

(51) International Patent Classification:
H04N 7/173 (2011.01)(21) International Application Number:
PCT/IB2011/051528(22) International Filing Date:
8 April 2011 (08.04.2011)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
1052718 9 April 2010 (09.04.2010) FR(71) Applicant (for all designated States except US): **CANON KABUSHIKI KAISHA** [JP/JP]; 3-30-2 Shimomaruko, 3-chome, Ohta-ku, Tokyo (JP).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **DENOUEL, Franck** [FR/FR]; La Ville-Es-Ray, F-35190 Saint Domineuc (FR). **FABLET, Youenn** [FR/FR]; La Roulais, F-35390 La Dominelais (FR).(74) Agent: **LE TOURNEAU, Augustin**; Santarelli, 14, Avenue de la Grande-Armée, F-75017 Paris (FR).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM,

AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— of inventorship (Rule 4.17(iv))

Published:

— with international search report (Art. 21(3))

(54) Title: METHOD FOR ACCESSING A SPATIO-TEMPORAL PART OF A COMPRESSED VIDEO SEQUENCE

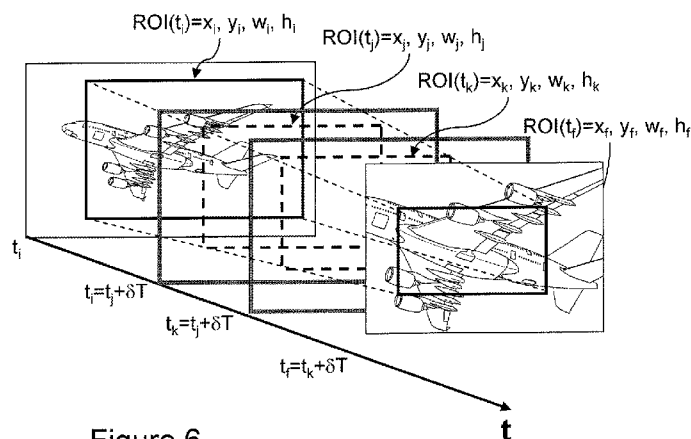


Figure 6

(57) **Abstract:** The present invention concerns a processing method for accessing a spatio-temporal part (162) of a compressed video sequence (160), as well as an associated device. The method comprises the following steps: - obtaining (S403) a request for access (REQACC) to said part by identifying a temporal section defined between initial (t_i) and final (t_f) times of the initial ($ROI(t_i)$) and final ($ROI(t_f)$) spatial regions - different from the initial region) of the sequence at the initial (t_i) and final (t_f) times; - decomposing (S406, S904, S1003, S1006, S1303, S1309) said access request into a plurality of elementary requests (REQ_n) for access to a video fragment (164_n), each elementary request identifying a fixed spatial region ($maxROI(t_n, t_{n+1})$) to extract in a temporal sub-interval of said temporal section; and - exploiting (S407, S907, S1310) at least one elementary request to access video fragments (164_n) constituting said part to access (162).

**METHOD FOR ACCESSING A SPATIO-TEMPORAL PART OF A COMPRESSED
VIDEO SEQUENCE**

5 The present invention concerns a method of accessing a spatio-temporal part of a video sequence, as well as an associated device. It applies in particular to the compressed sequences which constitute the conventional and economical format for storing video data, in particular on Internet servers.

10 Video compression algorithms, such as those standardized by the standardization organizations ITU, ISO, and SMPTE, exploit the spatial and temporal redundancies of the images in order to generate bitstreams of data of smaller size than the original video sequences. Such compressions make the transmission and/or the storage of the video sequences more efficient.

15 Most of the video compression schemes, such as the MPEG-1, MPEG-2, MPEG-4, H.261, H.263, H.264 or H.265 formats, take advantage of the so-called "temporal" redundancies between several successive images of the same sequence. Thus, most of the images are coded relative to one or more reference images by copying similar blocks and coding the prediction error. This prediction is commonly referred to as temporal or "Inter".

20 In the case of the MPEG-2 format for example, images of I type (I for Intra) are encoded without reference to other images of the sequence. Thus, when all the compressed data of such an image are available, a decoder may decode and display that image immediately. An image of I type thus constitutes a conventional point of access to the video sequence. It is to be noted that, conventionally, these images of I
25 type are presented periodically, with a period of the order of several tenths of a second to a few seconds. In the case of the H.264 format, these images are denoted "IDR" or "SI".

30 The MPEG-2 format also implements images of P type (prediction on the basis of the last I image) or B (bi-directional prediction on the basis of preceding and following images of P or I type) which are encoded by prediction relative to one or more reference images. The data compressed relative to such images (i.e. data coding the prediction errors) are not sufficient to obtain an image that can be displayed. This is because the data of the reference images which were used at the time of the prediction must be obtained. Images of P type and B type do not therefore constitute efficient
35 points of access to the video sequence.

The temporal prediction mechanism consequently proves to be extremely efficient in terms of compression, but imposes constraints on the video decoders that wish to provide proper reconstruction of the images of the same sequence, in particular by limiting the temporal random access for the compressed video sequence only to the images of I type.

Cumulatively with the exploitation of temporal redundancies, the video coders also take advantage of so-called "spatial" redundancies within the same image. For this, each image is decomposed into spatial units, blocks or macroblocks, and a block may be predicted from one or more of its spatially neighboring blocks, which is commonly referred to as spatial prediction or "Intra" prediction.

This mechanism when applied in particular in the case of the Intra images referred to previously also notably improves the compression of a video sequence. However, dependency between the blocks is introduced, and this complicates spatial extraction of part only of the sequence.

To mitigate this drawback, certain coding schemes such as H.264 provide an organization of the blocks into interdependent packets or "slices" of blocks not having spatial dependencies with blocks outside that packet. The organization into packets relies on a technique known as FMO for "*Flexible Macroblock Ordering*".

These packets are very often signaled by markers enabling a decoder to obtain access thereto without performing decoding and complete reconstruction of the image, and in particular of the blocks which precede them in the image. Each packet or slice thus constitutes a point of "access" to the video sequence or of spatial synchronization on the basis of which the decoder has no difficulty in performing decoding independently of the other packets.

Nevertheless, for these blocks, temporal dependencies may remain if the image is coded with reference to one or more other images. Thus, the accumulation of the temporal and spatial predictions means that generally the extraction of a spatio-temporal part of a video sequence, that is to say a spatial portion during a temporal section of several consecutive images of the sequence, is a complex operation.

The extraction of a spatio-temporal part from a video sequence is desirable in numerous applications and these days is the subject of extensive developments.

The W3C ("World Wide Web Consortium", an organization producing standards for the Web) is working on the development of a mechanism for addressing temporal segments or spatial regions in resources that are available on the Web such

as video sequences, by using in particular URIs ("Uniform Resource Identifiers") making it possible to identify, via a string, a physical or abstract resource.

This mechanism, independently of the format of representation of the resource, is termed "Media Fragments".

5 The RFC ("Request For Comments") memorandum number 3986 defines a syntax for the URIs, and integrates in particular the concepts of "fragment" and of "queries" or requests. In this context, a fragment is in particular a part, a subset, a view or a representation of a primary resource.

10 The "Media Fragments" addressing enabling the access to sub-parts of the audio or video stream or within images, by addition of parameters to the request, following the URI address makes it possible for example to address:

- temporal segments (or "temporal fragments") defined by initial and terminal times: *t=00:01:20,00:02:00* identifying the segment from 1min20s to 2min00s; and/or
- 15 – spatial regions (or "spatial fragments") defined by a generally rectangular viewing region: *xywh= 10, 10, 256, 256* specifying the upper left corner (10, 10), the width (256) and the height (256) of the rectangle; and/or
- substreams (or "track fragments"), for example a particular audio track associated with a video *track='audio_fr '*; and/or
- 20 – passages (or "named fragments") pre-defined via an identifier, a scene of a film for example: *id= 'the_scene_of_the_kiss'*.

In addition to the syntax of the fragments/requests for the addressing thereof, the same working group is in course of producing a client-server communication protocol based on the HTTP protocol ("Hyper Text Transfer Protocol")
25 used on the Web.

In particular, the protocol defines the HTTP requests sent by a client wishing to obtain fragments as well as the responses sent by the server responsible for the extraction and for the sending of those fragments. Each HTTP request or associated HTTP response is composed of header information and data information.
30 The header information may be considered as description/signaling information (in particular as to the type of the data exchanged and as to the identity of the data sent back – region finally sent back) whereas the data information correspond to the spatial and/or temporal fragment of the resource requested by the client.

When the requested fragment can be converted into "byte ranges" either
35 because the client has already received a description of the resource before sending

his request, or because the server performs an analysis of the resource before sending it, the exchange of fragments is similar to a conventional exchange of data via HTTP, which makes it possible to exploit cache mechanisms and thereby be fully integrated into a Web architecture.

5 If on the other hand the fragment cannot be converted into one or more byte ranges belonging to the original resource, transcoding is then necessary at the server, and the new resource so created is sent to the client like any other resource.

 This addressing mechanism and the associated communication protocol are advantageously flexible in that they are independent from the video format used,
10 from the encoding options of the video sequences and from the capacities available on the servers processing such requests.

 Furthermore, the setting up of this addressing mechanism and of the associated communication protocol will eventually make it possible to significantly reduce the quantity of information exchanged between a client requesting parts of a
15 video and a server storing that video and supplying those requested parts.

 This is because, if a client only wishes to view a temporal interval of a video sequence, he henceforth no longer needs to download the entirety of the video stream, but only the desired temporal interval.

 For this, he may indicate that time interval (example of $t = 1\text{min}20\text{s}$ to $t = 2\text{min}$) in the request to transmit, in the corresponding URI:

`http://www.example.org/my_video.mp4#t=00:01:20,00:02:00.`

 Thus, the server will only search for this portion (or "fragment") of video stream identified in the request and only this is exchanged between the server and the client. Furthermore, the transmission time as well as the bandwidth used on the
25 communication network from the server to the client are reduced.

 Of course, other filtering criteria may be added in this request.

 In other applications, this addressing mechanism also makes it possible to define points of entry into a video stream for which comments, annotations or links to other resources are available, thereby facilitating hyper navigation in the video.

30 As shown above, the "Media Fragments" addressing mechanism enables elementary fragments of temporal or spatial type to be requested but does not make it possible to formulate more sophisticated requests such as the tracking of a spatio-temporal region. In what follows, the "Media Fragments" requests may be referred to using the terms "elementary requests", as opposed to the more advanced requests or
35 "complex requests".

For example if a user is considered who views a video summary or a low resolution of a video with objects or characters that are of interest to him (a ROI(t_i) region at a time t_i and a ROI(t_f) region at a time t_f , **Figure 1**, even though a greater number of regions may be considered in other instances), this user may wish to
5 retrieve a part of the video which corresponds to a region around the object of interest over a given time interval.

This amounts to formulating complex requests mixing at the same time temporal fragments and variable spatial fragments.

However, as described, the "Media Fragments" syntax does not enable
10 such a request to be expressed. The only possibility is to formulate an elementary request over the time interval considered (in our example, between t_i and t_f) with a spatial fragment corresponding to the rectangle encompassing the set of the regions of interest (maxROI rectangle in **Figure 1**).

In this case, the maxROI set of the data is extracted by the server over the
15 temporal section $[t_i, t_f]$, then sent to the client. A major drawback lies in the fact that this approach may lead to the transmission of a very large number of data, and thus unnecessarily occupy the bandwidth of the communication network between the server and the client.

Furthermore, the client must manipulate a high number of data (storage,
20 placing in random access memory, etc.) in order, before being able to exploit the data, to perform a selection of the relevant data among the data received.

There is thus a need to enable tracking of a moving region of interest in a video sequence, that is precise and of lower-cost, in particular in terms of data sent.

Earlier trials on the JPEG-2000 technology are however known, for which a
25 communication protocol (JPIP) for the exchange of portions of image or of video was developed. This protocol also relies on the use of URI addressing and a request-response exchange.

The publication US 2006/262345 in particular describes an example of implementation in which a region of interest in a video sequence is tracked. In
30 particular, the server storing the sequence computes the region of interest to send in each of the successive images according to an estimation of the motion between those successive images. Once the new region has been computed, it is sent by the server to the client who thus has its viewing window updated automatically.

A major drawback lies in the fact that by indicating the region of interest only at the start, the user cannot himself define the spatio-temporal fragments of his choice.

Another publication, US 7 423 649, describes a method of generating an animation on the basis of a region of interest and a trajectory.

This method relies on dedicated JPIP requests in the JPEG-2000 format that are generated for each image of the Motion-JPEG2000 sequence. The method consists of eliminating from that list of requests those for which data have already been received and of re-ordering the remaining requests so as to have a constant-throughput animation.

A major drawback of this approach is that a very high number of requests is generated then sent to the server.

Furthermore, these different mechanisms are closely linked to the prior knowledge of the JPEG-2000 or Motion-JPEG2000 format, working image by image, without a temporal component. In other words, they are not applicable to the coding formats that create in particular inter-image dependencies.

The present invention thus aims to mitigate at least one of the aforesaid drawbacks.

To that end, the invention concerns in particular a processing method for accessing a spatio-temporal part of a compressed video sequence, comprising a step of obtaining a request for access to said spatio-temporal part through identifying a temporal section defined between an initial time and a final time in said sequence, **characterized in that:**

- said access request identifies an initial spatial region of the sequence at the initial time, and a final spatial region of the sequence at the final time that is different from said initial spatial region;

and the method comprises the steps of:

- decomposing said access request into a plurality of elementary requests for access to a video fragment of the sequence, each elementary request identifying a fixed spatial region to extract in a temporal sub-interval of said temporal section; and
- exploiting at least one elementary request to access at least one video fragment constituting said spatio-temporal part to access.

According to the invention, the dividing up of the desired temporal section into several temporal sub-intervals enables finer fixed spatial regions to be defined. This is because, during a temporal sub-interval, the regions of interest tend to undergo

less modification than during the whole of the temporal section. This results in a reduction in the quantity of data transmitted.

Furthermore, as will be seen below, this decomposition makes it possible to modify, without complexity, the temporal sub-intervals and thus the associated fixed
5 spatial regions, to adjust the quantity of data transmitted to according to certain constraints.

Lastly, the obtainment of elementary requests as defined here enables compatibility to be maintained with the "*Media Fragments URI*" recommendation in course of development.

10 This results in efficient access to the regions of interest which vary over time, that is compatible with the aforementioned recommendation, which recommendation is moreover adapted to any type of coding of the video sequence.

In an embodiment, the method comprises a step of dynamically adjusting said decomposition of the access request during the processing of elementary
15 requests. This adjustment makes it possible in particular to modify the duration of the temporal sub-interval associated with each elementary request. This results in an adjustment of the fixed spatial regions, and thus of the data transmitted over the communication network.

It will be understood here that the present invention is concerned in
20 particular with seeking a compromise between a high number of precise elementary requests (that is to say extracting less image data) sent out for example by the client terminal and a reduced number of less precise elementary requests (extracting more image data) still sent out by the client. It is to be noted that a high number of precise requests lightens the workloads of the client terminal and of the communication
25 network (little image data conveyed), but in compensation increases the workload of the server (high number of requests to process). Conversely, a reduced number of less precise requests lightens the workload of the server, but ends up with more data exchanged (increase in the load for the network) and with more processing at the client side in order to extract the relevant data.

30 In particular, the adjusting step comprises the re-computation of the elementary requests not yet exploited into a plurality of elementary requests associated with temporal sub-intervals of modified duration. Of course, the duration of the temporal sub-intervals may be extended if the conditions for processing the data are optimum, or else be reduced if certain processing resources reach saturation. To be precise, in this

last case, the fixed spatial regions will be of reduced size, and thus a reduced number of video data will be transmitted.

According to a particular feature, the re-computation of the elementary requests not yet exploited and the modification of said duration depend on a performance criterion of a client terminal establishing said access request and/or of a server device storing said video sequence and/or of a communication network linking the client terminal and the server device. The rate of the data transmitted may thus easily be adjusted to a limiting capacity in the processing process.

According to a feature of the invention, the method comprises the steps of:

- transmitting said elementary request to a server device storing said video sequence and receiving, in return, at least one corresponding extracted fragment; and

- re-computing the elementary requests not yet exploited on the basis of said received fragment.

The quality of the responses received (by a terminal client or a proxy, for example) progressively with the exploitation of the elementary requests constitutes a good indicator of the efficiency of the settings adopted, (for example the duration of the sub-intervals), and reflects in particular the state of occupancy of the server device and/or of the communication network. Thus, an adjustment of the elementary requests on the basis thereof proves to be efficient.

In an embodiment, said decomposing depends on a performance criterion of a client terminal establishing said access request and/or of a server device storing said video sequence and/or of a communication network linking the client terminal and the server device. This provision makes it possible, from the outset, to have a relatively efficient decomposition, given the constraints of workload and resources involved in the processing.

In particular, said performance criterion is chosen from among:

- the available computation capacity on the client terminal;
- a priority value associated with the client terminal;
- a speed of processing by said client terminal;
- a time limit for access to said sequence part by the client terminal;
- the occupancy rate of the communication network bandwidth;
- a time of response by the server device to an elementary request submitted to him;

– the presence of an error response of specific type by the server device to an elementary request submitted to it.

In an embodiment of the invention, the duration of the temporal sub-intervals corresponding to the elementary requests depends on the spatial accessibility
5 of image data composing said video sequence.

"Spatial accessibility" refers to the concept whereby image data may be obtained independently from a greater or lesser number of other image data at the same point in time (that is to say depending on the presence of a greater or lesser number of "packets" or "slices" in the image).

10 As the resources necessary to manipulate image data (for example to decode them) depend on their dependencies (temporal and/or spatial), this provision makes it possible to take account of this criterion to choose optimally the decomposition adapted to the equipment and network involved.

In an embodiment of the invention, the video sequence being compressed
15 using a temporal prediction mechanism, the duration of the temporal sub-intervals corresponding to the elementary requests depends on the period separating two images without temporal dependency in the video sequence, that is to say in particular between two consecutive "Intra" images.

In particular, at least one temporal sub-interval is aligned with images of the
20 video sequence which are not compressed by temporal prediction. In other words, the intermediate times which define the temporal sub-intervals correspond to images without temporal dependency.

By way of example, they may be Intra images "I" in the MPEG-2 or MPEG-4 format or else IDR images in the H.264 format.

25 This provision makes it possible to simplify the operations of extraction at the server. This is because, in the absence of temporal dependency between two consecutive intermediate images, the server only has to be concerned with the extraction, and not with the possible spatial dependency.

It will moreover be noted that the determination of the intermediate times
30 aligned with images without temporal dependency is of low cost since these images are very often indexed and easily detectable in the compressed sequence.

Furthermore, it may be provided that said initial time be offset to align with the last image of the video sequence that precedes said initial time and that is not compressed by temporal prediction. This simplifies in particular the determination of

temporal sub-intervals aligned with *Intra* images by mere application of a synchronization distance provided in the sequence compressed.

In an embodiment, said decomposition comprises a step of performing linear interpolation, along a temporal axis, of at least one intermediate spatial region on the basis of said initial and final spatial regions. This mechanism enables simple definition of the region of interest tracked during its motion, so as to deduce therefrom the fixed spatial region for the needs of the invention.

In particular, said decomposition comprises the linear interpolation of two intermediate spatial regions corresponding to the two extreme times of a said temporal sub-interval. The extreme positions of the tracked region of interest are thus identified, to obtain a video fragment corresponding to a temporal sub-interval.

In particular, said fixed spatial region associated with said temporal sub-interval encompasses said corresponding two intermediate spatial regions. In particular, said fixed spatial region is formed from the minimum rectangle encompassing said two intermediate spatial regions. In this way, the fixed spatial region is easily constructed, while optimizing the quantity of data finally extracted.

Of course, geometric groups other than the rectangle may be implemented, in particular a circle or more complex shapes adapted to encompass several regions of interest.

As a variant of the linear interpolation, it may be provided that the decomposition comprises the following steps:

- determining at least one set of geometric data (for example a shape, an object, etc.) present in both the initial and final regions;
- estimating a variation (spatial movement and/or spatial deformation, of greater or lesser complexity: straight line, curve, etc.) of said set in said sequence between the initial and final times; and

said fixed spatial region associated with an elementary request taking into account said estimation of variation.

This provision however requires more computations (for the estimation of a complex movement in particular), but however enables more precise tracking of the desired region of interest during each temporal sub-interval.

In an embodiment of the invention, said decomposition into elementary requests is implemented at a client terminal establishing said access request. In this case, the exploitation of the elementary requests may be their transmission to the server for the purposes of being executed.

This enables the client terminal to dynamically modify the temporal decomposition of the spatio-temporal part according to the accessibility of the stream, in particular reflected by the responses from the server.

5 As a variant, said decomposition into elementary requests is implemented at a server device storing said compressed video sequence. An example of exploitation may then be the execution of the elementary requests.

This enables the server to make provision, according to the computation capacities of various clients that may access the spatio-temporal part, for different decompositions into requests for elementary fragments.

10 Furthermore, in another approach, the server may thus publish the access (for example a link to a Web page) to that spatio-temporal part using a list of requests for elementary fragments. It is these requests which are then accessible to standard clients.

15 In this case, said server device makes available to a client terminal the list of said elementary requests. For the author of a video, this configuration enables easy sharing of sub-parts of it with other users.

In another variant, said decomposition into elementary requests is implemented at an intermediate device located on a communication network between a client terminal making said access request and a server device storing said
20 compressed video sequence. The use of an intermediate device makes it possible to provide a solution for access to a video sequence spatio-temporal part, which integrates servers of fragments implementing the invention with standard servers of fragments. The intermediate device may, like the client, adapt the granularity of the requests according to the responses from the servers.

25 In particular, said intermediate device is a proxy which receives said access request from the client terminal, sends each of said elementary requests to said server device, stores each of the video fragments received from the server device and transmits a response to the access request from the client terminal that is established on the basis of said stored video fragments. This provision has a double advantage: of
30 reducing the processing load at the client terminal, and also of storing, in temporary memory of the proxy, the responses to complex spatio-temporal requests with a view to responding more rapidly to future identical requests.

In an embodiment of the invention, the decomposition of the access request, in particular by said server device, comprises iteratively determining the
35 duration of said temporal sub-intervals.

In particular, said iterative determining comprises iteratively sub-dividing the temporal sub-intervals for as long as the ratio between the estimated costs of transmitting the video fragments respectively corresponding to a sub-interval after iteration and to a sub-interval before iteration is less than a threshold value, generally a
5 threshold with the value 1/2 when iterative division by 2 is carried out.

It is to be noted that when this determining is conducted on the server device, the estimated costs may be substantially the real transmission costs since it has available the video sequence to evaluate these costs. A better evaluation of the compromise between the number of requests and the quantity of data transmitted is
10 thereby obtained.

When this determining is conducted on the client terminal or on the intermediate device, these costs may be estimated on the basis of external information, for example the compression rate over the desired sequence part.

In an embodiment of the invention, said plurality of elementary requests
15 corresponds to a plurality of distinct temporal sub-intervals fully decomposing said temporal section. In this way, the group of fragments obtained constitutes the entirety of the desired part of the video sequence.

According to a particular feature of the invention, a said elementary request comprises at least one parameter defining the fixed spatial region, and a component
20 representing a spatial variation of said fixed spatial region during said corresponding temporal sub-interval. The spatial variation in particular concerns a movement and/or a deformation of said fixed region. This provision provides a syntax that is simple (as close as possible to that provided for example in the "*Media Fragments URI*" recommendation) and efficient for accessing each elementary fragment of the spatio-
25 temporal part to access.

In a complementary manner, the invention concerns a processing device for accessing a spatio-temporal part of a compressed video sequence on the basis of an access request, said access request identifying a temporal section defined between an initial time and a final time in said sequence, **characterized in that:**

30 – said access request furthermore identifies an initial spatial region of the sequence at the initial time, and a final spatial region of the sequence at the final time that is different from said initial spatial region;

and the device comprises:

– a decomposition module adapted for decomposing said access request
35 into a plurality of elementary requests for access to a video fragment of the sequence,

each elementary request identifying a fixed spatial region to extract in a temporal sub-interval of said temporal section; and

– a module for exploiting at least one elementary request to access at least one video fragment constituting said spatio-temporal part to access.

5 The access device has similar advantages to those of the method set forth above, in particular of accessing, at optimized costs, a part of the video sequence by tracking a region of interest which may vary and move over time.

Optionally, the device may comprise means relating to the features of the method set out above.

10 The invention also concerns a system for accessing a spatio-temporal part of a video sequence stored on a server device, comprising:

– an access device as described above;
– a client terminal provided with means for establishing said access request; and

15 – a server device storing said video sequence and adapted to process at least one said elementary request to extract at least one video fragment constituting said spatio-temporal part to access.

The invention also concerns an information storage means, possibly totally or partially removable, that is readable by a computer system, comprising instructions
20 for a computer program adapted to implement a method in accordance with the invention when that program is loaded and executed by the computer system.

The invention also concerns a computer program readable by a microprocessor, comprising portions of software code adapted to implement a method in accordance with the invention, when it is loaded and executed by the
25 microprocessor.

The information storage means and computer program have features and advantages that are analogous to the methods they implement.

Still other particularities and advantages of the invention will appear in the following description, illustrated by the accompanying drawings, in which:

30 - **Figure 1** illustrates the access to a part of a video sequence using the mechanisms of the W3C recommendation on media fragments;

– **Figure 2** represents an example of a system for the implementation of the invention;

– **Figure 3** illustrates the access to a part of a video sequence by a
35 method according to the invention;

- **Figure 4** illustrates, in flow diagram form, general steps of the present invention;

- **Figures 5 and 6** illustrate the implementation of the steps of **Figure 4** via an example;

5 - **Figure 7** illustrates, in flow diagram form, steps for the adjustment of decomposition parameters for complex requests, in an embodiment in which this processing is carried out at the client terminal of **Figure 2**;

- **Figure 8** is a diagrammatic representation of a client terminal and of a server device for an implementation of the invention;

10 - **Figure 9** illustrates, in flow diagram form, steps of an implementation of the invention when the decomposition of the requests is carried out at the server device of **Figure 2**;

- **Figure 10** illustrates in more detail, in flow diagram form, steps of adjusting parameters for decomposition at the steps of **Figure 9**;

15 - **Figure 11** illustrates, in flow diagram form, steps of generating elementary requests, implemented at the time of a particular processing operation of **Figure 10**;

- **Figure 12** illustrates, in flow diagram form, steps of generating elementary requests, implemented at the time of another processing operation of **Figure 10**;

20 - **Figure 13** illustrates, in flow diagram form, steps of an implementation of the invention when the decomposition of the requests is carried out at an intermediate node between the client terminal and the server device of **Figure 2**;

- **Figure 14** illustrates examples of possible syntax when the decomposition of the requests is carried out at the intermediate node; and

- **Figure 15** shows a particular hardware configuration of a device adapted for an implementation of the method or methods according to the invention.

As represented in **Figure 2**, an example of a system 100 for the implementation of the invention comprises a server 110 linked to a client terminal 120, via a communication network 130 and possibly a proxy 140. Other client terminals (not represented) may be connected to the same network 130.

The communication network 130 may be of any nature, wired or wireless, mobile phone based or computer based such as the Internet.

According to the case concerned, the server 110 may be a web server which a user accesses using a mobile terminal 120 (telephone, personal digital

assistant, etc.) or a fixed terminal (computer, television set equipped with Internet access).

In a multimedia application, the server 110 may be merely a fixed video camera or in a network, a portable video camera or a video server.

5 The server 110 stores, in a multimedia database 150, video sequences 160 that are compressed to reduce the costs of storage and distribution. The video sequences 160 have in particular been compressed using temporal and spatial prediction mechanisms. In what follows, any image coded independently of the other images of the same sequence is termed "*synchronization image*". This is the case for
10 example of the "Intra" images in MPEG-1, 2 or 4, or the "IDR" or "SI" images in H.264. These images may be decoded by themselves, without needing the other images of the compressed video sequence. Thus, they enable a video decoder to begin to play the video sequence at a given time.

 From this definition also arises the definition of "*synchronization distance*"
15 which represents the time interval separating two synchronization images of the same video sequence. Generally this distance is constant in the sequence. It may be expressed in milliseconds or seconds or even in images when the frame rate is known.

 The invention concerns in particular the access, by the client terminal 120, to a spatio-temporal part of the video sequence 160 stored on the server device 110. A
20 spatio-temporal part results in particular from spatial filtering over target spatial regions of the images of the video sequence and of possible temporal filtering by applying the spatial filtering solely to a temporal section of the video sequence.

 The client terminal 120 (and/or the proxy 140 the case arising) and the server device 110 are equipment that implement the "*Media Fragments*" mechanisms
25 for addressing and communicating (request/response) as introduced previously. In this context, the client terminal 120 (or the proxy 140) is an apparatus sending elementary requests in accordance with the "*Media Fragments URI*" recommendation (under the W3C specification) and receiving the fragments extracted by a Media Fragments server. Similarly, the server device 110 is an apparatus adapted to process *Media*
30 *Fragments* requests and to return binary data corresponding to those requests.

Figure 3 provides a diagram of an example of a spatio-temporal part 162 to access in the video sequence 160, defined by two times, initial t_i and final t_f , temporally delimiting that part in the sequence ($t_0=0s$ being for example the start of the sequence), and by two spatial regions in the images at these two initial and final times.

In particular, in the initial image I_{ti} , the region of interest (or "ROI") is defined by a grayed horizontal rectangle $ROI(t_i)$, while in the final image I_{tf} , the region of interest is defined by a rectangle $ROI(t_f)$, also grayed. Other shapes of region of interest may however be used.

5 These four items of information (t_i , t_f , $ROI(t_i)$, $ROI(t_f)$) defining the part to access are in particular specified by a user at the dedicated interface on the client terminal 120.

 According to an embodiment of the invention, the client terminal 120 establishes a request for access REQACC to that part by identifying the temporal
10 section of said sequence, defined between an initial time t_i and a final time t_f , and the regions of interest including the initial spatial region $ROI(t_i)$ at the initial time, and the final spatial region $ROI(t_f)$ at the final time, this latter being different from said initial spatial region such that the region of interest for the user moves in the part of the sequence of interest to him.

15 As shown conjointly in **Figures 2** and **3**, this access request REQACC is decomposed or converted into a plurality of elementary requests REQ_n for access to a video fragment 164 of the sequence 160, each elementary request identifying a fixed spatial region $maxROI(t_n, t_{n+1})$ to extract in a temporal sub-interval $[t_n, t_{n+1}]$ of said temporal section $[t_i, t_f]$. Next at least one elementary request REQ_n is exploited, for
20 example sent by the client terminal or processed at said server device, to extract at least one video fragment 164 constituting said part to access.

 The extracted fragments may then be sent back, in the form of elementary responses REP_n and grouped together to form the part to access 162 returned to the client terminal 120 in the form of a response REPACC.

25 In the example of **Figure 2**, this decomposition may be carried out at the proxy 140. However, as will be seen subsequently, other embodiments may be envisaged, such as an implementation on the server 110 or preferably on the client terminal 120.

 In the example of **Figure 3**, the access request REQACC has been
30 decomposed into three elementary requests REQ_n , each in accordance with the *Media Fragments URI* recommendation. These three elementary requests respectively address the three following temporal sub-intervals: $[t_i, t_j]$, $[t_j, t_k]$ and $[t_k, t_f]$, and the three fixed spatial regions: $maxROI(t_i, t_j)$, $maxROI(t_j, t_k)$ and $maxROI(t_k, t_f)$, as described later and defined on the basis of the grayed intermediate spatial regions $ROI(t_j)$ for the time t_j
35 and $ROI(t_k)$ for the intermediate time t_k .

The three fragments then obtained are the fragments identified as 164_1 , 164_2 and 164_3 . The desired part 162 which is returned to the client terminal 120 is thus constituted by these three fragments. It is thus noted that in the first instance, the invention enables the quantity of image data transmitted to be considerably reduced
5 relative to a conventional approach using *Media Fragments* requests (the sum of the volumes of the parallelepipeds 164_n is very appreciably less than the volume of the parallelepiped of **Figure 1**).

As also described later, in an embodiment, the invention furthermore makes it possible to optimize the volume of video data exchanged between the client
10 and the server, by dynamically adjusting the decomposition of the access request REQACC during the processing of elementary requests, and in particular the duration of the temporal sub-intervals.

Figures 4 to 6 illustrate in more detail an embodiment of the invention, using an example in which, at the step S400, a user receives on his communicating
15 device (PC, mobile telephone, etc.), an address in URI form of a video 160 (**Figure 6**) produced by one of his friends as well as a summary (**Figure 5**) of that video.

This summary may be in the form of a video of very low resolution and/or of very low quality, or in the form of a series of key images such as those represented in **Figure 5**.

On the basis of the summary and a graphical interface (not represented), the user may request access, by extraction, to a part of the video 160 of his choice, by defining key times t_i and t_f (step S401) and for each key time the associated region of interest (step S402), i.e. respectively $ROI(t_i)$ defined by the information x_i , y_i , w_i and h_i , and $ROI(t_f)$ defined by the information x_f , y_f , w_f and h_f . In the Figure, only two key times
20 have been represented. Of course, a greater number of such times and their associated regions of interest will generally be defined by the user, the temporal sections then being processed in turn. In particular, care could be taken to choose key times that are not too far apart (of the order of a few times the synchronization distance), in order for the movement between the regions of interest defined over a
25 temporal section to be able to be assimilated to a linear movement. In this way, the implementation of a linear interpolation as described below will not introduce too many errors.

The user next clicks on a "download" button of the interface. The processing of the present invention may then be implemented to take on the task of
35 retrieving the video data so selected.

For this, at the step S403, the client terminal 120 translates the series of key times into a series of complex spatio-temporal requests REQACC for access to the video stored at the address represented by the URI. It may be recalled that these requests specify two items of temporal information t_i and t_f and two associated items of spatial information, $ROI(t_i)$ and $ROI(t_f)$, corresponding to two successive key times.

In the embodiment now described, the client terminal 120 incorporates a decision module which operates for the decomposition of these REQACC requests according to the invention.

As, on starting up, this module has very little or no information as to the video format and the possibilities of random access in the video (that is to say the presence and the frequency of synchronization images), it initializes a downsampling factor " δT " to an initialization value (step S404). The object of this downsampling factor is to indicate the duration of the temporal sub-intervals on the basis of which will be created the elementary requests in accordance with the *Media Fragments URI* recommendation. Below, " δT " will be designated in the same way by the term "factor" or "duration".

According to the level of prior knowledge of the video format possessed by the decision module, the factor δT may take several initial values:

- $\delta T = 1$ second where the client terminal 120 has no indication as to the video stream;
- δT takes a value of time such that the decomposition of the regions of interest ROI makes it possible to obtain a reduction in volume of the higher data at a certain threshold considered to be significant, for example a factor of 2 (comparing for example the volumes of **Figures 1** and **3**);
- $\delta T =$ synchronization distance of the video 160, if the client terminal 120 was able to obtain that information, through knowledge for example of the images without temporal dependency.

However, if a complex request REQACC has already been processed, the last δT factor used for that processing may be kept to begin the decomposition of the following complex request.

The processing continues at step S405 at which the decision module retrieves the following complex request from the list of complex spatio-temporal requests. At the first iteration, the decision module retrieves the first complex request, i.e. that which corresponds to the time interval between the first key image (I_{t_i}) and the second key image (I_{t_f}), as well as to the associated spatial regions $ROI(t_i)$ and $ROI(t_f)$.

This request REQACC is decomposed, at step S406, into a series of elementary requests REQ_n over time intervals $[t_n, t_{n+1} = t_n + \delta T]$ as illustrated in **Figure 6** for $n=i, j$ and k . It is to be noted that according to the value of δT , the final time t_f does not necessarily satisfy $t_f = t_{f-1} + \delta T$.

5 In particular, the intermediate times t_n are chosen in order for them to be aligned with the images without temporal dependency, for example with *Intra* images explicitly indexed in an MPEG-2 or MPEG-4 stream. This choice simplifies the processing operations to perform on the server.

It is implemented in particular where the synchronization distance is known.
10 Moreover, in an embodiment, the initial time t_i indicated by the user is offset to a preceding time aligned with the last image without temporal dependency (*Intra* image). Thus, the following intermediate images t_n automatically fall on images without temporal dependency since the temporal section to access is decomposed into multiples of the synchronization distance.

15 It is to be noted that an offset of t_f towards the *Intra* image following the portion indicated by the user is possible. However, the absence of offset is not detrimental, the last sub-interval created then containing fewer images than the previous ones.

A decomposition example is described in more detail below with reference
20 in particular to **Figure 10**.

As a result of this decomposition, at the intermediate times t_n so obtained, the processing determines an intermediate region of interest $ROI(t_n)$ (see **Figure 3** for example, or the region x_j, y_j, w_j, h_j in **Figure 6**) in each of the intermediate images I_{t_n} . This determination may simply rely on a linear interpolation, along the temporal axis,
25 between the initial and final regions of interest $ROI(t_i)$ and $ROI(t_f)$.

At the step S407, the client terminal 120 next exploits these standard requests REQ_n to which has been given the form $URI\#t=t_n, t_n + \delta T \&xywh=maxROI(t_n, t_n + \delta T)$, with t_n comprised between t_i and t_f of the current request REQACC, by sending them to the server 110. In each of these
30 requests, the spatial region defined by $maxROI(t_n, t_n + \delta T)$ is fixed, which ensures that these requests are in accordance with the *Media Fragments URI* recommendation.

As illustrated in the example of **Figure 3**, $maxROI(t_n, t_n + \delta T)$ is a rectangle encompassing the regions of interest $ROI(t_n)$ and $ROI(t_n + \delta T)$, in particular the smallest rectangle encompassing these regions. Other shapes for this fixed spatial region may
35 however be used, in particular depending on the shapes of the regions of interest.

At step S407, the decision module thus sends an elementary request to the server 110, in the temporal order of the requests.

Further to step S407, the server 110 also exploits the elementary request REQ_n received by executing it to extract the corresponding fragment 164_n , then sends
5 back to the client 120, either the corresponding binary data or a super-set of those data (in particular by extracting the entirety of the "slices" concerned in the request), or an error message, at step S408.

The response REP_n from the server 110 is next analyzed at S409 in order to adjust the temporal downsampling factor δT , as described below with reference to
10 **Figure 7.**

The adjustment of this factor δT and thus of the duration of the temporal sub-intervals $[t_n, t_n + \delta T]$ may depend on a performance criterion of the client terminal 120 and/or of the server device 110 and/or of the communication network 130, and in particular on the following criteria:

- 15
 - the available computation capacity on the client terminal;
 - a priority value associated with the client terminal;
 - a speed of processing by said client terminal;
 - a time limit for access to said sequence part by the client terminal;
 - the occupancy rate of the communication network bandwidth;
- 20
 - a time of response by the server device to an elementary request submitted to him;
 - the presence of an error response of specific type by the server device to an elementary request submitted to it;
 - the accessibility of the video sequence or the spectral granularity of the
- 25
 - video data in the sequence (in particular the dimension of the *slices* in the images or the synchronization distance).

Further to this adjustment, the client terminal 120 determines (test S410) whether there is still an elementary request REQ_n to be processed.

In the affirmative, the client terminal 120 sends the following elementary
30 request REQ_n at step S407.

In the negative, the client terminal 120 considers the following complex request REQ_{ACC} by returning to step S405 and loops again on the steps of decomposition and analysis so long as the last complex request has not been reached.

With reference to **Figure 7**, a description is now given of processing for adjustment of the decomposition parameters for the complex requests REQACC, in a preferred embodiment in which this processing is carried out at the client terminal 120.

Of course, this processing may also be carried out on the server device 110
5 as described below, or even at the proxy 140 as also mentioned below.

The processing commences by a step S700 of initializing the decision module, consisting of retrieving the current duration δT as well as the current complex request REQACC and the response REP_n from the server 110 to the last elementary request REQ_n processed if it exists.

10 Step S701 makes it possible to test whether the response REP_n from the server 110 is an error message.

If that is the case, the decomposition of step S406 may be called into question and performed again depending on the type of error.

In this case, the client terminal 120 determines (test S702) whether the
15 server 110 is overloaded, that is to say whether the processing resources are overloaded with regard to properly processing the elementary requests REQ_n not yet processed or sent.

To perform this test, the client terminal 120 relies on the HTTP communication protocol. To be precise, in accordance with the *Media Fragments URI*
20 recommendation, the REP_n responses are generally based on the HTTP protocol. In this case the conventional error codes are the 5xx codes defined by that same HTTP protocol and more particularly the codes 500 "*Internal Server Error*" and 503 "*Service Unavailable*".

If the server replies with the error 500 "*Internal Server Error*" the decision
25 module then has no precise information on the problems that exist at the server 110. If the preceding elementary requests did not fail (that is to say that it did indeed receive the video fragments corresponding to those preceding requests), it may thus consider that the server 110 is becoming overloaded, and thus that the test S702 is true.

Similarly, if on reception of an error response 503 "*Service Unavailable*",
30 the "*Retry-after*" header is present in that response, the test S702 is then considered as true.

In this case (output "yes" from test S702), the decision module increases the duration δT at the step S703, for example by doubling it, which leads to a new computation of the elementary requests REQ_n of the current complex request
35 REQACC (step S704), but uniquely for the elementary requests not yet sent, that is to

say generally for the temporal subsection $[t_{\alpha}, t_f]$, if REQ_{α} is the last request processed. This re-computation applies the same mechanisms as those provided at step S406, i.e. determining the intermediate times t_n with the new value δT , then the intermediate regions of interest $ROI(t_n)$ and lastly the fixed spatial regions $maxROI(t_n, t_{n+1}=t_n+\delta T)$ between two consecutive intermediate times.

On the other hand, if no indication is given by the server 110, the test S702 may then be considered as false, and the decision module leaves the factor δT unchanged. It then proceeds to the following step S410.

If the response REP_n is not an error message (test S701 false), the headers of that response as well as the response times are analyzed at step S705 in order to know whether or not the server 110 was precisely able to extract the data from the video fragment concerned in the request REQ_n , and whether or not it is overloaded.

For this, the client terminal 120 analyses the values of the "*Content-Range*" HTTP headers of the response REP_n from the server 110, and compares them with the "*Range*" headers (in particular the values representing the spatial part of the video fragment) of the request REQ_n sent by the client 120.

Because the server extracts the data from the images by whole slices to simplify its processing, the values returned by the server reflect, indirectly, the structuring of the images in slices. Thus the separation between these values is a good indication of the accessibility level of the stream, in particular in terms of spatial granularity (test S706).

The closer these values, the greater the precision with which the server 110 is capable of extracting the fragments. Conversely, the greater the separation, the more the client 120 will be called upon to extract the relevant data from the response (the entire slices returned).

This is why, in case of low spatial accessibility (test S706 false), a performance criterion of the client terminal 120 is evaluated at S707 then compared to a threshold at S708 in order to determine whether it has sufficient resources and/or capacities available to extract the relevant data.

The performance criterion of the client terminal may in particular consist in an evaluation of the workload of that terminal, that is to say the level of occupancy of internal resources (for example the processor, the random access memory, etc.), but also in an evaluation of a level of priority attributed to said terminal, of a processing speed or of a downloading time limit entered by the user.

In case of low performance of the client terminal 120 (overload, output "yes" from test S708), the decision module reduces the duration δT of downsampling, for example by half and/or by maintaining the alignment on the images without temporal dependency, at step S709. This is followed by updating of the elementary requests REQ_n of the current complex request REQ_{ACC} , by computing them again
5 using the new period δT , but uniquely for the requests not yet processed.

This reduction in particular makes it possible, ultimately, to have a greater number of elementary requests REQ_n , and thus to obtain more precise fragments 164_n . The processing load then on the client terminal is lower.

10 If the client terminal 120 is not overloaded (test S708 false), the decision module evaluates the load of the communication network 130, at the step S710. For this, the decision module keeps statistics (response time to a *ping* command, occupancy of the bandwidth [estimated for example by a ratio between the number of bytes received in response to a request and the time elapsed between sending that
15 request and the response to it]) on the volume of the data exchanged and periodically evaluates the bandwidth available for the client terminal 120.

At test S711, the decision module determines whether the network is overloaded or saturated, by comparison of the statistics and/or bandwidth with threshold values.

20 In case the network is underused (output "no" from test S711), the decision module tends to give preferential treatment to the less frequent and less precise requests by increasing the duration δT , at the step S712.

Conversely, if the network is saturated (output "yes" from test S711), the decision module will seek to send a greater number of more precise requests, by
25 reducing the duration δT (step S709).

Further to these updates of the duration δT , the decomposition of the current complex request REQ_{ACC} is re-computed for the remaining elementary requests at S704, and this new duration δT is henceforth considered for the future decompositions (step S406).

30 It is to be noted that the updating of the duration δT may also be implemented in the form of a linear function of the characteristics (in terms of computational power) of the client terminal 120 and/or of the bandwidth of the communication network 130 between the client and the server.

Figure 8 diagrammatically illustrates a client terminal 120 and a server
35 device 110 for this implementation of the invention.

These two items of equipment possess a network interface (respectively 121 and 111) with the communication network 130, so as to be able to communicate requests and responses in accordance with the *Media Fragments URI* recommendation.

5 The client terminal 120 also comprises a video stream description analyzer 122 by which it obtains an identification of a video sequence 160 to process, a description of the key times (t_n) and of the regions of interest ($ROI(t_n)$) specified by the user. All this information is stored in a database 123. It also comprises a module 124 for load analysis with regard to the use of the internal resources of the terminal 120, of
10 the network 130 and of the server 110 by analysis of the responses REP_n .

The decision module mentioned above is represented here by the module 125 and, using the information in the database 123 and from the load analysis, produces complex requests REQ_{ACC} then decomposes (possibly a simple update) those requests into elementary requests REQ_n .

15 These latter are formed at the construction block 126 before transmission to the server 110 by the interface 121.

The responses REP_n received are analyzed by the module 127 in order to provide the data from the video fragments to a video decoder, internally or externally of said client terminal 120.

20 For its part, the server 110 is linked to a device 150 for storage of the video sequences 160, which may possibly be equipped with a video encoder.

These video sequences are analyzed by an analysis module 112 to obtain a video stream description which is then stored in a database 113.

25 The requests REQ_i received via the interface 111 are analyzed by the module 114, to obtain in particular the temporal interval concerned and the fixed spatial region to extract.

Based on this information, description data in the database 113 and video data in the device 150, a fragment extraction module 115 extracts the fragments concerned in the requests received.

30 A response construction module 116 retrieves these fragment video data and arranges them in the form of a response REP_n in accordance with the *Media Fragments URI* recommendation, to be transmitted to the client 120.

With reference to **Figure 9**, a description is now given of an embodiment of the invention in which said decision module 125 is implemented in the server 110. This

embodiment makes it possible for example in particular for the author of a video to share sub-parts thereof with other authors.

For this, at step S900, the user-author views the video 160 he has just recorded. He selects passages of interest therefrom at S901 by specifying an initial
5 time t_1 and a final time t_2 , then frames the regions of interest $ROI(t_1)$ and $ROI(t_2)$ at S902.

At step S903, the user may specify other key times $\{t_c\}$ in the selected passage, in particular if the latter is very long and if the regions of interest vary greatly. The key times are in particular chosen aligned with images of the video sequence
10 which do not have temporal dependency. On this occasion, he also specifies the regions of interest $ROI(t_c)$ for each of these key times, on which will be based the interpolation of the intermediate regions ROI and the decomposition .

As input to the step S904, a list is thus available of temporal sections $[t_i, t_f]$ and of associated regions of interest $ROI(t_i)$ and $ROI(t_f)$, with $t_i=t_1$ or t_c and $t_f=t_c$ or t_2 .
15 This list is equivalent to an aforementioned list of complex requests REQACC.

It is to be noted, as a variant, that this information may be sent by a client terminal to the server, in a REQACC request format as explained below with regard to **Figure 13**.

Step S904, described in **Figure 10**, then consists of translating one of
20 these temporal sections and associated regions of interest into a series of elementary requests REQ_n in accordance with the *Media Fragments URI* recommendation.

The list of elementary requests thus produced is temporarily stored on the server 110 at step S905.

Step S906 enables each of these temporal sections to be processed
25 successively, until the final time t_2 is reached.

When the last temporal section has been processed (S906 true), the list of elementary requests enabling each spatio-temporal part 162 of the video to be obtained is then exploited by the server, here published (step S907) to be made available and communicable to any user/device wishing to retrieve the new video so
30 obtained.

For example, the user may send this list of requests by email to his friends in the form of a *playlist*. This list, which is loaded onto a video sharing server on the Web at the same time as the video itself, thus enables any client 120 connecting to that sharing server, to play or download that subset of the video.

Furthermore, if the author of the video only wishes to publish the spatio-temporal part (and not the whole video) on a video sharing Web server, also client terminal 120, the latter may easily download the fragments 164 composing that video part, so avoiding the author having to load the whole of his video on the sharing server.

5 With reference to **Figures 10 to 12**, a description is now given of step S904 of decomposing the spatio-temporal requests REQACC into elementary requests REQ_n.

The first step S1000 consists, for the decision module of the server 110, of retrieving the times t_i and t_f delimiting the current temporal section as well as the
10 associated regions ROI(t_i) and ROI(t_f).

Next, at S1001, the server selects, from a predefined list, what type of client 120 he wishes to exchange his video with (mobile telephone, PC, Web server, etc.). This predefined list classifies the types of clients 120 according to their computation/processing capacities.

15 A quick analysis of the capacities of the type of client chosen is carried out at S1002, by comparison for example with predefined threshold values.

If the type is a type of client with limited computation capacities (test S1002 true), the server 110 then generates elementary requests over short intervals of time (δT small) with more precise regions of interest ROI at S1003. To be precise, as the
20 client terminal 120 will not necessarily be able to process a high volume of data, the video fragments that will be sent are made as precise as possible.

Figure 11 illustrates an embodiment of this step S1003 during which a complex spatio-temporal request REQACC is decomposed into elementary requests REQ_n over a small time interval (δT small) and over which the same fixed spatial region
25 applies.

For this processing, the decision module 125 commences by retrieving, at S1100, the synchronization distance T of the recorded video.

At 1101, it retrieves the initial and final times t_i and t_f corresponding to the current complex spatio-temporal request to determine the duration D of the temporal
30 section corresponding to that current request.

The test S1102 compares this duration D to the synchronization distance T.

If the duration D of the temporal section is equal to or less than the synchronization distance T, the decision module 125 sets $\delta T = D$ at S1103. This means that the temporal section is not decomposed into sub-intervals. In this case, a *Media*

Fragments URI compatible request is nevertheless formed between the times t_i and t_f for the spatial region $\max\text{ROI}(t_i, t_f)$.

If the duration D of the temporal section is greater than the synchronization distance T , the server then analyses (S1104) the video stream in terms of spatial accessibility (organization of the packets/slices enabling the synchronization; size of those packets/slices; number of packets/slices per image; etc.).

The higher the number of slices (output "yes" from test S1105), the better is the spatial accessibility and the greater the duration δT that the decision module can consider. In this case, it sets $\delta T = D$ at S1103.

Conversely, if the stream does not have numerous spatial resynchronization points (coarse granularity – output "no" from test S1105), the decision module sets $\delta T = T$ (S1106) in order to give precedence to the retrieval of video fragments that are more precise.

Returning to **Figure 10**, conversely if the type of client chosen possesses high computation capacities (test S1002 false), the server selects (at S1004), from a list of predefined network bandwidths, a bandwidth on which it envisages communicating. This may for example be a choice from among the different means of communication available to the server (wifi link, Ethernet link, mobile telephone link, ADSL link, etc.).

If it is a rather narrow bandwidth, less than the megabit per second (Mbit/s) for example, (the test S1005 is then true), the server 110 generates requests over shorter intervals of time with more precise regions of interest ROI (S1003 as described earlier), that is to say with a δT supposed to be low, close to the synchronization distance known by the server.

It is thus noted that step S1003 is carried out as soon as the server considers that the client terminal 120 or the network 130 is liable to experience saturation.

Conversely, if the server 110 selects a wide bandwidth, it may generate less precise elementary requests at S1006, typically with a δT that is a multiple of the synchronization distance.

This step is thus carried out principally when the server considers that the recipient 120 of the fragments and the network 130 are not liable to experience saturation. **Figure 12** illustrates an embodiment of this step S1006.

At the start of processing the server 110 possesses information relative to the current complex request REQACC, i.e. in addition to the video sequence

concerned, the times t_i and t_f delimiting the current temporal section as well as the associated regions $ROI(t_i)$ and $ROI(t_f)$.

The decision module starts by retrieving the synchronization distance T of the video sequence at S1200.

5 At S1201, it retrieves the initial and final times t_i and t_f corresponding to the current complex spatio-temporal request to determine the duration D of the temporal section corresponding to that current request.

The test S1202 determines whether this duration D is greater than twice the synchronization distance T .

10 In the negative, the decision module sets $\delta T = D$ at S1203 (in similar manner to S1103).

In the affirmative, the current temporal section may be decomposed into smaller sub-intervals of time, notably for example of the order of T . In this case, the decision module 125 performs a loop to search for the best temporal decomposition at
15 steps S1204 to S1212.

In detail, at S1204, it initializes two parameters $d_1 = D/2$ and $d_2 = D$.

Next, at S1205, it computes the intermediate regions of interest $\max ROI(t_n, t_n + \delta T)$ for the current value of the factor $\delta T = d_1$. These intermediate regions are in particular obtained by linear interpolation along the temporal axis, between the
20 initial and final regions $ROI(t_i)$ and $ROI(t_f)$.

At S1206, the decision module computes the transmission cost, denoted $C1$, of the video fragments for the current factor δT .

These costs correspond in particular to the quantity of video data to transmit moreover taking into account the accessibility of the video stream (which
25 requires the transmission of the whole packets/slices). It may be estimated by approximation by computing the volumes 164_n , or on the basis of statistics (cost in bits of the macro-blocks of the images and regions to transmit) available to the coder, or by extraction of the corresponding data and retrieval of their size.

At S1207, it computes the transmission cost, denoted $C2$, of the video
30 fragments for the factor δT at the preceding iteration. The computation may consists in retrieving the cost $C1$ computed at the preceding iteration.

At S1208, the decision module 125 compares the costs $C1$ and $C2$ relative to a threshold value, for example $C1/C2$ with $1/2$.

If the decomposition enables a significant saving in terms of rate (for
35 example: $C1 < C2/2$ – output "yes" from test S1208), the decision module continues the

decomposition into more precise requests, while first of all ensuring that the factor δT (d_1) remains greater than the synchronization distance T (test S1209) then by updating (S1210) the two parameters: $d_1 = d_1/2$ and $d_2 = d_1$. It then loops again to step S1205.

5 If the decomposition does not enable a saving (output "no" from test S1208), the decision module chooses $\delta T = d_2$ (E1211), which thus corresponds to the decomposition of the preceding iteration.

Lastly, if the factor δT risks being lower than T (output "no" from test S1209), the decomposition stops with the value d_1 (S1212) of the decomposition of the current iteration as factor δT .

10 Thus, the decomposition continues so long as the saving in terms of transmission cost remains worthwhile (test S1108 true) and so long as the time interval obtained remains greater than or equal to T (test S1110 true). A good compromise between the precision of the elementary requests REQ_n , their frequency and the size of the data to transmit is thus obtained.

15 Returning to **Figure 10**, once the duration δT has been obtained either at step S1003 or at step S1006, the processing continues at S1007 with the downsampling of the current temporal section into sub-intervals of duration equal to δT . It is to be noted that this step may already have been carried out at step S1205.

20 Next the intermediate regions of interest $ROI(t_n)$ for each of the intermediate times (t_n) separating two sub-intervals are computed (S1008) by linear interpolation on the temporal axis between the initial and final regions $ROI(t_i)$ and $ROI(t_f)$ (also carried out at step S1205).

25 An elementary request REQ_n for access to the part 162 of video sequence 160 is then generated (S1009) by indicating a said sub-interval and the rectangle $\max ROI(t_n, t_{n+1} = t_n + \delta T)$ encompassing the two computed intermediate regions $ROI(t_n)$ and $ROI(t_{n+1})$ corresponding to that interval.

Steps S1010 and S1011 enable iteration over all the sub-intervals until the end t_f of the current temporal section is attained.

30 With reference to **Figure 13**, a description is now given of an embodiment of the invention in which said decision module 125 is implemented in an intermediate node of the network 130, in particular in a proxy or "proximity" server 140.

35 In this embodiment, the client terminal 120 sends the proxy 140 complex spatio-temporal requests $REQACC$. A particular syntax (not compatible with the *Media Fragments URI* recommendation in its current state) is adopted for transmitting all the necessary parameters.

Figure 14 illustrates examples of possible syntax. In these examples, the client terminal 120 signals the temporal section $[t_i, t_f]$ of the desired part 162 in conventional manner, and gives information on the spatial part by adding a component "t" to the conventional parameter "xywh": "xywh_t" in which the component "t" enables a spatial variation to be indicated (including spatial movement and/or spatial deformation) of the region of interest $ROI(t_i)=xywh$ over time.

In the first example given relative to a simple translation (the region of interest maintains its dimensions), the component "t" corresponds to a translation vector (T_x, T_y) of the region of interest between t_i and t_f .

In the second example given relative to a translation and a zoom with use of the SVG syntax, the parameter "t" combines a translation vector and transformation factors, here homothetic factors (sx, sy) along the horizontal and vertical axes.

Optionally, the client terminal 120 may also transmit, in its request, a "granularity" parameter indicating whether or not it wishes to obtain fine granularity (precise and more frequent requests). This is the case of the third example provided in which the parameter "t" is an affine transformation matrix A.

Of course parameters describing more complex transformations may be provided.

Step S1300 corresponds to the reception of these requests by the proxy 140. The latter then determines the presence or not of a granularity indication at S1301. For this it goes through the complex spatio-temporal request and stores the parameters present in the request (time interval requested, first spatial region requested, transformation parameters to apply and, the case arising, the parameter indicating whether the client wishes to have fine granularity or not).

If the request indicates a request for fine granularity (output "yes" from test S1301), at S1302 the proxy 140 initializes the factor δT to a value T which may either be one second, or the synchronization period if it is known to the proxy 140, or the last value used by the proxy for the video requested.

This factor δT is applied to the complex request REQACC received at S1300 to generate a list of elementary requests REQ_n over time intervals of duration δT (S1303).

The elementary requests REQ_n are then successively submitted to the server 110 (for example in their temporal order). Step S1304 first of all consists in transmitting the first requests REQ_i corresponding to the interval $[t_i, t_i+\delta T]$, then, at the following iterations, the requests REQ_n .

At S1305, the response REP_n from the server 110 is received and kept in memory of the proxy. This response comprises in particular the extracted video fragment 164_n .

Further to this response, the proxy may decide to adjust its factor δT at step S1306. Conditions justifying the adjustment of the factor δT correspond for example to the following cases:

- an error of the server. This case leads to the doubling of the factor δT ;
- a temporal extraction greater than the time interval requested. This case leads to adjusting the value of factor δT to the duration of the extracted temporal segment;
- a spatial extraction different from the requested region. If the extracted region is larger than the requested region, this case leads to dividing the factor δT by 2; otherwise in doubling the factor δT since it indicates that the requested video is providing good spatial granularity.

Other performance criteria cited previously may also be used.

If an adjustment is necessary, the processing loops on step S1303 to update the decomposition into elementary requests, in particular for the requests not yet submitted.

If no adjustment is necessary, the decision module performs a loop on the following elementary request at S1307 until the last one.

When the last request has been reached (test S1307 false), the proxy 140 sends a response $REPACC$ to the client 120 in the form of an HTTP response with multiple byte ranges, each byte range corresponding to a response REP_n from the server 110 stored at S1305.

If the client has not expressed any wish as to the granularity of the response or indicated a coarse granularity (output "no" from test S1301), at S1309 the proxy 140 then applies a decomposition similar to that described above with reference to steps S1202 to S1212.

The elementary requests REQ_n so obtained are successively sent to the server 110 (S1310) and each response REP_n is stored (S1311) to form the response $REPACC$ at S1308.

With reference now to **Figure 15**, a description is given by way of example of a particular hardware configuration of a device for video sequence access adapted for an implementation of the methods according to the invention.

A device implementing the present invention is for example a micro-computer 50, a workstation, a personal assistant, or a mobile telephone connected to different peripherals. According to still another embodiment of the invention, the device takes the form of a camera provided with a communication interface to enable
5 connection to a network.

The peripherals connected to the device comprise for example a digital camera 64, or a scanner or any other means of image acquisition or storage, connected to an input/output card (not shown) and supplying multimedia data, for example of video sequence type, to the device according to the invention.

10 The device 50 comprises a communication bus 51 to which there are connected:

- a central processing unit CPU 52 taking for example the form of a microprocessor;

- a read only memory 53 in which may be contained the programs whose
15 execution enables the implementation of the methods according to the invention. It may be a flash memory or EEPROM;

- A random access memory 54, which, after powering up of the device 50, contains the executable code of the programs of the invention necessary for the implementation of the invention. As this memory 54 is of random access type (RAM), it
20 provides fast accesses compared to the read only memory 53. This RAM memory 54 stores in particular the various images and the various blocks of pixels as the processing is carried out on the video sequences;

- a screen 55 for displaying data, in particular video and/or serving as a graphical interface with the user, who may thus interact with the programs according to
25 the invention, using a keyboard 56 or any other means such as a pointing device, for example a mouse 57 or an optical stylus;

- a hard disk 58 or a storage memory, such as a memory of compact flash type, able to contain the programs of the invention as well as data used or produced on implementation of the invention;

- an optional diskette drive 59, or another reader for a removable data carrier, adapted to receive a diskette 63 and to read/write thereon data processed or to process in accordance with the invention; and

- a communication interface 60 connected to the telecommunications network 130, the interface 60 being adapted to transmit and receive data.

In the case of audio data, the device 50 is preferably equipped with an input/output card (not shown) which is connected to a microphone 62.

The communication bus 51 permits communication and interoperability between the different elements included in the device 50 or connected to it. The
5 representation of the bus 51 is non-limiting and, in particular, the central processing unit 52 unit may communicate instructions to any element of the device 50 directly or by means of another element of the device 50.

The diskettes 63 can be replaced by any information carrier such as a compact disc (CD-ROM) rewritable or not, a ZIP disk or a memory card. Generally, an
10 information storage means, which can be read by a micro-computer or microprocessor, integrated or not into the video sequence processing device, and which may possibly be removable, is adapted to store one or more programs whose execution permits the implementation of the methods according to the invention.

The executable code enabling the device for accessing a spatio-temporal
15 part of a video sequence to implement the invention may equally well be stored in read only memory 53, on the hard disk 58 or on a removable digital medium such as a diskette 63 as described earlier. According to a variant, the executable code of the programs is received by the intermediary of the telecommunications network 130, via the interface 60, to be stored in one of the storage means of the device 50 (such as the
20 hard disk 58) before being executed.

The central processing unit 52 controls and directs the execution of the instructions or portions of software code of the program or programs of the invention, the instructions or portions of software code being stored in one of the aforementioned storage means. On powering up of the device 50, the program or programs which are
25 stored in a non-volatile memory, for example the hard disk 58 or the read only memory 53, are transferred into the random-access memory 54, which then contains the executable code of the program or programs of the invention, as well as registers for storing the variables and parameters necessary for implementation of the invention.

It will also be noted that the device implementing the invention or
30 incorporating it may be implemented in the form of a programmed apparatus. For example, such a device may then contain the code of the computer program(s) in a fixed form in an application specific integrated circuit (ASIC).

The device described here and, particularly, the central processing unit 52, may implement all or part of the processing operations described in relation with

Figures 1 to 14, to implement the methods of the present invention and constitute the devices of the present invention.

The preceding examples are only embodiments of the invention which is not limited thereto.

- 5 In particular, the alignment of the initial t_i , final t_f , and intermediate or key t_n/t_c times with images having no temporal dependency may be advantageously implemented in any one of the implementations of the invention.

CLAIMS

1. A processing method for accessing a spatio-temporal part (162) of a compressed video sequence (160), comprising a step of obtaining (S403, S901-903, S1000, S1300) a request for access (REQACC) to said spatio-temporal part by identifying a temporal section defined between an initial time (t_i) and a final time (t_f) in said sequence, **characterized in that:**
- said access request (REQACC) identifies an initial spatial region ($ROI(t_i)$) of the sequence at the initial time (t_i), and a final spatial region ($ROI(t_f)$) of the sequence at the final time (t_f) that is different from said initial spatial region;
- and the method comprises the steps of:
- decomposing (S406, S904, S1003, S1006, S1303, S1309) said access request into a plurality of elementary requests (REQ_n) for access to a video fragment (164_n) of the sequence, each elementary request identifying a fixed spatial region ($maxROI(t_n, t_{n+1})$) to extract in a temporal sub-interval of said temporal section; and
 - exploiting (S407, S907, S1310) at least one elementary request to access at least one video fragment (164_n) constituting said spatio-temporal part to access (162).
2. A method according to claim 1, comprising a step of dynamically adjusting (S409, S704, S1306) said decomposition of the access request (REQACC) during the processing of elementary requests.
3. A method according to claim 2, wherein the adjusting step comprises the re-computation of the elementary requests (REQ_n) not yet exploited into a plurality of elementary requests associated with temporal sub-intervals of modified duration (δT).
4. A method according to claim 3, comprising the steps of:
- transmitting (S407, S1304, S1310) said elementary request (REQ_n) to a server device (110) storing said video sequence and receiving (S408, S1305, S1311), in return, at least one corresponding extracted fragment ($164_n, REP_n$); and
 - re-computing the elementary requests (REQ_n) not yet exploited on the basis of said received fragment.
5. A method according to one of the preceding claims, in which said decomposition or possible re-computation of the elementary requests (REQ_n) not yet exploited and the associated modification of said duration (δT) depend on a performance criterion of a client terminal (120) establishing said access request and/or

of a server device (110) storing said video sequence and/or of a communication network (130) linking the client terminal and the server device.

6. A method according to one of the preceding claims, in which the duration (δT) of the temporal sub-intervals corresponding to the elementary requests (REQ_n) depends on the spatial accessibility of image data composing said video sequence (160).

7. A method according to one of the preceding claims, in which the video sequence (160) being compressed using a temporal prediction mechanism, the duration (δT) of the temporal sub-intervals corresponding to the elementary requests (REQ_n) depends on the period (T) separating two images without temporal dependency in the video sequence.

8. A method according to claim 7, wherein a temporal sub-interval is aligned with images of the video sequence which are not compressed by temporal prediction.

9. A method according to claim 7 or 8, in which said initial time (t_i) is offset to align with the last image of the video sequence that precedes said initial time and that is not compressed by temporal prediction.

10. A method according to one of the preceding claims, in which said decomposition comprises a step of performing linear interpolation (S1205), along a temporal axis, of two intermediate spatial regions (ROI(t_n), ROI(t_{n+1})) corresponding to the two extreme times (t_n , t_{n+1}) of a said temporal sub-interval, on the basis of said initial (ROI(t_i)) and final (ROI(t_f)) spatial regions.

11. A method according to claim 10, wherein said fixed spatial region (maxROI(t_n , t_{n+1})) associated with said temporal sub-interval encompasses said corresponding two intermediate spatial regions (ROI(t_n), ROI(t_{n+1})).

12. A method according to one of claims 1 to 11, wherein said decomposition into elementary requests (REQ_n) is implemented at a client terminal (120) establishing said access request (REQACC).

13. A method according to one of claims 1 to 11, wherein said decomposition into elementary requests (REQ_n) is implemented at a server device (110) storing said compressed video sequence (160).

14. A method according to claim 13, wherein said server device (110) makes available (S907) to a client terminal (120) the list of said elementary requests.

15. A method according to one of claims 1 to 11, wherein said decomposition into elementary requests (REQ_n) is implemented at an intermediate

device (140) located on a communication network (130) between a client terminal (120) making said access request (REQACC) and a server device (110) storing said compressed video sequence (160).

5 16. A method according to claim 15, wherein said intermediate device (140) is a proxy which receives (S1300) said access request (REQACC) from the client terminal (120), sends (S1304) each of said elementary requests (REQ_n) to said server device (110), stores (S1305) each of the video fragments (164_n) received from the server device (110) and transmits (S1308) a response (REPASS) to the access request (REQACC) from the client terminal (120) that is established on the basis of
10 said stored video fragments.

 17. A method according to one of the preceding claims, in which the decomposition (S406, S904, S1003, S1006, S1303, S1309) of the access request (REQACC) comprises iteratively determining (S1210) the duration (δT) of said temporal sub-intervals.

15 18. A method according to claim 17, wherein said iterative determining comprises iteratively sub-dividing (S1210) the temporal sub-intervals for as long as the ratio between the estimated costs ($C1$, $C2$) of transmitting the video fragments respectively corresponding to a sub-interval after iteration and to a sub-interval before iteration is less than (S1208) a threshold value.

20 19. A method according to one of the preceding claims, in which a said elementary request (REQ_n) comprises at least one parameter ($xywh$) defining the fixed spatial region ($maxROI(t_n, t_{n+1})$), and a component (t) representing a spatial variation of said fixed spatial region during said corresponding temporal sub-interval.

 20. A processing device for accessing a spatio-temporal part (162) of a
25 compressed video sequence (160) on the basis of an access request (REQACC), said access request identifying a temporal section defined between an initial time (t_i) and a final time (t_f) in said sequence, **characterized in that:**

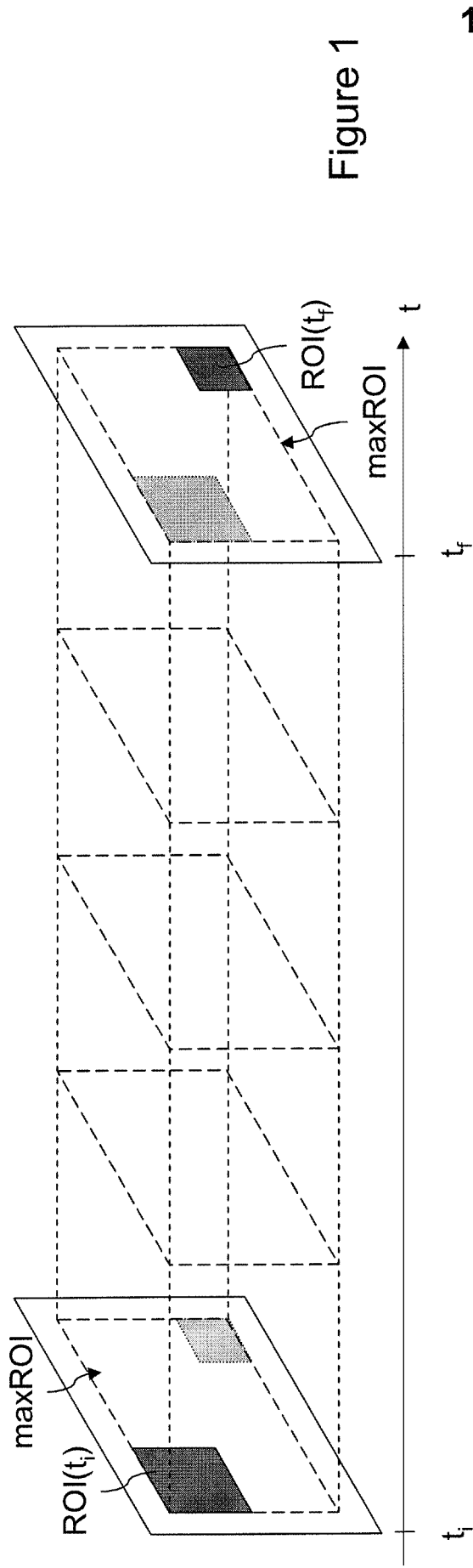
 – said access request (REQACC) furthermore identifies an initial spatial region ($ROI(t_i)$) of the sequence at the initial time (t_i), and a final spatial region ($ROI(t_f)$)
30 of the sequence at the final time (t_f) that is different from said initial spatial region;
 and the device comprises:

 – a decomposition module (125) adapted for decomposing said access request into a plurality of elementary requests for access (REQ_n) to a video fragment (164_n) of the sequence, each elementary request identifying a fixed spatial region
35 ($maxROI(t_n, t_{n+1})$) to extract in a temporal sub-interval of said temporal section; and

– a module for exploiting at least one elementary request to access at least one video fragment (164_n) constituting said spatio-temporal part (162) to access.

21. A means of information storage, possibly totally or partially removable, that is readable by a computer system, comprising instructions for a computer program
5 adapted to implement the method according to any one of claims 1 to 19, when the program is loaded and executed by the computer system.

22. A computer program product readable by a microprocessor, comprising portions of software code adapted to implement the method according to any one of claims 1 to 19, when it is loaded and executed by the microprocessor.



1/10

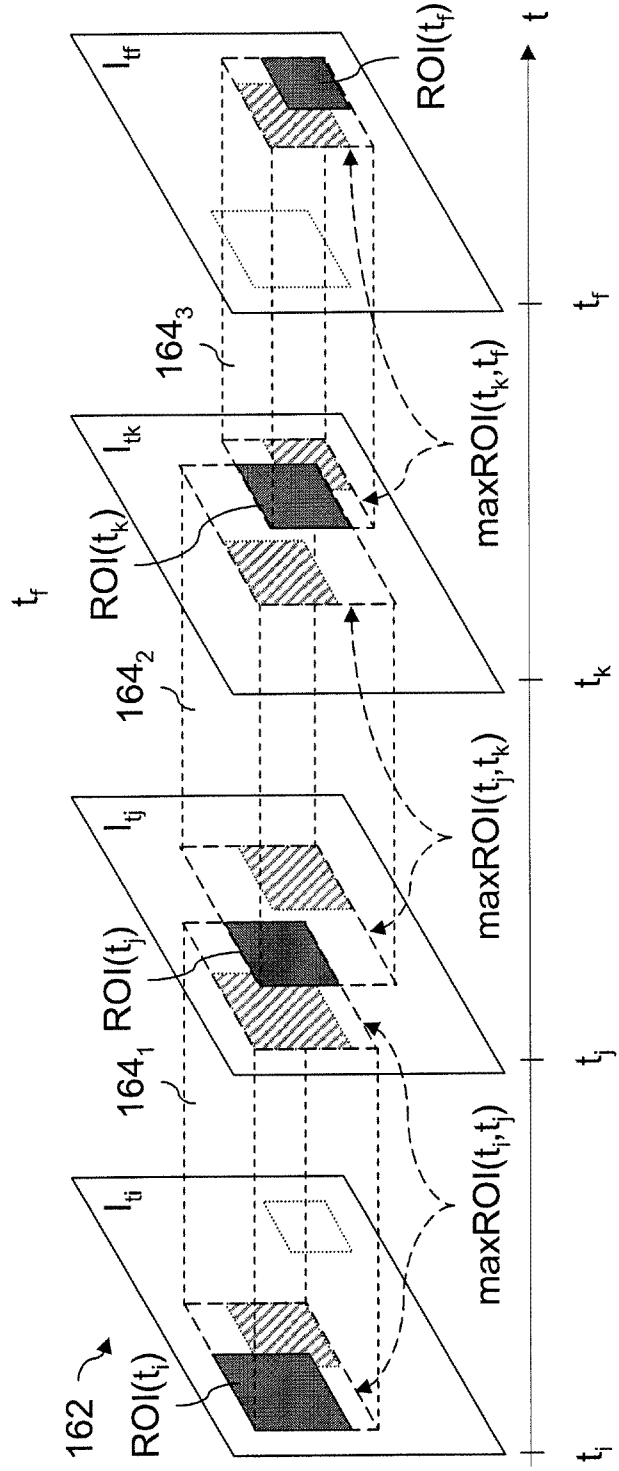


Figure 3

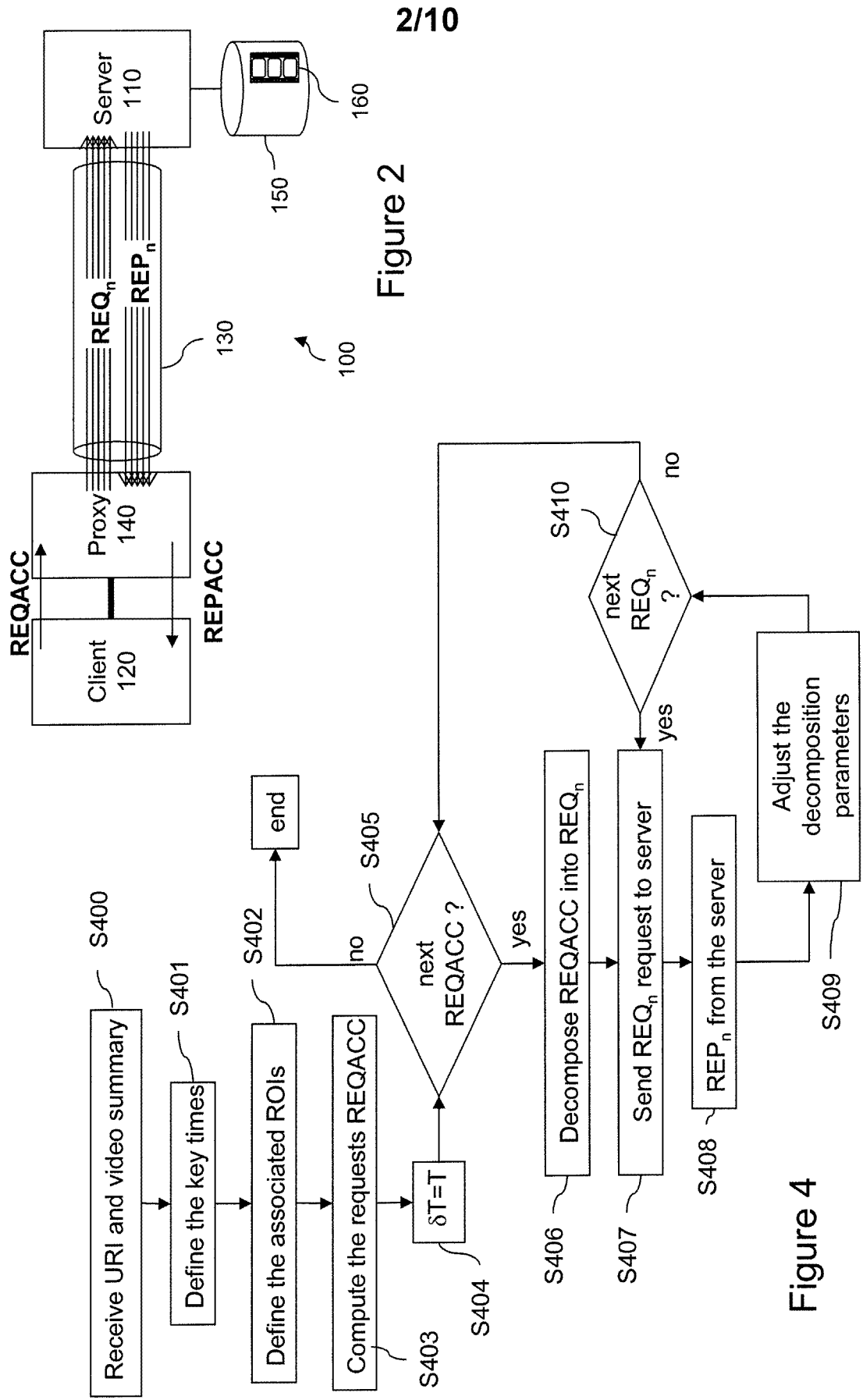


Figure 4

3/10

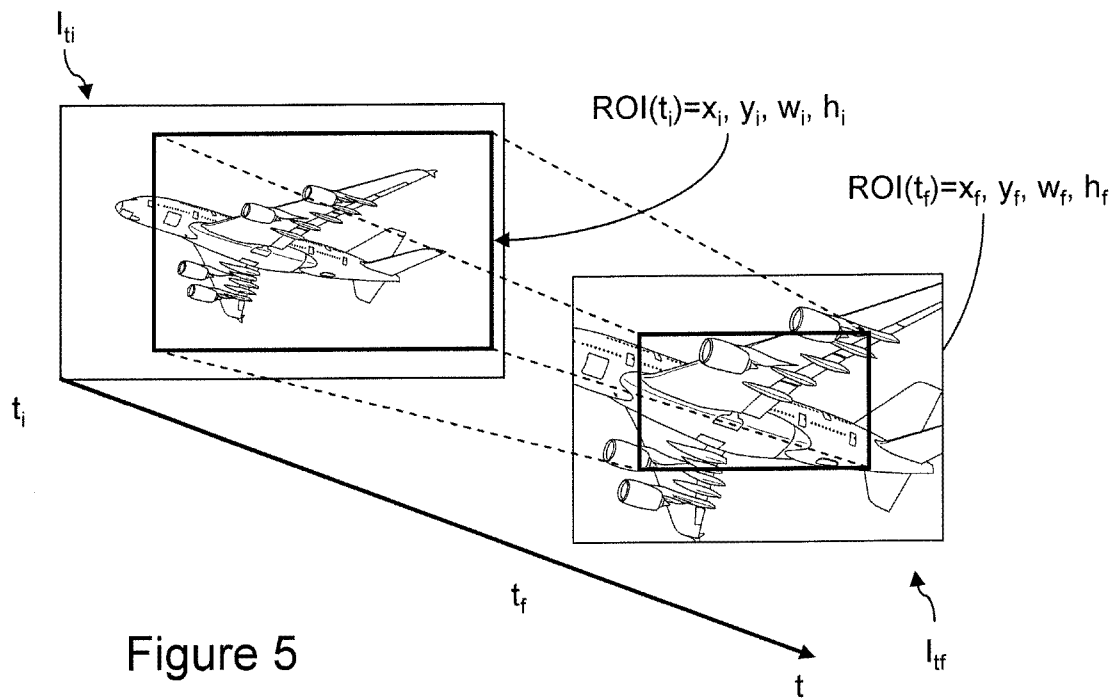


Figure 5

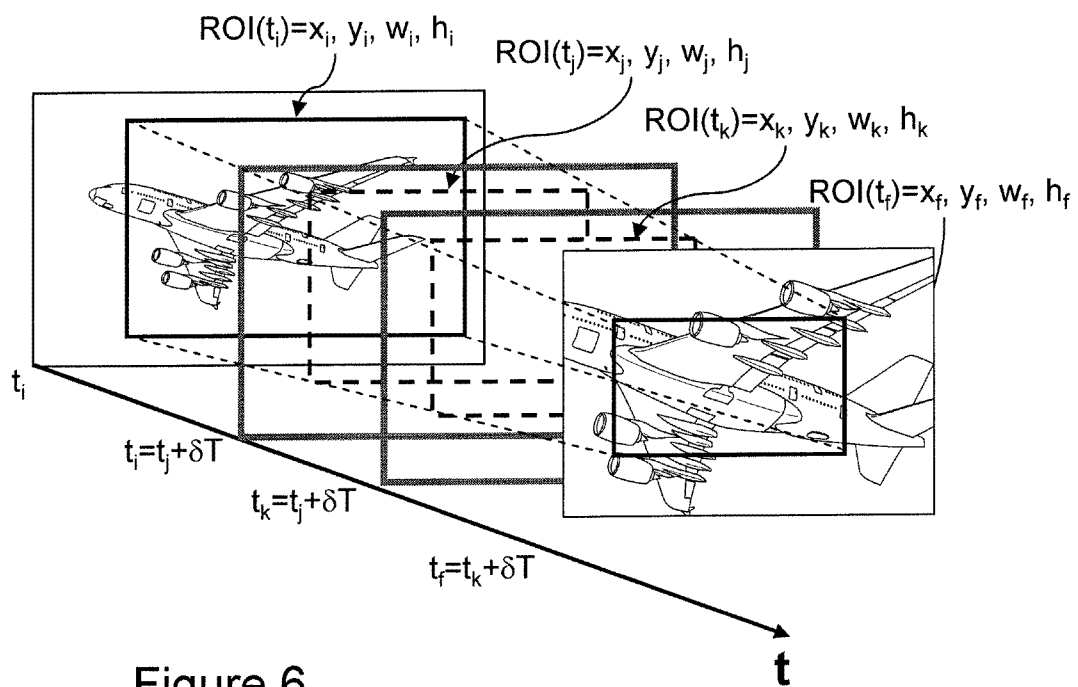


Figure 6

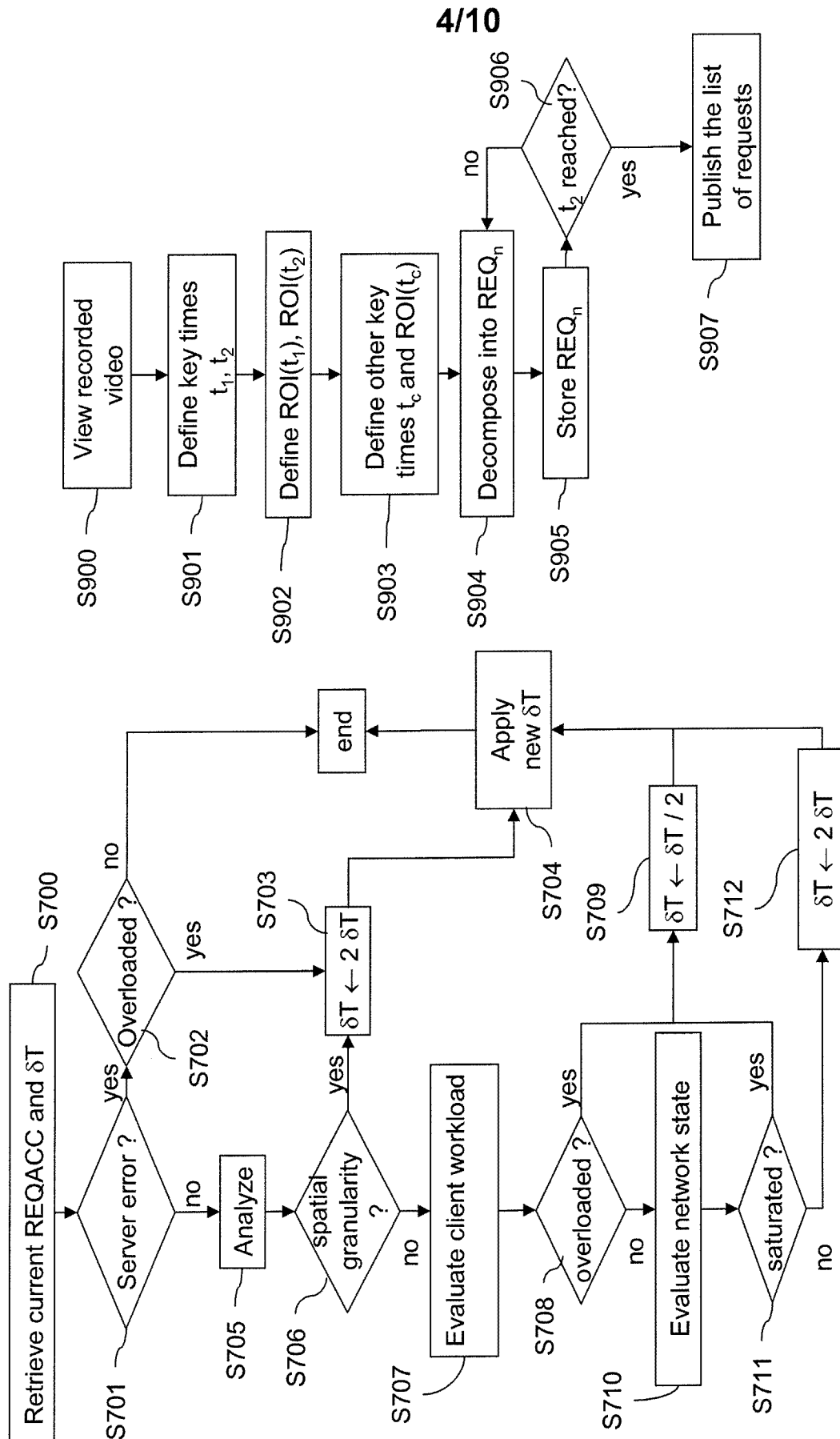


Figure 9

Figure 7

5/10

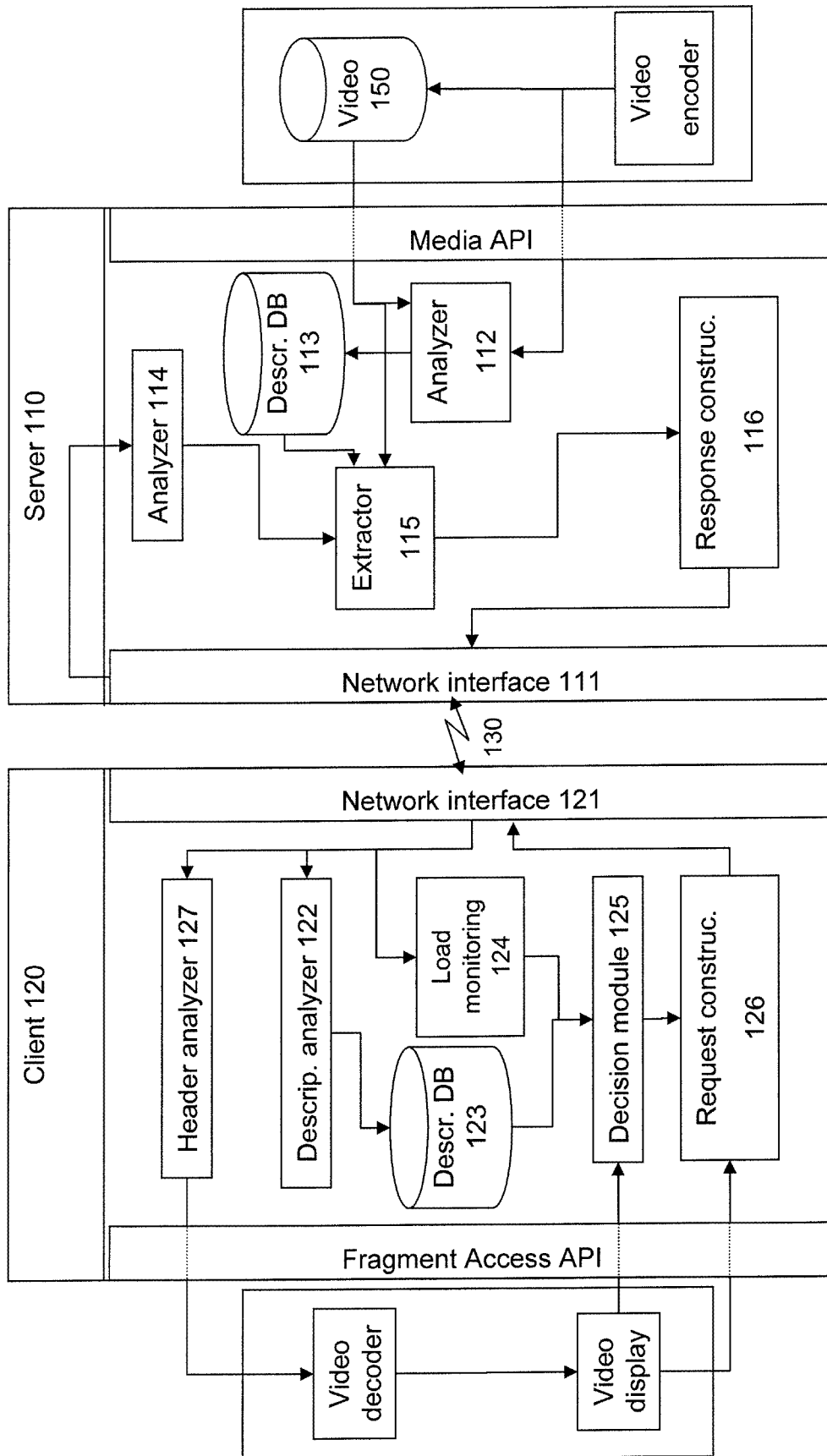


Figure 8

6/10

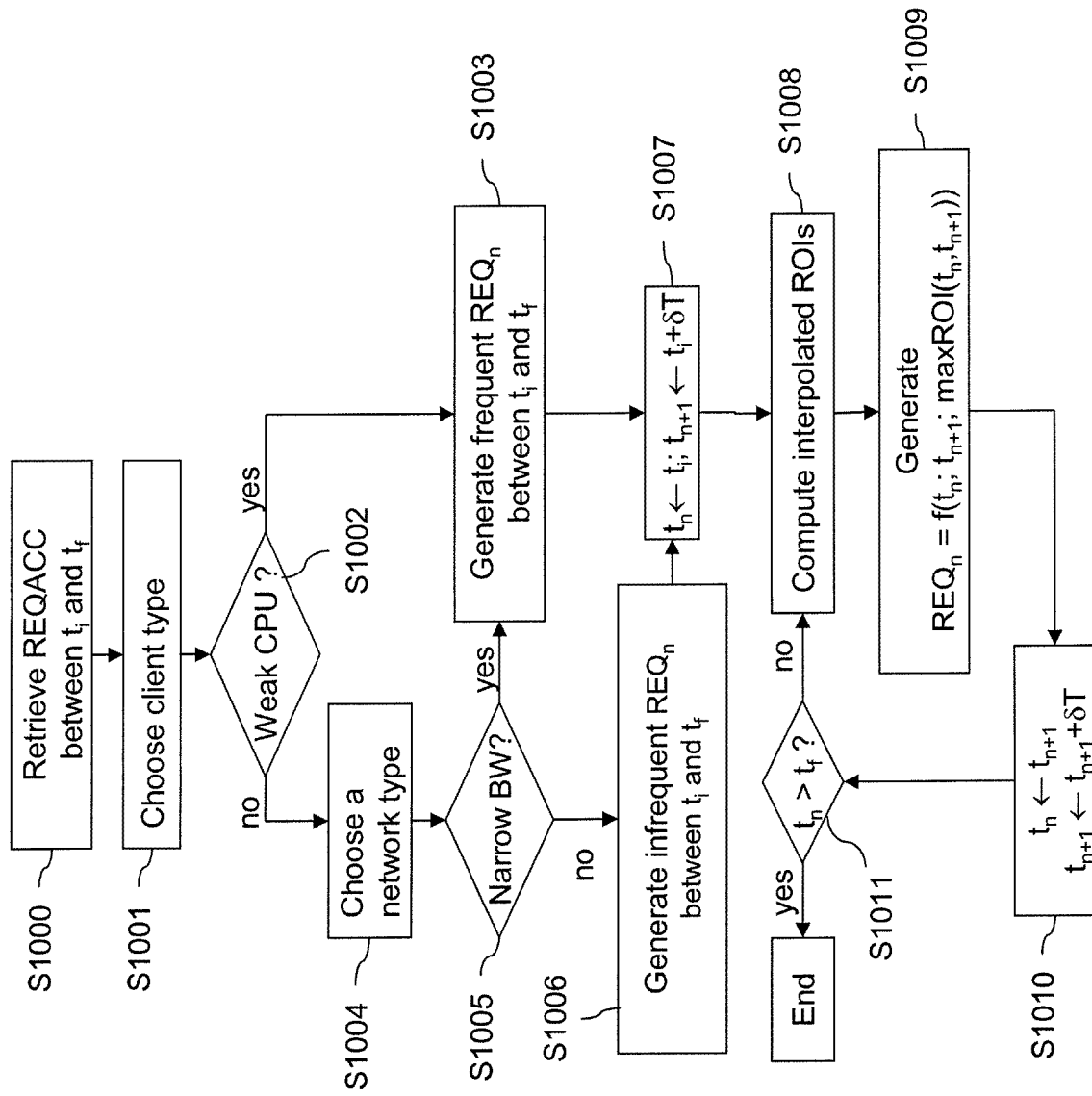


Figure 10

7/10

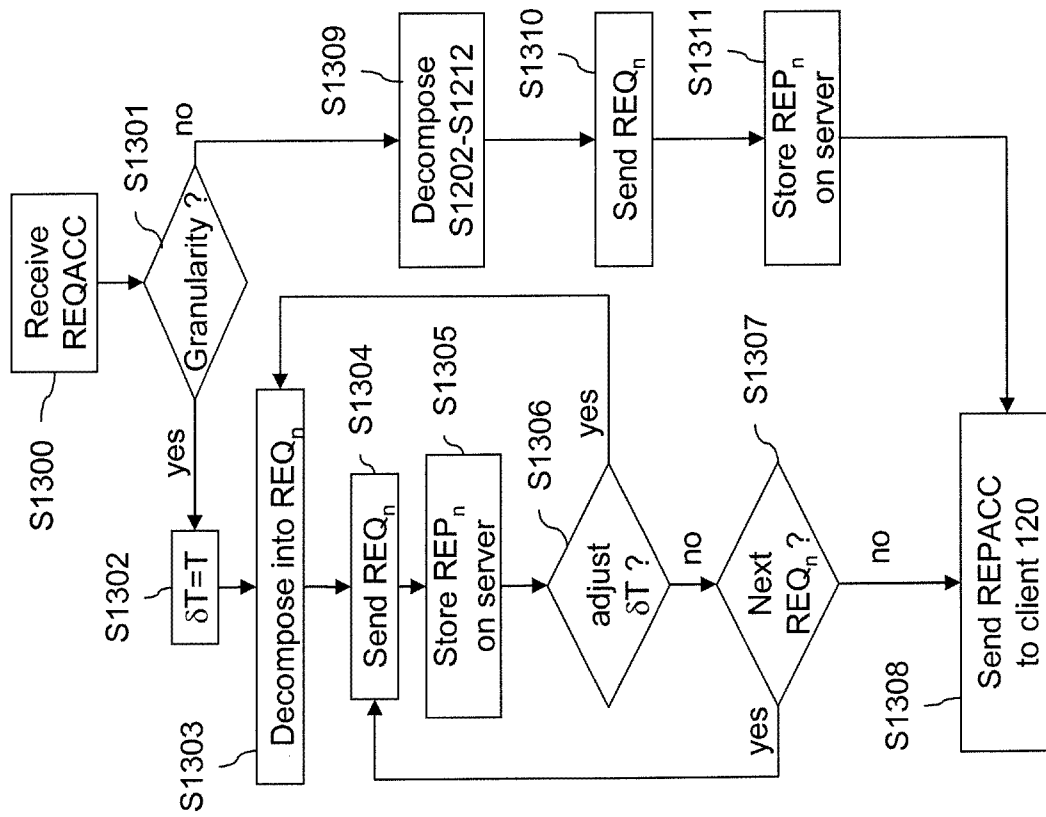


Figure 13

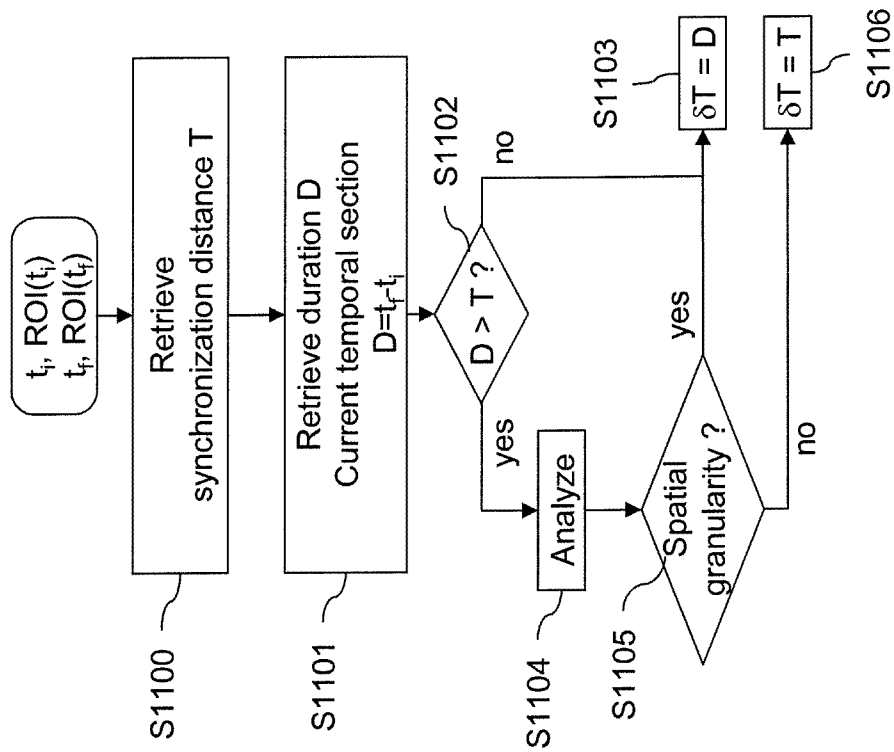


Figure 11

8/10

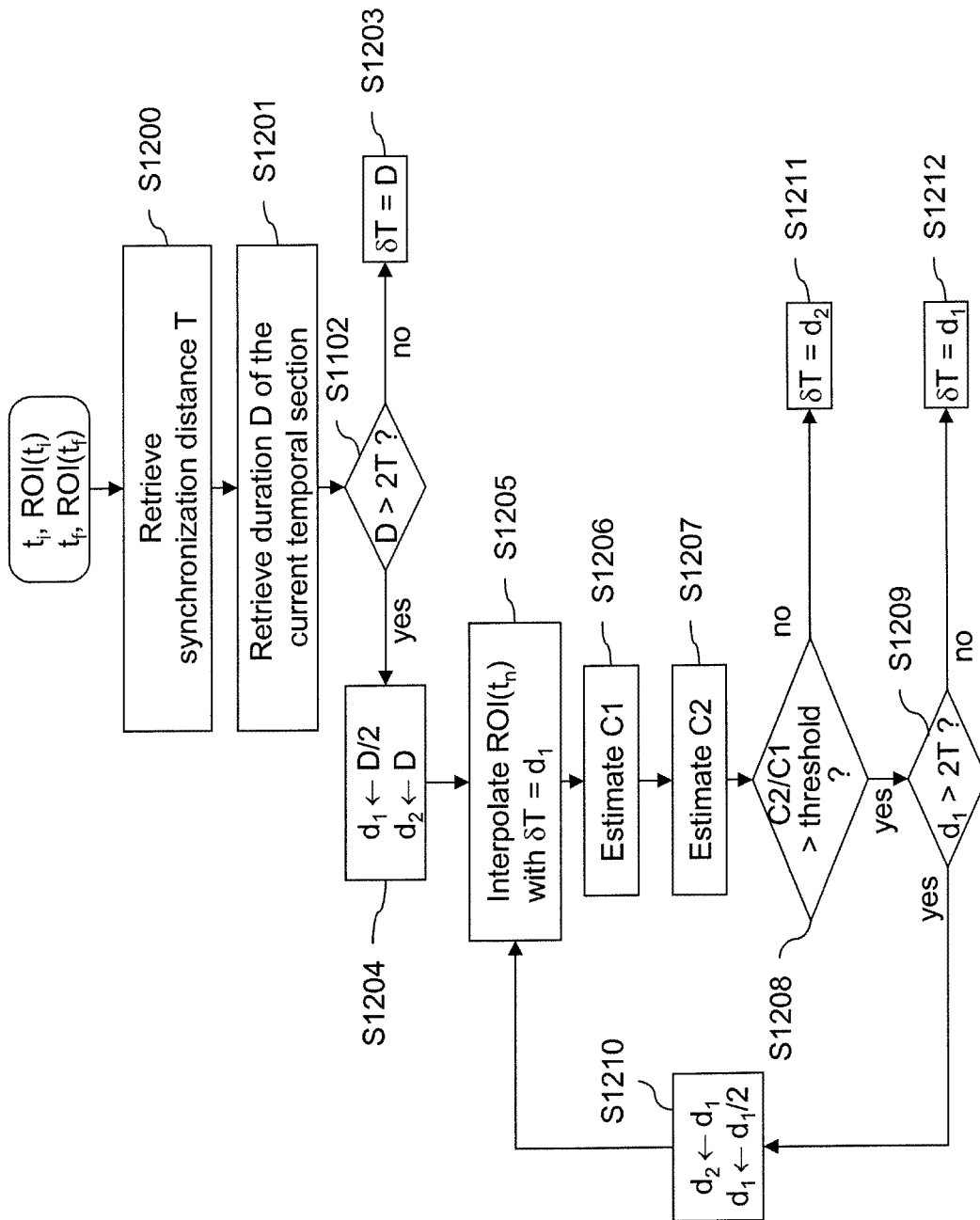


Figure 12

9/10

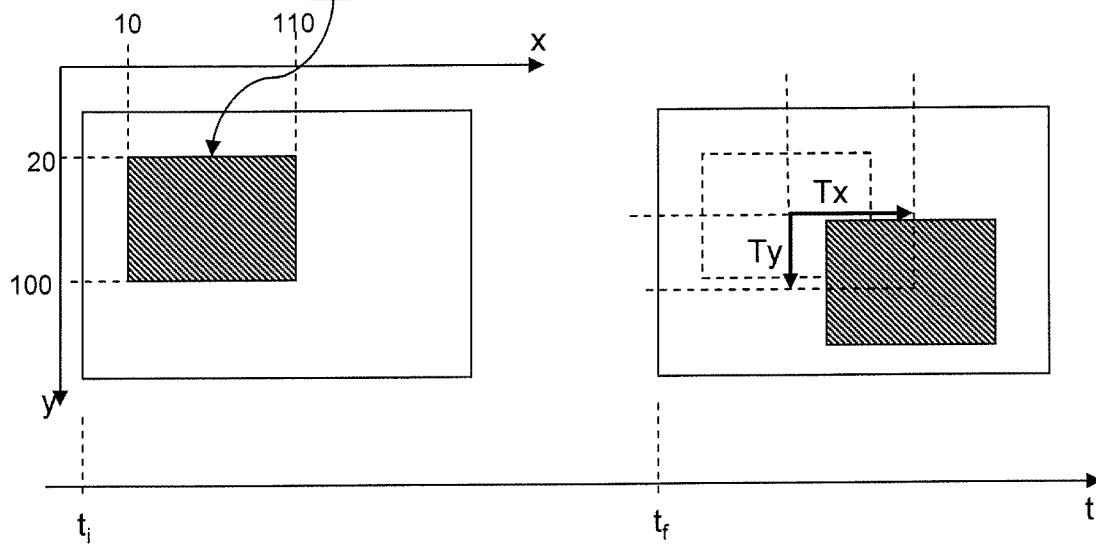
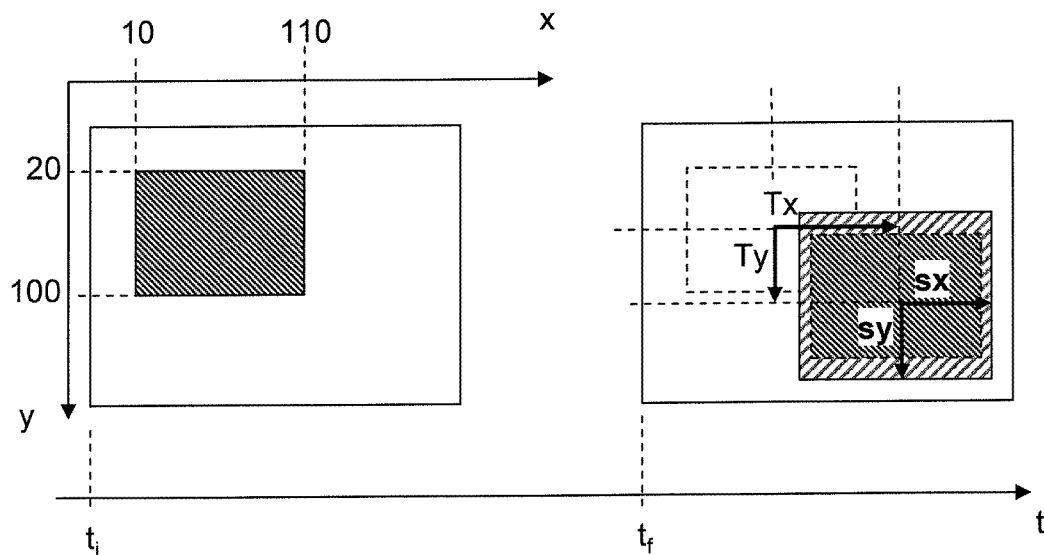
*Example 1:**URI#xywh=10, 20, 100, 80 & T=Tx, Ty & t=t_i, t_f**Example 2:**URI#xywh=10,20,100,80 & translate=(Tx, Ty) & scale=(sx, sy) & t=t_i, t_f**Example 3:**URI#xywh=10,20,100,80 & A=sx, 0, 0, sy, Tx, Ty & t=t_i, t_f & granularity=high*

Figure 14

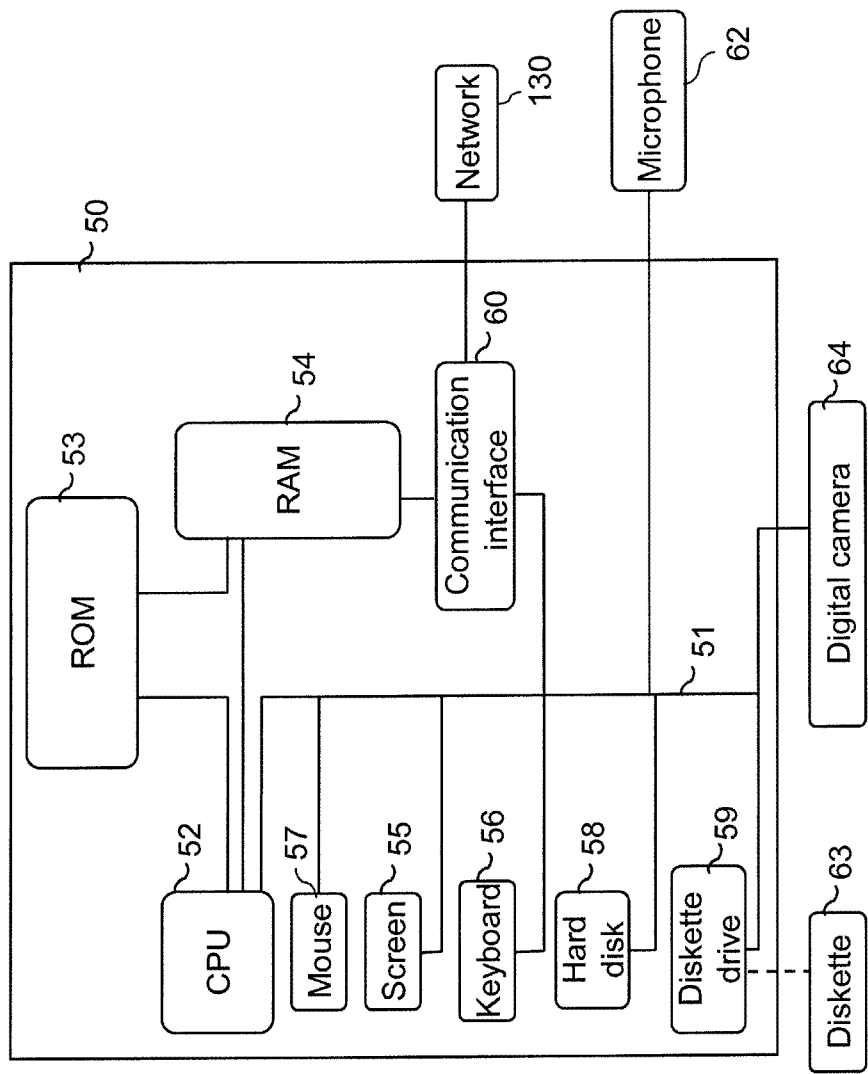


Figure 15

INTERNATIONAL SEARCH REPORT

International application No
PCT/IB2011/051528

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04N7/173
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04N

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2009/300701 A1 (KARAOGUZ JEYHAN [US] ET AL) 3 December 2009 (2009-12-03) abstract figures 1,6 paragraphs [0024] - [0029], [0039] - [0049]	1-22
X	US 2002/021353 A1 (DENIES MARK [US]) 21 February 2002 (2002-02-21) abstract figure 4 paragraphs [0020] - [0032], [0039] - [0053], [0056] - [0059] ----- -/-	1-22



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search

7 July 2011

Date of mailing of the international search report

15/07/2011

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Dobbelaere, Dirk

INTERNATIONAL SEARCH REPORT

International application No

PCT/IB2011/051528

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2006/062478 A1 (CETIN AHMET E [TR] ET AL) 23 March 2006 (2006-03-23) abstract figures 1-5 paragraphs [0007] - [0009], [0014] - [0018], [0039] - [0041], [0056] - [0067] -----	1-22
A	EP 1 710 719 A1 (RICOH KK [JP]) 11 October 2006 (2006-10-11) abstract paragraphs [0006] - [0016], [0031] - [0033], [0042] - [0062] figures 1-3,5-13 -----	1-22
A	ISLAM A ET AL: "JPEG2000 for Wireless Applications", PROCEEDINGS OF THE INTERNATIONAL SOCIETY FOR OPTICAL ENGINEERING (SPIE), SPIE, USA, vol. 5203, 1 January 2003 (2003-01-01), pages 255-271, XP002316046, ISSN: 0277-786X, DOI: DOI:10.1117/12.512536 pp 262-265: Part II: Region-Of-Interest Coding in Motion JPEG2000 -----	1-22

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/IB2011/051528

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2009300701	A1	03-12-2009	NONE
US 2002021353	A1	21-02-2002	NONE
US 2006062478	A1	23-03-2006	NONE
EP 1710719	A1	11-10-2006	JP 2006295386 A 26-10-2006 US 2006228033 A1 12-10-2006