" PROX/DISTANCE<3 "SECTION") OR CTXT=("VIDEO" PROX/DISTANCE<3 "CLIP") OR CTXT=("VIDEO" PROX/DISTANCE<3 "SHOT")) AND CTXT ALL "METADATA" AND (CTXT ANY "AI" OR CTXT=("ARTIFICIAL " PROX/DISTANCE<3 "INTELLIGENCE") OR CTXT=("MACHINE" PROX/DISTANCE<3 "LEARNING")) AND (CL ALL "G06F16/732" OR CL ALL "G06F16/738" OR CL ALL "G06F16/783" OR CL ALL "G06V20/49")

PATENTSCOPE:

Sr. No.	Key Strategies

1	EN_ALLTXT:(((VIDEO) NEAR3 (VIRTUALIZATION OR RETRIEVAL)) AND (SEGMENT OR CLIP OR PORTION OR SAMPLE OR FRAME) AND ((("AI" OR "MACHINE LEARNING" OR "ML") NEAR3 (UNDERSTAND OR INTERPRET)) NEAR5 ("NATURAL LANGUAGE" OR QUERY OR INTENT)))
2	EN_CL:("VIDEO VIRTUALIZATION"~5) AND (SEGMENT OR FRAME OR SAMPLE) AND ("AI MODEL" OR "MACHINE LEARNING" OR "ML" OR "NEURAL NETWORK") AND ((INTERPRET OR UNDERSTAND) NEAR3 ("USER PROMPT" OR "USER INTENT" OR "USER PREFERENCE"))
3	EN_ALLTXT:(((VIDEO) NEAR3 (VIRTUALIZATION OR RETRIEVAL)) AND (SEGMENT OR CLIP OR PORTION OR SAMPLE OR FRAME) AND ((("AI" OR "MACHINE LEARNING" OR "ML") NEAR3 (UNDERSTAND OR INTERPRET)) NEAR5 ("NATURAL LANGUAGE" OR QUERY OR INTENT)))
4	EN_CL:("VIDEO VIRTUALIZATION"~5) AND (SEGMENT OR FRAME OR SAMPLE) AND ("AI MODEL" OR "MACHINE LEARNING" OR "ML" OR "NEURAL NETWORK") AND ((INTERPRET OR UNDERSTAND) NEAR3 ("USER PROMPT" OR "USER INTENT" OR "USER PREFERENCE"))
5	EN_CL: (VIDEO NEAR/3 (VIRTUALIZATION OR RETRIEVAL OR PLAYBACK)) AND (SEGMENT* OR FRAME* OR CLIP) AND (WITHOUT NEAR/3 (RE-ENCODING OR DUPLICATION OR MODIFICATION)) AND ((INTENT OR QUERY OR "NATURAL LANGUAGE") NEAR/3 (DETECT OR INTERPRET OR UNDERSTAND)) AND ("AI" OR "MACHINE LEARNING" OR "ARTIFICIAL INTELLIGENCE" OR "LLM") AND (REASSEMBLE OR REARRANGE OR RESEQUENCE)
6	EN_ALLTXT: (("VIDEO VIRTUALIZATION") OR ("VIDEO RETRIEVAL") OR ("VIDEO PLAYBACK")) AND (VIDEO NEAR/3 (SEGMENT* OR FRAME* OR CLIP)) AND (RECONSTRUCT OR REBUILD OR REASSEMBLE) AND (WITHOUT NEAR/3 (RE-ENCODING OR DUPLICATION OR MODIFICATION)) AND ((INTENT OR QUERY OR "NATURAL LANGUAGE") NEAR/3 (DETECT OR INTERPRET)) AND ("AI" OR "MACHINE LEARNING") AND (SEGMENT OR CLIP) AND (ASSEMBLE OR COMPOSE) AND CLASSIF: G06T2207/10016

7. CLASSES

a. IPC: G06F16/248, G06F16/732; G06F16/738; G06F16/783; G06F16/22, G06F16/242, G06F16/383, H04N21/25, G06V20/40, H04N21/234, H04N21/8549, H04N19/176; H04N19/119, H04N21/478; H04N21/432; H04N21/44; H04N21/472, H04N21/845

b. CPC: H04N21/43072, G06N3/0464, G06N20/10, G06T2207/20084, G06V20/48, G06V20/49, H04N21/845, G06V20/46, H04N21/4532, G06V10/764, H04N19/147, G06N3/006

b. USPC: 348/042, 348/E13.001, 715/719, 600/300, 725/093, 725/062, 382/199, 725/087, 725/109

8. CONCLUSION

A total of 8 patents/patent applications were identified and reviewed. The following are the main findings:

2 patents and 1 patent application were found to be an immediate infringement risk disclosing "The prior arts collectively describe various methods related to content segmentation, scene navigation, and retrieval within digital video systems. Prior art discloses a method for generating a video segment based on a user query, enabling dynamic creation of content portions bounded by metadata matches. Prior arts details a system for processing spoken scene playback requests within a media player, where scene metadata containing semantic and timestamp information is used to navigate directly to specific scenes upon user request. Further a content retrieval approach that leverages machine learning to interpret user queries, identify relevant scenes within a content database, and display these scenes for user selection or viewing. Overall, these prior arts encompass techniques for content segmentation, precise scene navigation, and targeted scene retrieval driven by metadata analysis, user input, or machine learning".

2 patent application and 3 patents were found that are related to "systems and methods for intelligent, user-driven video search, editing, and playback. These systems receive user input instructions—such as preferences, natural language queries, or voice commands—to determine intended video outputs. They process metadata, subtitles, or extracted entities from video content and compare them with query parameters using techniques like distance measures or AI-based classification. Several disclosures emphasize real-time or near-real-time processing, including filtering or editing video streams to remove undesired segments and assembling personalized outputs. Advanced implementations incorporate context-aware AI that performs frame-level analysis, refines user intent through iterative queries, and retrieves relevant segments from databases. The methods also support mapping video content to temporal snippets and enabling interactive selection via user interfaces. Overall, the prior art demonstrates programmable and queryable video systems that dynamically identify, retrieve, edit, and deliver customized video segments in response to user instructions."

9. REFERENCE CRITERIA

Reference Criteria	Categorization
Relevant References	In-force or Active patents/applications with claim(s) which completely overlaps key feature/s of the subject product.
Closely Related References	In-force or Active patents/applications with claim(s) which partially overlaps key feature/s of the subject product. (OR) Legally not In-force patents/applications with claim(s) which completely or partially overlaps primary key feature/s of the subject product.

10. DISCLAIMER

This report is work of analysis and interpretation of publicly available information on various free and paid online sources and should not be construed as a legal opinion. This report is shared with you with mutual understanding and trust that no part of this report shall be publicly distributed or used by you without explicit permission.