MAIWALD · Elisenstrasse 3 · 80335 München

Europäisches Patentamt

80298 München

17 July 2020

lts- und
lschaft mbH

Dusseldorf

Elisenhof, Elisenstrasse 3
80335 München
T +49 89 7472660
F +49 89 776424
E
H www.maiwald.eu

Official Ref.: EP 09 793 127.3 / 2 347 591
Patentee: Velos Media International Limited
Opponent: Unified Patents, LLC

Our Ref.: U07908EPOP/SQL

Dr. Simon Quartus Lud
Partner
lud@maiwald.eu

Grounds of Opposition

This opposition is filed against grant of European Patent EP 2 347 591 B1 titled *"VIDEO CODING WITH LARGE MACROBLOCKS"* (in German: *"VIDEOCODIERUNG MIT GROSSEN MAKROBLOCKEN"*) in the name and on behalf of "the Opponent",

UNIFIED PATENTS, LLC.
1875 Connecticut Ave., NW, Floor 10
Washington, D.C. 20009

Table of contents

A. Introduction and Formalities

1 The Opponent hereby opposes European Patent EP 2 347 591 B1 (hereinafter "the Patent") granted on 08 April 2020 to Velos Media International Limited Dublin 18, (hereinafter "the Patentee").

1. Requests

2 It is hereby requested that the Patent be revoked **in its entirety** under Article 99 EPC on the grounds of:

(a) Article 100(a) EPC because the subject matter of the Patent lacks novelty contrary to Articles 52(1), 54 EPC and/or lacks inventive step contrary to Articles 52(1), 56 EPC; and

(b) Article 100(b) EPC because the subject matter of the Patent lacks sufficiency of disclosure contrary to Articles 83 EPC; and

(c) Article 100(c) EPC because the subject matter of the Patent extends beyond the content of the European Application as originally filed contrary to Article 123(2) EPC.

3 In the event that the Opposition Division is not in a position to revoke the Patent in its entirety (see request (1) above), Oral Proceedings pursuant to Article 116 EPC are hereby requested.

2. The Prior Art

4 In these grounds of opposition, Opponent relies upon the following documents:

Ref.	*Publication No.*	*Publication date*
D1	U.S. Patent 9,930,365 ("'365 Patent")	
D2	File History for '365 Patent ("'365 File History")	
D3	File History for U.S. Patent 8,503,527 ("Pat. 8,503,527 File History")	
D4	File History for U.S. Patent 8,948,258 ("Pat. 8,948,258 File History")	

)

D5 File History for U.S. Patent 9,788,015 (“Pat. 9,788,015 File History”)

D6 U.S. Patent 5,999,655 to Kalker et al. (“Kalker”)

D7 U.S. Pub. 2005/0123282 to Novotny et al. (“Novotny”)

D8 U.S. Patent 6,084,908 to Chiang et al. (“Chiang”)

D9 Declaration of Dr. Immanuel Freedman (“Freedman Decl.”)

D10 Curriculum Vitae of Immanuel Freedman, Ph.D.

D11 Iain E. G. Richardson, H.264 and MPEG-4 Video Compression, John Wiley & Sons Ltd. (2003) (“Richardson”)

D12 Peter Symes, Video Compression Demystified, McGraw-Hill (2001) (“Symes”)

D13 ITU-T Recommendation H.264, International Telecommunication Union (Nov. 2007) (“ITU H.264”)

D14 U.S. Pub. 2006/0002464 to Au et al. (“Au”)

D15 Int’l Pub. No. WO 2005/038603 to Woods et al. (published Apr. 28, 2005) (“Woods”)

D16 U.S. Pat. 6,233,017 to Chadda (filed Jun. 30, 1997) (“Chadda”)

D17 U.S. Pat. 6,778,709 to Taubman et al. (field Mar. 12, 1999) (“Taubman”)

D18 Matthew Drake et al., MPEG-2 Decoding in a Stream Programming Language, Proceedings 20th IEEE International Parallel & Distributed Processing Symposium (Apr. 2006)

D19 ITU-T Recommendation H.262, International Telecommunication Union (Feb. 2000) (“ITU H.262”)

D20 U.S. Pub. 2007/0074265 to Bennett et al. (published Mar. 29, 2007) (“Bennett”)

D21 WO 2008 027 192 A2

D22 US 5 107 345 A

D23 US 6 660 836 B1

[5] The list of references is annexed hereto as

Annex M1.

)

6 Each of the documents D6 to D23 listed above was published before the earliest priority date of the Patent, which is 03 October 2008. Therefore, each of cited documents D6 to D23 constitutes prior art under Article 54(2) EPC, thus relevant for the assessment of both lack of novelty and inventive step.

7 Since the vast majority of the cited documents was available to the public prior to the claimed priority date, the issue of entitlement to priority is not discussed in detail herein below. However, this is no admission that the claimed subject matter would validly claim the priority.

8 References to the cited prior art documents are to paragraphs or sections of the publications listed above. These references, however, are exemplary and not exhaustive. Opponent reserves the right to refer to other sections of the prior art documents if necessary later in the proceedings.

B. European Patent EP 2 347 591 B1

1. Formalities

9 The Patent, EP 2 347 591, derives from European Patent Application Number 09 793 127.3 (hereinafter “the European Application”) which has a filing date of 29 September 2009.

10 The European Application was published as EP 2 347 591 A2 on 22 July 2011. The Patent claims the priority date of 18 September 2009, which is the filing date of U.S.-American Patent Application US20090562504 (hereinafter “the Priority Application”).

11 In the following, references to the Patent and paragraphs thereof (shown in square brackets […]) relate to the published B1-specification, unless specified otherwise.

2. Subject matter of the alleged invention

12 The alleged invention generally concerns systems and methods for encoding and decoding video data for “large” block sizes, i.e., block sizes larger than 16x16 pixels, and in particular multiples thereof.

13 However, the concept of employing video blocks larger than 16x16 pixels was already practiced years before the time of the Patent. The claims of the Patent were allowed during prosecution based on the requirement of using two distinct syntax elements representative of a minimum and maximum size for blocks in a sequence of pictures.

14 However, there is no suggestion in the specification that the concept of using minimum and maximum syntax elements was a novel concept, or that the use of syntax elements indicating minimum and maximum block sizes was somehow necessary or even beneficial to enabling the use of larger block sizes in video coding.

15 In any case, the use of syntax elements representative of the minimum and maximum block sizes in a coding unit was known long before the time of the Patent, as shown by both Kalker (D6) and Novotny (D7), discussed in more detail below. The Challenged Claims are therefore obvious over the prior art cited herein and should be found unpatentable.

3. Technological Background

16 Digital video is formed from a sequence of video frames that include picture element (or pixel) data. *See Freedman Decl.* (D9) at ¶34 (citing Richardson, D11).

17 During playback, the frames are successively displayed at a certain rate, rendering the video for display. *Id*. The rate at which successive frames are displayed should be high enough such that the transition from frame to frame is imperceptible to the human eye. *Id*.

18 Each frame is an array of pixels organized in rows and columns to form the image represented by the frame, which reflect characteristics of objects represented in a scene of a video.

19 Video files can be large due to the large amounts of image data associated with each frame. *Id*. at ¶35. Therefore, video coding techniques are used to compress (i.e. encode) video files for efficient transmission for receipt and decompression (i.e., decoding) and output at an end-user display device. *Id*. Such compression is achieved by removing redundancy in and between frames. *Id*. at ¶36.

20 Specifically, within a particular sequence of video images, individual frames can be correlated to benefit from redundant video information from within a given frame (spatial correlation) and from successive frames captured at around the same time (temporal correlation):

21

Figure 3.2 Spatial and temporal correlation in a video sequence

22 *Richardson* (D11) at 53, Fig. 3.2.

23 Many aspects of video coding were well-known long before the Patent, including block-based video coding employing prediction techniques to remove spatial and temporal redundancy in coded video data. *See* the Patent at ¶5.

24 To do so, video coders would use four processes (*inter alia*), discussed below: (1) partitioning frames into different sections, such as slices, macroblocks, and sub-blocks, (2) removing redundancy by identifying

predicted blocks and their respective reference block or frame, (3) removing residual data that contains unimportant visual information using transform operations and quantization, and (4) encoding the reduced amount of data using various techniques, such as Huffman coding and/or variable-length coding (describing frequent events with shorter code-words than those used for less frequent events).

25 In block-based video coding, such as the H.264 standard mentioned in the background of the Patent at ¶4, each video frame is partitioned into macroblocks containing a certain number of pixels, and each macroblock could be further partitioned into sub-blocks or blocks. *Id*.; *see also Freedman Decl.* (D9) at ¶37.

26 For original blocks of data, a prediction technique or mode generates a corresponding block elsewhere in the video frame or sequence. *Id*. As acknowledged in the background of the Patent (and illustrated in Fig. 3.2 above), it was well-known to generate predicted blocks using **intra-prediction** (spatial prediction) and **inter-prediction** (temporal prediction). *See the Patent* at ¶5.

27 **Intra-prediction** involves generating a predicted block using similarities that exist between an original block and other blocks within the **same frame**. *Id*.; *see also Freedman Decl.* (D9) at ¶¶37-38.

28 **Inter-prediction** involves generating a predicted block using similarities existing between neighboring blocks in the same frame or temporal prediction with respect to corresponding blocks in **other frames** in the video sequence. *See Freedman Decl.* (D9) at ¶¶37-38.

29 The block is next subject to mathematical transform operations, such as a discrete cosine transform, that converts frame pixel data from the spatial domain into a frequency domain. *Id*. at ¶39.

30 This operation discards the less important visual information within a predicted frame. *Id*. These transformed pixel values are referred to as “transform coefficients.” *Id*. The coefficients may be further compressed via an irreversible process called “quantization,” whereby a matrix of transform coefficients are divided by a corresponding quantization value and the resulting coefficient is rounded.

31 Quantized blocks containing all zero values are often referred to as “skipped” or “zero” blocks and encoded with very few bits to indicate that the predicted block is rendered as identical to the reference block.

32 Tables 3.9-11 of *Richardson*, D11, illustrate this concept. Table 3.9 shows residual data for an 8×8 block of pixels:

Table 3.9 Residual luminance samples (upper-right 8 × 8 block)

−4	−4	−1	0	1	1	0	−2
1	2	3	2	−1	−3	−6	−3
6	6	4	−4	−9	−5	−6	−5
10	8	−1	−4	−6	−1	2	4
7	9	−5	−9	−3	0	8	13
0	3	−9	−12	−8	−9	−4	1
−1	4	−9	−13	−8	−16	−18	−13
14	13	−1	−6	3	−5	−12	−7

33

34 Table 3.10 shows the same residual data with a DCT transform operation assigned to it to transform the pixels into the frequency domain:

35

Table 3.10 DCT coefficients

−13.50	20.47	20.20	2.14	−0.50	−10.48	−3.50	−0.62
10.93	−11.58	−10.29	−5.17	−2.96	10.44	4.96	−1.26
−8.75	9.22	−17.19	2.26	3.83	−2.45	1.77	1.89
−7.10	−17.54	1.24	−0.91	0.47	−0.37	−3.55	0.88
19.00	−7.20	4.08	5.31	0.50	0.18	−0.61	0.40
−13.06	3.12	−2.04	−0.17	−1.19	1.57	−0.08	−0.51
1.73	−0.69	1.77	0.78	−1.86	1.47	1.19	0.42
−1.99	−0.05	1.24	−0.48	−1.86	−1.17	−0.21	0.92

36 And Table 3.11 shows this transformed data quantized by rounding the result of the coefficients divided by some quantization step size or parameter, which was 12 in this example:

Table 3.11 Quantized coefficients

−1	2	2	0	0	−1	0	0
1	−1	−1	0	0	1	0	0
−1	1	−1	0	0	0	0	0
−1	−1	0	0	0	0	0	0
2	−1	0	0	0	0	0	0
−1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

37

38 *See Freedman Decl.* (D9) at ¶39 (citing *Richardson* (D11)).

39 After the video data is encoded, it is stored or transmitted to a receiver for eventual decoding and display to a user. Decoders generally reverse the coding process performed by the corresponding encoder. *Id*. at ¶43.

40 As acknowledged in the specification of the Patent, and argued by the applicant during prosecution of a parent patent, a person skilled in the art would have recognized that the decoding side of the video codec simply performs a "decoding pass generally reciprocal to the encoding pass." *See the Patent* at ¶89 and ¶102.

41 *See Pat. 9,788,015 File History* (D5) at 3471 ("[O]ne of ordinary skill in the art would certainly appreciate that any data encoded by a video encoder must necessarily be decoded by a video decoder."); *see also Freedman Decl.* (D9) at ¶¶43-44.

42 For example, decoders contain an inverse quantizer and transformer for reversing the transformation/quantization phase of the compression process. The inverse quantizer cannot perfectly reverse the quantization process performed by the encoder due to the rounding step; instead, it re-scales, or multiplies, the rounded coefficients by some value, such as the quantization parameter. *Freedman Decl.* (D9) at ¶44. The re-scaled coefficients are then subject to inverse transformation operations to reverse the DCT process. *Id*.

43 Macroblock Partitioning

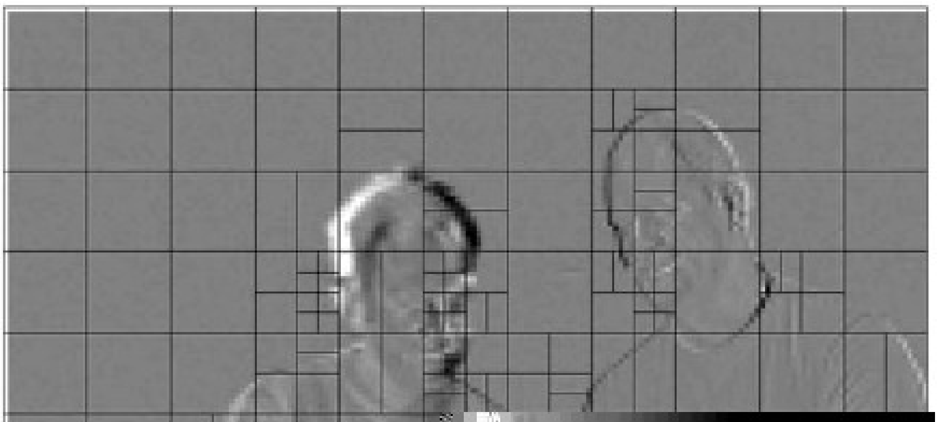
44 In the partitioning phase, the chosen partition size (e.g., 4×4 as compared to 32×32) involves a trade-off between the quality of the image and the quantity of data needed to represent the sequence of

images. *See Freedman Decl.* (D9) at ¶40. Frames made up of larger blocks require less data and, therefore the *rate* for transmission and decoding of such frames is higher; however, frames made up of smaller blocks, while more complex to encode, are more likely to account for anomalies and therefore contain less *distortion*. *Id*. To determine the best block size to use, many video encoders employ Lagrangian optimization functions that attempt to minimize distortion at a desired bit rate. *Id*.

45 Further, by 2008, video encoders were not limited to a single macroblock size per picture frame. Instead, it was conventional to partition frames into macroblocks and sub-blocks of varying size in a "tree" structure arranged in blocks of N×N, N×N/2, or N/2×N pixels, where *N* is an integer that is a power of two (e.g., 4, 8, 16, 32, 64, 128, 256). *See Freedman Decl.* (D9) at ¶41.

46 The example figure below shows a residual frame with different block sizes superimposed – the largest blocks in this example are 16×16 pixels, covering background areas where there is a significant amount of redundancy (i.e., relatively less color variation):

47



[redacted]

[redacted]

[redacted]

48 *See id*. (*Richardson* (D11)). Although the H.264 and H.262 coding standards assumed a 16×16 macroblock size, it was conventional in various encoding systems to employ starting macroblocks of greater size. For example, *Chiang* (D8), discussed in more detail below, assumes an initial block size of 256×256 pixels. *See Chiang* (D8) at 5:43-60; *see also Freedman Decl.* (D9) at ¶¶41-42.

4. Description of the Alleged Invention of the Patent

49 The Patent is directed to techniques for encoding digital video data using large macroblocks, i.e., macroblocks larger than a 16x16 array of pixels. Id. at ¶ 7, summary.

50 As mentioned, one benefit of encoding video frames using larger macroblocks is that a higher compression efficiency can be achieved, particularly in video data generated with higher spatial resolutions and frame rates (i.e., the number of frames displayed in a given unit of time). See id. at ¶39. The Patent notes that while a large macroblock generally refers to initial macroblocks of greater than 16×16 pixels, "large" macroblock includes a conventional 16×16 block, depending on the video resolution and frame rate. Id. at 7:43-57; see also id. at ¶38.

51 The claims of the Patent are generally directed to the decoding of encoded video data employing two syntax elements. In the Patent, a syntax element is used to describe information that is communicated to and used by the decoder to understand the characteristics or processing of encoded data so that the decoder can determine how to decode the encoded data. See, e.g., the Patent at ¶58; see also Freedman Decl. (D9) at ¶54. Particular claim limitations specify the information contained in particular syntax elements.

52 For example, the Challenged Claims describe a first syntax element representing a maximum size value, the maximum size value indicating the size of the plurality of video blocks in the coded unit and a second syntax element representing a minimum size value, the minimum size value indicating a size of the smallest partition for the coded unit, See id. at Claims 1 and 4.

53 Accordingly, when processing encoded video data that has been partitioned into blocks, if a block of data is equal to the minimum size as indicated by the second syntax element, the decoder understands that the sub-block does not have further partitions. If the sub-block does not

have further partitions, the decoder will decode the block according to the encoding mode as represented by the first syntax element.

54 While the claims require both larger-sized macroblocks and the use of minimum and maximum syntax elements, the specification fails to elaborate on how the use of such syntax elements enables the use of larger macroblocks. Nonetheless, both concepts are found in the prior art as detailed below.

5. The person skilled in the art

55 The person skilled in the art would have been a person having, as of October 3, 2008: (1) at least an undergraduate degree in electrical engineering or closely related scientific field, such as physics, computer engineering, or computer science, or similar advanced post-graduate education in this area; and (2) two or more years of experience with video or image processing systems. See Freedman Decl. (D9) at ¶¶30-32.

6. Granted independent claims

56 The Patent comprises an independent device claim 1 related to *an apparatus* and in total two dependent device claims.

57 Further, the Patent comprises an independent method claim 4 related to *a method* and in total two dependent method claims.

58 Finally, the patent comprises an independent memory product claim 7 related to *a* computer-readable medium referring back to the methods claims.

6.1 Granted claim 1

59 For ease of reference, independent device claim 1 is shown below, divided into separate integers in the form of a feature analysis.

1.1 An apparatus comprising:

1.1a a memory configured to store a coded unit, and

1.1b a processor, in communication with the memory, configured to encode the coded unit comprising a plurality of equally sized, square shaped video blocks of size NxN, 1.2 where N is an integer multiple of 16;

1.3 wherein each of the plurality of video blocks can be partitioned into partitions using hierarchical partitioning with one or more levels,

1.4 wherein, at each hierarchical partitioning level, square partitions can be further partitioned into two equally sized rectangular partitions or into four square partitions,

1.5 and the encoding of each of the plurality of video blocks uses a hierarchical coded block pattern;

1.6 generate syntax information for the coded unit wherein the syntax information includes:

1.7 a first syntax element representing a maximum size value, the maximum size value indicating the size of the plurality of video blocks in the coded unit;

1.8 a second syntax element representing a minimum size value, the minimum size value indicating a size of the smallest partition for the coded unit.

60 Opponent cannot exclude that **feature 1.3** is construed by Patentee to merely define that partitioning results into rectangular or square partitions.

6.2 Granted claim 4

61 Further, independent method claim 4 is shown below, divided into separate integers in the form of a feature analysis.

4.1 A method comprising:

4.1b encoding a coded unit comprising a plurality of equally sized, square shaped video blocks of size NxN;

4.2 where N is an integer multiple of 16;

4.3 wherein each of the plurality of video blocks can be partitioned into partitions using hierarchical partitioning with one or more levels, 4.4 wherein, at each hierarchical partitioning level, square partitions can be further partitioned into two equally sized rectangular partitions or into four square partitions;

4.5 and the encoding of each of the plurality of video blocks uses a hierarchical coded block pattern;

4.6 generating syntax information for the coded unit, wherein the syntax information includes:

4.7 a first syntax element representing a maximum size value, the maximum size value indicating the size of the plurality of video blocks in the coded unit;

4.8 a second syntax element representing a minimum size value, the minimum size value indicating a size of a smallest partition for the coded unit.

62 The feature analysis of both granted claim 1 and claim 4 is enclosed herewith as

Annex M2.

C. Grounds for Opposition

I. Added subject matter – Articles 100 (c), 123 (2) EPC

63 As discussed in the following, at least **features 1.7** and 1.8 of granted claim 1 contains subject matter extending beyond the content of the application as originally filed, contrary to the requirements of Article 123(2) EPC.

64 In response to the Summons to Oral proceedings, Patentee has filed an amended claim 1 with their letter dated 13 August 2018, in which the term *"a first syntax element"* and *"a second syntax element"* have been incorporated into claim 1, as highlighted in the following:

> *"a first syntax element representing a maximum size value, the maximum size value indicating the size of the plurality of video blocks in the coded unit;*
>
> *a second syntax element representing a minimum size value, the minimum size value indicating a size of a smallest partition for the coded unit."*

Patentee argued that basis for this feature can be found in the priority application.

65 Page 26, p186 of the originally filed application, however, describes with reference to a minimum size value solely:

"In some examples, video encoder 20 may include a minimum size value in syntaxinformation for a coded unit. In some examples, the minimum size value indicates the minimum partition size in the coded unit. The minimum partition size, e.g., the smallest block in a coded unit, in this manner may be used to determine a maximum length for the hierarchical coded block pattern."

(emphasis added)

66 In contrast thereto, amended claim 1 recites a specific data tuple is used, a first syntax element and a second syntax element.

67 In other words, granted claim 1 covers embodiments, in which specific data tuples of two different elements are used, whereas the specification as originally filed explicitly states syntax information for a coded unit in general.

68 At least for these reasons, granted claim 1 extends beyond the content of the application as originally filed, contrary to Article 123(2) EPC.

69 Therefore, the Patent is to be revoked in its entirety under Articles 100(c), 123(2) EPC.

II. Lack of inventive step – Articles 100 (a), 56 EPC

70 The subject matter of each of claims 1 to 7 of the Patent does not involve inventive step over numerous prior art references, as discussed in detail in the following.

1. Lack of inventive step over document D6

1.1 Independent claim 1

71 Independent claim 1 is anticipated by D6, at least for the following reasons.

72 To the extent the preambles of **features 1.1, 1.1a, 1.1b, 1.2, 1.3, and 1.4,** are limiting, *Kalker* (D6) teaches, or at least renders obvious the

preambles. *See Kalker* (D6) at 4:36-42; *see also id*. at 5:31-57, 3:8-18, Figs. 1, 5, Claim 8.

73 Further, *Kalker* (D6) teaches a video-receiving station (i.e., a *non-transitory computer-readable storage medium*) that includes a video decoder (i.e., *a device for decoding video data* that performs a *method of decoding video data*):

74 Hence, the video decoder constitutes the preambles of **features 1.1, 1.1a, 1.1b, 1.2, 1.3, and 1.4**. Hierarchical positioning would also be disclosed by cited document D15, FIG. 8 of D15 illustrates the variable block size case, which arises from 5-level hierarchical variable size block matching. See this regard also D16, col. 13, lines 30 et sequentia.

75 *Kalker* (D6) at Fig. 1 (annotated); *see also id*. at Fig. 5; 3:8-18 (describing components of the receiving station), 4:36-47, 5:36-42.

76 The receiving station includes a map decoder circuit 9 (i.e., *a processor* containing *instructions* on a *computer-readable storage medium*) that processes video in accordance with the scanning process of the encoder. *See id*. at 3:8-18; *see also id*. at 4:36-50; *see also id*. at 3:4-7, Fig. 1 (showing storage medium 10 at both the encoder and decoder sides).

77 The map decoder circuit reconstructs encoded video frames using a segmentation map reconstruction circuit, which includes memory for storing video data as it is processed (i.e., *memory configured to store decoded video blocks* that is *in communication with the processor*).

78 *See id*. Further, as a person skilled in the art would have known, a decoding device, such as the receiving station in *Kalker* includes some memory, even if only to temporarily store blocks and frames that have been decoded before being displayed. *See Freedman Decl.* (D9) at ¶51.

79

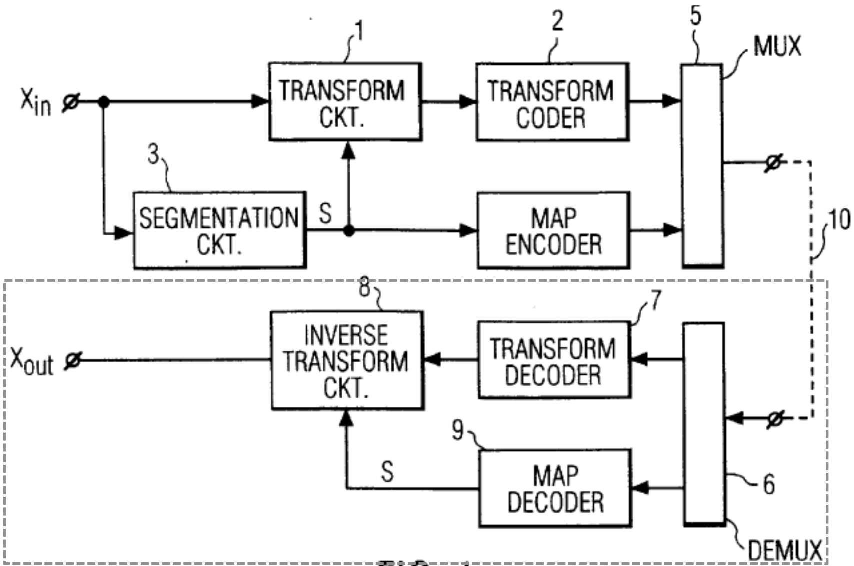


FIG. 1

80 *Kalker* (D6) teaches that the receiving station includes a decoder and segmentation map reconstruction circuit that stores and scans segmented blocks of video data to perform a "reconstruction process" of the compressed video data based on elements describing the scanned blocks. *See id*. at 4:43-67; *see also id*. at 5:31-57 (describing processing blocks on the basis of the largest block size); *see also Freedman Decl.* (D9) at ¶51.

81 Further, like the Patent, *Kalker* (D6) teaches both encoding and decoding video data. Its most detailed discussion is provided from the perspective of the encoding process, while much of the decoding process is generally described with respect to the information and data received from the encoder. *See id*. at 3:8-18 (receiving station includes demultiplexer and decoding components to perform the inverse operations of the coder), 4:36-42 (decoder circuits need not be described in detail given extensive description of coding counterparts); *compare with* the Patent at ¶89 (decoder performs decoding pass "generally reciprocal" to the encoding pass); *see also id*. at ¶13 to ¶16 and at at ¶19 to ¶21 (decoder decodes based on block-type syntax information), Fig. 17.

82 However, as a person skilled in the art would have recognized, *Kalker*'s (D6) teachings of its encoding steps would be reversed by a corresponding decoder device. *See Freedman Decl.* (D9) at ¶52; *see also Kalker* (D6) at 4:48-50 (scanning order performed by decoder corresponds to scanning order in the encoder).

83 Therefore, to the extent limiting, these preambles are taught, or at least rendered obvious by, *Kalker* (D6).

84 The method described in D6 comprises encoding said segmentation map comprises assigning a block-size code to each block size and scanning the segmentation map in accordance with a predetermined scanning pattern to obtain a one-dimensional series of block-size codes. It is thereby achieved that only block-size codes are transmitted for blocks which are not divided into smaller blocks (**feature 1.5**).

85 **Feature 1.6** requires the step of generating syntax information for the coded unit, wherein the syntax information includes: a first and a second syntax element.

86 Novotny (D7) teaches a system in which syntax information may be graphically displayed over corresponding decoded video content. Id. at [0037]; see also id. at Abstract, [0042], [0045], [0003].

87 Novotny (D7) describes different GUIs for displaying the various syntax elements that are transmitted with encoded video and decoded by the decoder for display to an end user.

88 For example, as discussed in more detail below, syntax elements such as macroblock and sub-block types and prediction directions (inter-, intra-) may be encoded with the bitstream displayed to an end-user. See id. at [0050]-[0065].

89 Novotny (D7) is cited below both to complement the teachings of Kalker with respect to limitations taught by Kalker and for its specific teachings regarding the architecture of an encoding/coding system, the types of syntax elements known and used in the art, and for its teaching of using starting macroblock sizes of greater than 16x16 pixels.

90 **Feature 1.7** requires that a first syntax element representing a maximum size value, the maximum size value indicating the size of the plurality of video blocks in the coded unit. This limitation is obvious over Kalker (D6) in view of Novotny (D7). See also in this regard D17, col. 5, lines 40 et sequentia, which clearly discloses to define a typical maximum dimension for sub-blocks.

91 Cited document *Kalker* (D6) teaches providing a block-size code of "S=3," that is set to represent the actual maximum size of the blocks in the coded unit (i.e., second syntax element…associated with the sequence of pictures). This S=3 code is separate from the block size code "S=1" representing the smallest block size (i.e., the first syntax element).

92 The maximum block-size code (S=3) can be set to represent, for example, 16x16 blocks in a given coded unit, which is then scanned on the basis of a 16x16 grid size:

93 In a further embodiment of the scanning circuit 41 (See FIG. 2), the segmentation map is scanned on the basis of the largest block size. If a block comprises smaller blocks, it is scanned on the basis of the next smaller block size. This is an iterative process.

94 FIG. 9 shows a segmentation map illustrating this embodiment. The scanning pattern is denoted 91 in this Figure. First, the top left 16x16 block is analyzed. As this block is not further divided into smaller blocks, the block size code S=3 is generated. Then, the next (top right) 16x16 block is analyzed.

95 See cited document *Kalker* (D6) at 5:31-41, Fig. 9 (annotated to highlight the largest blocks, represented by S=3).

96 As with the preceding limitation, cited document *Kalker* (D6) teaches that the largest block size represented by S=3 can vary in size from one coded unit to another, with another provided example being a maximum block size of 8x8. See id. at 5:15-20; see also id. at 1:50-54. Thus, the encoder defines the value represented by the block-size code S=3 in terms of an actual maximum size of blocks for a given coded unit. See

id. at Abstract, 1:7-13, 1:20-21, claim 1. Freedman Decl. (D9) at ¶56.

97 Once the maximum block size is established, according to the second embodiment of Fig. 9, the scanning grid used in encoding and decoding corresponds to that maximum block size, and that value is represented by the syntax element S=3. See id.; see also id. at 4:48-50.

98 Kalker is able to use the knowledge that block size code S=3 represents the largest block size to enhance coding and decoding efficiency. See Freedman Decl. (D9) at ¶56.

99 Upon learning that a block is equal to the maximum block size (which corresponds to the grid size), the encoder (and therefore, the decoder) may jump immediately to the next grid location (or the largest block partition) without further scanning.

100 Further, Kalker explains that block-size codes are only generated for a given block size that is not divided into smaller blocks at least once. See id. at 1:50-52 ("[O]nly block-size codes are transmitted for blocks which are not divided into smaller blocks.").

101 Therefore, once the largest block size is determined, the grid size is set to correspond to that block size, which in turn minimizes the required scanning to the extent possible. See Freedman Decl. (D9) at ¶56.

102 The highest "S" value (S=3 in the examples provided) is thus directly representative of the largest, or maximum block size for the entire coded unit and is the basis of the grid for the entire coded unit and ultimately the decoder's image reconstruction process for that entire coded unit. Id.

103 To be clear, the encoder communicates two things to a decoder related to a maximum block size.

104 First, it communicates the largest block size of a given picture, represented by S=3 and equal to a 16×16 block size, so that the decoder can inverse the encoding process on the basis of a grid corresponding to the largest block size and represented by this syntax element. See id. at ¶56.

105 The encoder also communicates when a given block in the picture is an S=3 block, in which case the decoder knows that the given block has no partitions and can jump to the next 16×16 block in the picture, as that is the scanning grid size. See id.

106 It would have been obvious to a person skilled in the art that the teachings in Kalker would have been applicable to systems employing block sizes wherein the maximum size is greater than 16×16 pixels.

107 For example, Novotny (D7), which discloses an encoding system based on MPEG/H.264, teaches that the maximum size block stored in memory and output to a decoder may be larger than a 16x16 block.

108 See cited document *Novotny* (D7) at [0031] (providing an example of 32x32 pixel starting block); see also id. at [0030], [0037] (other size macroblocks may be implemented to meet the design criteria of an application).

109 Although 16x16 is an exemplary maximum in Kalker, a person skilled in the art would have recognized that encoding/decoding systems were not limited to 16x16 macroblock sizes. It would have been obvious to incorporate Novotny's teachings of using block sizes larger than 16x16 pixels into the similar system of Kalker.

110 Like Novotny, Kalker contemplates the use of MPEG-like coding methods. See, e.g., Kalker (D6) at Abstract, 2:4:35-42, Figs. 2-5; see also Novotny (D7) at [0002]-[0003], [0025]. Such block sizes are also discussed on page 18 of D19.

111 Further, a person skilled in the art would have recognized (a) that the use of 16x16 in Kalker is merely exemplary and (b) that the use of larger block sizes would be desirable to efficiently encode sequences of images where little variance occurs across pictures.

112 See Freedman Decl. (D9) at ¶¶40-42, 63-64; see also Kalker at 3:30-34 (providing the 16*16, 8*8, and 4*4 block sizes in "the present example"); 4:43-47 (describing an embodiment whereby the decoder performs reconstruction starting from the "smallest" size, where 4*4 blocks are used "in the present example"); 1:52-54 (noting that a "few" different block sizes are used).

113 A person skilled in the art would have been motivated to use larger blocks, as taught in Novotny, in Kalker's system because the use of larger blocks as a starting block size would have provided for higher compression efficiency, while small blocks require many bits. See Freedman Decl. (D9) at ¶¶40-42, 64; see also Kalker (D6) at 1:31-33 ("Small blocks have a plurality of higher level nodes and thus require many bits.").

114 By using a larger initial block size, a programmer could allow a video codec system to use very little data for areas of simple images with large redundancy, such as background elements of a scene, while not compromising the encoder's ability to partition the block further for scenes with greater variance. See id. at ¶64.

115 Further, a person skilled in the art would have had a high expectation of success in implementing these concepts into the system of Kalker, as it would have involved (1) increasing the minimum block size if only three block sizes were to be used, or (2) increasing the number of available block sizes (or "levels" or "modes" of partitioning). See id.

116 A person skilled in the art would have recognized that such modifications would still accomplish Kalker's desired bit savings because a system designer would only use a larger block size if such resulted in less encoded data being sent due to the use of the larger block size. See id.

117 Put another way, the availability of larger block sizes beyond 16×16 in a coded unit would have enhanced the flexibility of Kalker's system to maximize any available efficiency gains that may be had where a particular sequence of images has a high degree of redundancy in a predictable way. See id.

118 A person skilled in the art would have been motivated to gain this flexibility by providing for larger available block sizes than 16□16 and would have had a reasonable expectation of success in doing so in Kalker. Id.

119 As discussed above and for limitation 1[a], Kalker teaches the value for each block-size code may vary from one coded unit to the next. See id. at 1:50-52, 1:20-21, 5:15-21.

120 Because of this variability, a person skilled in the art would have reasonably understood that the encoder would communicate information to the decoder providing the value of each block-size code (including the maximum block-size code, S=3 in the example provided) as such varied for coded unit. See Freedman Decl. (D9) at ¶¶49, 55-57.

121 Although Kalker employs a frame or picture as the coded unit size for its teachings, for the same reasons discussed in limitation 1[a], Kalker's teachings would apply to any conventional coded unit size, including a sequence of pictures or frames, as in claim 1[b]. See id. at ¶58.

122 Thus, Kalker renders obvious the concept of decoding a first syntax element, separate from the second syntax element, associated with the

sequence of pictures, where the first syntax element represents the maximum block size of the sequence of pictures.

123 Therefore, this limitation is obvious over Kalker in view of Novotny.

124 **Feature 1.8** requires that a first syntax element is included in the generated syntax information. As noted above, a person skilled in the art would understand that a syntax element in the general sense is used to describe information that is communicated to and used by the decoder to understand the characteristics or processing of encoded data so that the decoder can determine how to decode the encoded data, as described in the Patent at ¶63 to ¶64. *See also Freedman Decl.* (D9) at ¶¶54-55.

125 In this claim limitation, the syntax element must represent the minimum size of blocks in the sequence of pictures. This limitation is taught by, or at least obvious over, *Kalker* (D6).

126 In particular, *Kalker* (D6) teaches an encoding-side transmitting station that *assigns* particular size values to multiple block-size codes (i.e., *syntax elements*) for an entire coded unit (e.g., a picture or frame *in Kalker* (D6)) and uses these codes to communicate the different block sizes contained in the coded unit; and these block-size codes for the coded unit are then used by the decoding-side receiving station to decode the data and reconstruct the image.

127 *See id*. at 3:8-18; *see also id*. at claim 1 ("[T]he step of encoding said segmentation map comprises **assigning a block-size code to each block size**."); and ("Each picture is segmented into picture blocks, the size of which is adapted to local picture contents."); 3:31-35, 4:55-56 ("If an **element** is not the EOS code, it represents a block size S"), claim 8.

128 For example, the encoder sets a block-size code of "3" to represent the largest blocks in the coded unit, such as 16×16 blocks, a block-size code of "2" can represent intermediary blocks, such as 8×8 blocks, and a block-size code of "1" represents the smallest, or minimum block size in the coded unit, e.g., 4×4 blocks. *See, e.g.*, *id*. at 3:25-34, 5:36-57; *see also id*. at 4:43-67 (in describing the reconstruction process at the decoding unit, referring to block-sized codes as "elements" extracted by the decoding unit).

129 Note that the actual size of the minimum size block in *Kalker* (D6) is variable picture by picture. *See id.* at 1:50-52 ("[O]nly block-size codes are transmitted for blocks which are not divided into smaller blocks," i.e.,

that exist in the picture); *see also id*. at 5:15-21 (providing alternative example of block sizes for a picture); 1:20-21 ("Each picture is segmented into picture blocks, the size of which is adapted to local picture contents.").

130 Therefore, the encoder must scan the entire coded unit (e.g., a picture or frame) and assign the value of S=1 for that coded unit, depending on the actual smallest block sizes in the given coded unit, with a 4×8 block provided as an alternative example in *Kalker* (D6). *See id*. at 5:10-21, Fig. 8; *see also id*. at 1:50-52, 1:20-21.

131 Because the block-size codes vary from one coded unit to the next, the code S=1 does not represent a constant size, but rather must be set in a given coded unit to represent the actual smallest block size in that coded unit. *See Freedman Decl.* (D9) at ¶¶49, 55.

132 Otherwise, the decoder would not be able to understand the block sizes that correspond to the S variable as it relates to each coded unit (e.g., a picture). Put another way, a person skilled in the art would have appreciated that *Kalker*'s (D6) encoder communicates to the decoder that S=1=4×4 and S=3=16×16 for each coded unit for which this is true, as these values for S are not a given for every coded unit. *Id.* at ¶49.

133 Therefore, a person skilled in the art would have reasonably understood that for each coded unit in *Kalker* (D6), the encoder instructs the decoder as to the particular smallest block value that is then characterized by the code S=1 for that particular coded unit. *See id*. at ¶¶49, 54-55, 57.

134 Therefore, the block-size code "S=1" is *a second syntax element representing a minimum size value indicating a size of a smallest partition for the coded unit*.

135 In one example in *Kalker* (D6), the minimum size of blocks in the coded unit being scanned is a 4×4 block, so the encoder sets the "S=1" code to represent a block size of 4×4 for the entire coded unit, or picture. *See id*. at 3:27-32; *see also id*. at 4:43-47, 5:49-52.

136 Then, while scanning a segmentation map, the encoder may assign a block-size code of S=1 to any 4×4 block in the coded unit, or picture, and communicate those block sizes via the S=1 code to the decoder.[1]

[1] To be clear, it is <u>not</u> the fact that an S-code is attached to a particular block to indicate the partitioning size for that block that satisfies this limitation; rather, it is the fact that the value corresponding to the block-size code (S=1) is set based on the actual minimum size of blocks in the coded unit, e.g., a picture.

137 In this example, upon receiving the bitstream for a given coded unit, the decoder will understand that the minimum block size in the coded unit is 4×4 (i.e., because S=1 corresponds to a 4×4 block size). *Kalker*'s decoder then uses the knowledge that S=1 represents the smallest block size to enhance efficiency. *See Freedman Decl.* (D9) at ¶57.

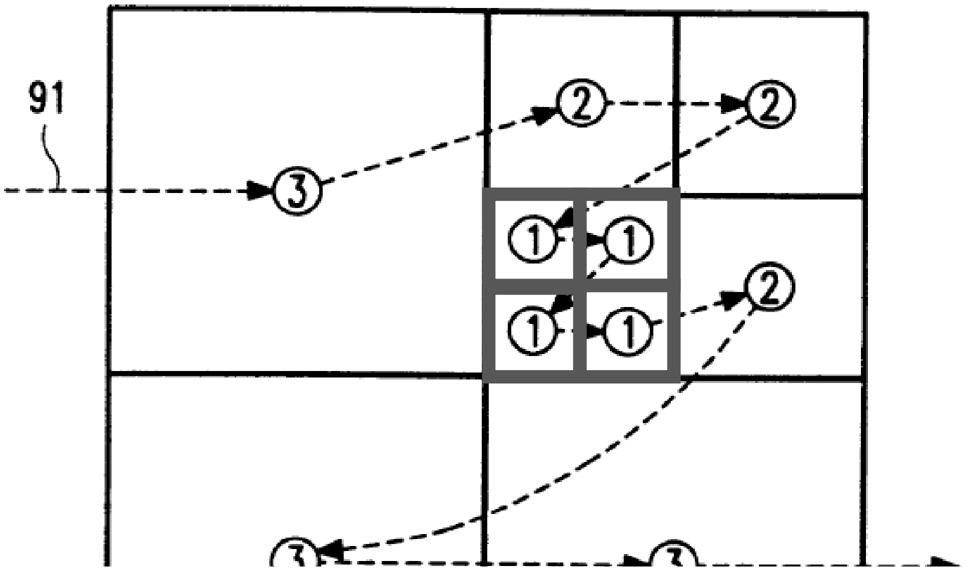
138 Specifically, when an S=1 block is encountered, it is “accordingly” known that any adjacent sub-blocks will also all have a size of “1” and will not contain any further partitions:

139 First, the top left 16*16 block is analyzed. As this block is not further divided into smaller blocks, the block size code S=3 is generated. Then, the next (top right) 16*16 block is analyzed. This block is segmented into smaller blocks and **will now completely be scanned** before proceeding to the next 16*16 block.

140 More particularly, the top left 8*8 block is now analyzed. As it is not further divided, the block size code S=2 is generated. Similarly, the block size code S=2 is generated for the next (top right) 8*8 block. **Then the bottom left 8*8 block is analyzed. It is segmented into smaller blocks and will thus be scanned before proceeding to the next 8*8 block. Accordingly, an S=1 block size code is generated for the top left 4*4 block, the top right 4*4 block, the bottom left 4*4 block and the bottom right 4*4 block, successively**.

141 The scanning then proceeds to the next (bottom right) 8*8 block for which, in this example, the block size code S=2 is produced. Now, the top right 16*16 block has completely been processed and the scanning proceeds to the left bottom 16*16 block (S=3) and the right bottom 16*16 block (S=3).

142



143 *Kalker* (D6) at 5:38-57, Fig. 9 (annotated to highlight the smallest blocks, represented by block-size code S=1); *see also id*. at 4:43-67, 5:1-3, Fig. 7.

144 Further, it would have been obvious to a person skilled in the art that *Kalker*'s use of the S=1 syntax element would be applicable to any conventional coded unit, including a group of pictures, also referred to as a *sequence of pictures* or frames.

145 *See Freedman Decl.* (D9) at ¶58. Although *Kalker* (D6) describes its embodiments in terms of an individual frame or picture (*see, e.g.*, claim 1, 1:20-21), the value of the code S=1 is set for an entire frame containing multiple blocks.

146 It would have required no great exercise of creativity to set the same block-size value indicated by the code S=1 for several consecutive video frames (i.e., *a sequence of pictures*). *See Freedman Decl*. (D9) at ¶¶46, 58.

147 In other words, because several consecutive video frames are likely to involve similarly sized blocks, (e.g., smallest blocks of 4×4 pixels), it would have been obvious to set the code S=1 as representing a 4×4 value for those consecutive frames.

148 A person skilled in the art would have recognized that by applying the same syntax information to a sequence of pictures rather than an individual picture, the system would reduce the amount of overhead data necessary to communicate the encoded syntax elements such as, for example, the value assigned to a given block-size code. *See id*.

149 A person skilled in the art would have would have been motivated to do so because reducing overhead data is a desired goal in video coding generally, including as disclosed in *Kalker*. *See id*. Furthermore, reduced overhead is a predictable result of using a larger coded unit to which syntax data is applied.

150 This is particularly true for a series of consecutive frames in the same scene, which are likely to have highly redundant data and block sizes. *Id*. Therefore, although *Kalker* provides examples of its teachings of assigning block-size codes on a picture-by-picture basis, a person skilled in the art would have would have found it obvious and would have been motivated to apply *Kalker*'s teaching to coding units of larger sizes such as a "sequence of pictures." *See id*.

151 A person skilled in the art would have would have had a reasonable expectation of success in making such a modification because it would

have required nothing more than a minor modification in software (i.e., where to designate the syntax element) to adjust the size of the coded unit using only one of a few available coding-unit sizes for which syntax information were already identified in existing coding standards. *See id*.

152 Furthermore, *Kalker* (D6) is a *video* decoding system that was already operating on a series of pictures that form a video. *See Kalker* (D6) at Abstract ("An advanced video compression coding system which employs variable block size transforms to improve compression efficiency for transmission of <u>*video pictures*</u>."); *see also id*. at 1:7-15.

153 Similarly, the Patent acknowledges that the size of the coding unit is not of particular import, noting that "[a] coded unit may comprise a video frame, a slice, or a group of pictures (also referred to as a "sequence")." See in this regard *the Patent* at ¶184; *see also id*. at ¶66 (listing a group, or sequence, of pictures among multiple "independently decodable unit[s] defined according to applicable coding techniques"); ¶46, ¶68, ¶185.

154 Indeed, in describing the use of the minimum syntax element in the specification, the Patent refers to a non-specific "coded unit," not to a sequence of pictures. *Id*. at ¶189.

155 There is no difference between an individual video frame (picture) or a sequence of pictures as a coded unit in the context of the Patent, and there is no technical difficulty associated with transitioning the system of *Kalker* from an individual frame-based system to one encoding and decoding a sequence of such frames based on the same syntax information. *See Freedman Decl.* (D9) at ¶58.

156 Therefore, this limitation is taught, or at least rendered obvious, by *Kalker* (D6).

157 Therefore, claim 1 lacks inventive step over D6, for multiple reasons as stated above, taken alone, or in combination with D7.

158 Therefore, the Patent is to be revoked in its entirety under Articles 100(a), 54, 56 EPC.

1.1 Independent claim 4

159 Independent claim 4 is anticipated by D6, in corresponding manner at least for the reasons as provided above for new claim 1 since claim 4 corresponds to claim 1 as far as the features 4.1b, 4.2 to 4.8 are

concerned, which clearly correspond to features 1.1b, and 1.2 to 1.8.

160 An intrinsic difference between apparatus claim 1 and method claim 4 is given in feature 1.1 and 4.1, which correspond to an apparatus and a method, respectively.

161 The only further difference between claims 1 and 4 is given in the absence of the structural feature 1.1a in the method claim 4.

162 Thus, it remains to show that also a method is disclosed or at least anticipated by the cited documents.

163 *Kalker* (D6) teaches a method and apparatus for transmission of video pictures which are segmented to form maps of picture blocks of variable block sizes, see claim 1 of *Kalker* (D6).

164 Therefore, claim 4 lacks inventive step over D6, for multiple reasons as stated above, taken alone, or in combination with D7.

1.2 Independent claim 7

165 Claim 7 of the Patent recites as follows:

> "7. A computer-readable medium comprising instructions for causing a programmable processor to perform the methods of any of claims 4-6.

166 *Kalker* (D6) teaches a video-receiving station (i.e., a *non-transitory computer-readable storage medium*). In particular, the description of Fig. 1 of cited document D6 recites as follows, emphasis added:

> *"The encoded transform coefficients and the encoded segmentation map are multiplexed by a multiplexer 5 and applied to a transmission channel or* ***storage medium 10."***

167 Further, it is respectfully submitted that a computer-readable medium comprising instructions for causing a programmable processor to perform the methods as claimed by claim 7 of the Patent would also be common general knowledge for the person skilled in the art, see also D12 and D13 in this context.

168 Further, D18, on page 8, and the abstract, also discusses computer and shared memory within such computer systems. Further, in this regard, see also memory 38 as shown in Fig. 1 of cited document D20.

1.2 Dependent claims

169 All the dependent claims 2 to 3 and 5 to 6 of the Patent are anticipated either by cited document *Kalker* (D6) taken alone or by cited document *Kalker* (D6) in combination with cited document *Novotny* (D7), as discussed in the following.

1.2.1 Claim 2

170 Claim 2 of the Patent recites as follows:

> *"2. The apparatus of claim 1, wherein the syntax information comprises a fixed-length code corresponding to the size of the plurality of video blocks."*

171 As is evident from Fig. 10 of cited document *Novotny* (D7) and the corresponding figure description, e.g. *Id*. at ¶70 et seqq. macroblock size parameter and a quantization parameter may be used as syntax information.

172 Therefore, claim 2 lacks inventive step over cited document *Novotny* (D7) taken alone or in combination of cited document *Kalker* (D6) combined with cited document *Novotny* (D7).

1.2.2 Claim 3

173 Cited document Novotny (D7) also anticipates the subject matter of dependent claim 3.

174 Claim 3 of the Patent recites as follows:

175 *"3. The apparatus of any of claims 1-2, wherein the coded unit comprises one of a frame, a slice, and a group of pictures.*

176 As is evident from Fig. 3 of D7 and the corresponding figure description, e.g. *Id*. at ¶29 et seqq. The coded unit is shown generally illustrating partitions or segments of pictures, *Id*. at ¶30 shows that macroblocks may be grouped in a number of slices, and that macroblocks generally comprise an array of pixels having vertical and horizontal dimensions of equal size.

177 Therefore, claim 3 lacks inventive step over D7 taken alone or in combination of D6 combined with D7.

1.2.3 Claims 5 and 6

178 Already from the discussion of lack of inventive step of the dependent claims, it is evident that also claims 5 and 6 are anticipated by D7 taken alone or in combination of D6 combined with D7.

179 Therefore, claim 5 lacks inventive step over D7 taken alone or in combination of D6 combined with D7. Further, claim 6 lacks inventive step over D7 taken alone or in combination of D6 combined with D7.

2. Lack of inventive step over document D21

2.1 Independent claim 1

180 Independent claim 1 is anticipated by D21, at least for the following reasons.

181 **Features 1.1, 1.1a, 1.1b, 1.3, and 1.4,** are disclosed by D21: p. 16, l. 22-31, in particular l. 24-29, p. 17, l. 32 - p. 18, l. 6, in combination with fig. 5B.

182 **Features 1.5, 1.6 and 1.7,** are disclosed by D21 p 16. l. 32 - p. 17, l. 2, p. 17, l. 11-20, p. 21, l. 21 - p. 22, l. 8, and p. 26, l. 7-28.

183 Further, **Features 1.5, 1.6 and 1.7,** are disclosed by D22, col. 7, l. 53 - col. 8, l. 13, col. 13, l. 4-42, fig 3a, 3b, or also by D23: col. 6, l. 41-45, col. 7, |. 4 col. 8, l. 12, fig. 4A, 4B, 4C.

184 **Feature 1.8** is disclosed by D21 on p. 17, l. 32 - p. 18, l. 6, fig. 5B, p. 28, l. 3-23, fig. 7, table 1.

185 The subject matter of cited document D21 differs from claim 1 of the Patent in that the shaped video blocks have a size of NxN, where N is an integer multiple of 16, basically feature 1.2.

186 Restricting the block size to integer multiple of 16 has advantages in the data processing as the size of 16 is common to most hardware systems. See in this regard pages 136 to 140 of D13. Further also Fig. 1 of D14.

187 The problem resulting from the effect may be given in that coding and decoding architectures need to follow major transformations in order to adapt them to the new initial block size.

188 The person skilled in the art would find a solution to the above stated problem by setting the size of shaped video blocks have a size of NxN, with N relating to multiples of 16.

189 Thus, the subject matter of claim 1 is obvious over and does not involve inventive step over cited document D21.

III. Sufficiency of Disclosure – Articles 100 (b), 83 EPC

190 As discussed in the following, features 1.7 and 1.8 of granted claim 1 of the Patent contain subject matter, which fails to meet the requirements of Art. 83 EPC since the person skilled in the art is not enabled to carry out the invention based on the disclosure of the European patent application documents of the Patent (see also T 487/89, T 297/90, T 541/97).

191 The same applies to features 4.7 and 4.8 of granted claim 4 of the Patent.

192 In particular, features 1.7 and 1.8 of granted claim 1 of the Patent recite as follows:

> *"a first syntax element representing a maximum size value, the maximum size value indicating the size of the plurality of video blocks in the coded unit;*
>
> *a second syntax element representing a minimum size value, the minimum size value indicating a size of a smallest partition for the coded unit."*

193 The Patent is directed to techniques for encoding digital video data using large macroblocks, i.e., macroblocks larger than a 16x16 array of pixels. Id. at ¶7 and ¶8, Id. at claim 1 feature 1.2, *"integer multiple"*.

194 As mentioned, one benefit of encoding video frames using larger macroblocks is that a higher compression efficiency can be achieved, particularly in video data generated with higher spatial resolutions and frame rates (i.e., the number of frames displayed in a given unit of time). See Id. at ¶40.

195 The Patent notes that while a large macroblock generally refers to initial macroblocks of greater than 16×16 pixels, "large" macroblock includes a conventional 16×16 block, depending on the video resolution and frame rate. Id. at 7:43-57; see also Id. at ¶36 and ¶38.

196 The independent claims of the Patent are generally directed to the decoding of encoded video data employing two syntax elements.

197 In the Patent, a syntax element is used to describe information that is communicated to and used by the decoder to understand the characteristics or processing of encoded data so that the decoder can determine how to decode the encoded data. See, e.g., The Patent at ¶58; see also Freedman Decl. (D9) at ¶54.

198 Particular claim limitations specify the information contained in particular syntax elements.

199 For example, the challenged claims 1 and 4 of the Patent describe one syntax element representing a minimum size of blocks in a sequence of pictures and one further syntax element representing a maximum size of blocks in a sequence of pictures, where the maximum is greater than 16×16 pixels. See claims 1 and 4 of the Patent.

200 Accordingly, when processing encoded video data that has been partitioned into blocks, if a block of data is equal to the minimum size as indicated by the first syntax element, the decoder understands that the sub-block does not have further partitions.

201 If the sub-block does not have further partitions, the decoder will decode the block according to the encoding mode as represented by the third syntax element.

202 While the claims require both larger-sized macroblocks and the use of minimum and maximum syntax elements, the specification fails to elaborate on how the use of such syntax elements enables the use of larger macroblocks.

203 In T 409/91 (OJ 1994, 653; ex parte) and T 435/91 (OJ 1995, 188; inter partes) it was pointed out that the protection conferred by a patent should correspond to the technical contribution to the art made by the disclosure of the invention described therein, which excludes the patent monopoly being extended to subject-matter which, after reading the patent specification, would still not be at the disposal of the skilled person.

204 Thus, the claims 1 and 4 requiring both larger-sized macroblocks and the use of minimum and maximum syntax elements are vaguely formulated and leaves several constructions open as possibilities and the the specification fails to elaborate on how the use of such syntax elements enables the use of larger macroblocks.

205 Thus, in summary, the subject matter of each of the claims 1 and 4 of the Patent fails to meet the requirements of Article 100(b) EPC or Article 83 EPC, respectively.

B. Summary and Conclusion

206 It has been shown hereinabove that the claims as granted contain subject matter extending beyond the content of application as originally filed.

207 Therefore, the Patent is to be revoked in its entirety under Articles 100(c), 123(2) EPC.

208 In addition, each of claims 1 to 7 lacks inventive step over numerous prior art documents.

209 Therefore, the Patent is to be revoked in its entirety under Articles 100(a), 54, 56 EPC.

210 In addition, it has been shown hereinabove that the claims as granted claim define an invention, which the person skilled in the art cannot carry out based on the disclosure of the originally filed patent application documents.

211 Therefore, the Patent is to be revoked in its entirety under Articles 100(b), 83 EPC.

Maiwald Patentanwalts- und Rechtsanwaltsgesellschaft mbH
- Association No. 174 -
Electronically signed
Dr. Simon Q. Lud
Patentanwalt | European Patent Attorney

Enclosures
Annex M1 (list of references);
Annex M2 (feature analysis of claims 1 and 4);
Documents D1 to D23