(12) UK Patent (19) GB (11) 2568326 (13) B

(45) Date of B Publication 19.05.2021

(54) Title of the Invention: Video image processing

(51) INT CL: ***H04N 19/17*** (2014.01) ***G06T 9/20*** (2006.01) ***H04N 19/119*** (2014.01) ***H04N 19/167*** (2014.01)

(21) Application No: **1803339.9**

(22) Date of Filing: **01.03.2018**

(30) Priority Data:
(31) **1718488** (32) **08.11.2017** (33) **GB**

(43) Date of A Publication **15.05.2019**

(56) Documents Cited:
GB 2548860 A
WO 2017/176330 A1
US 9185429 B1
JPS6196888
EP 3345125 A1
JP 2012081101 A
US 20030198393 A1

(58) Field of Search:
As for published application 2568326 A viz:
INT CL **G06T, H04N**
Other: **EPODOC, WPI**
updated as appropriate

Additional Fields
Other: **None**

(72) Inventor(s):
Colin Skinner
Matthew Parks

(73) Proprietor(s):
DisplayLink (UK) Limited
22 Cambridge Science Park, Milton Road, Cambridge, CB4 0GH, United Kingdom

(74) Agent and/or Address for Service:
Mathys & Squire LLP
The Shard, 32 London Bridge Street, LONDON, SE1 9SG, United Kingdom

GB 2568326 B

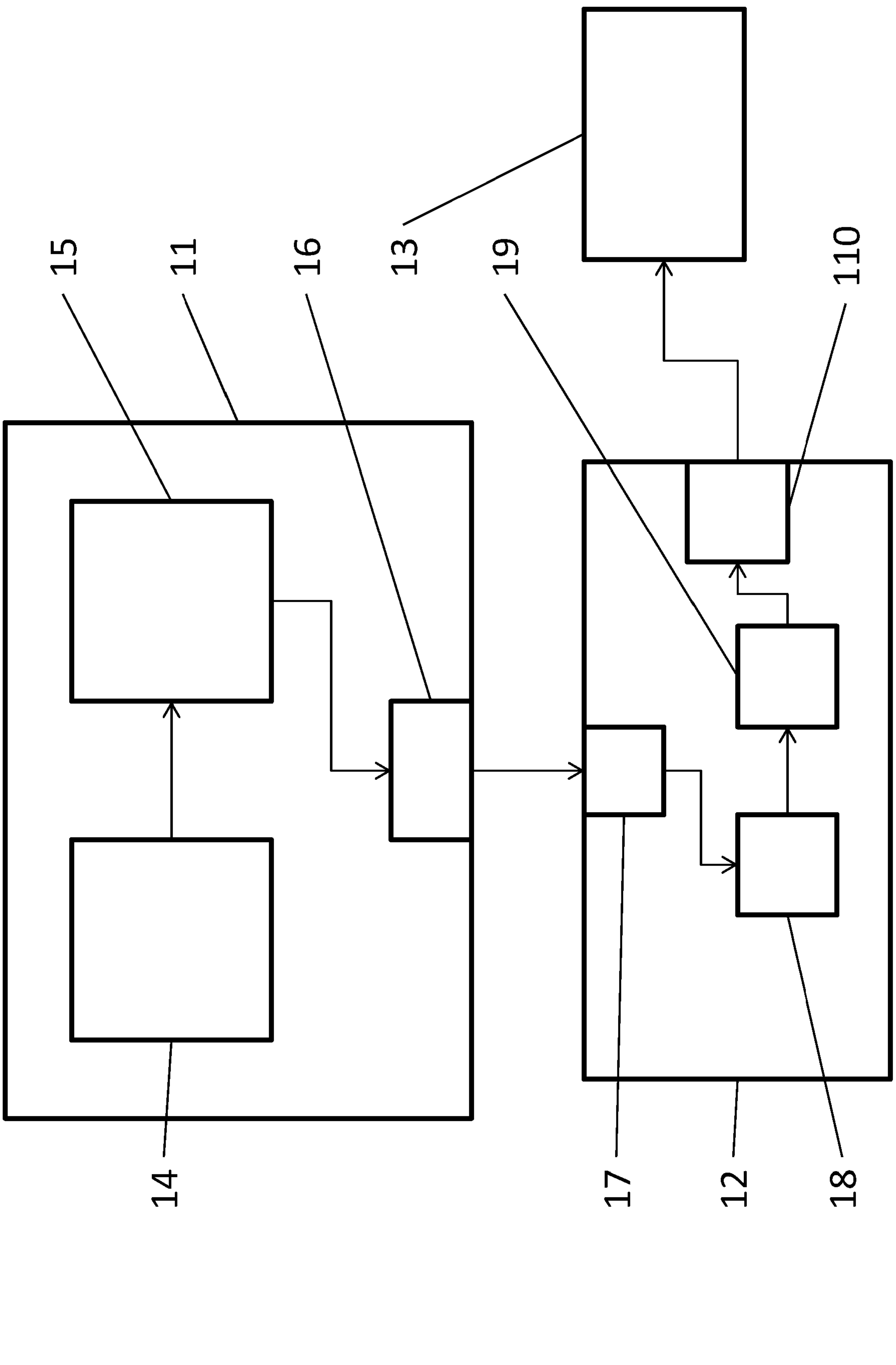

Figure 1

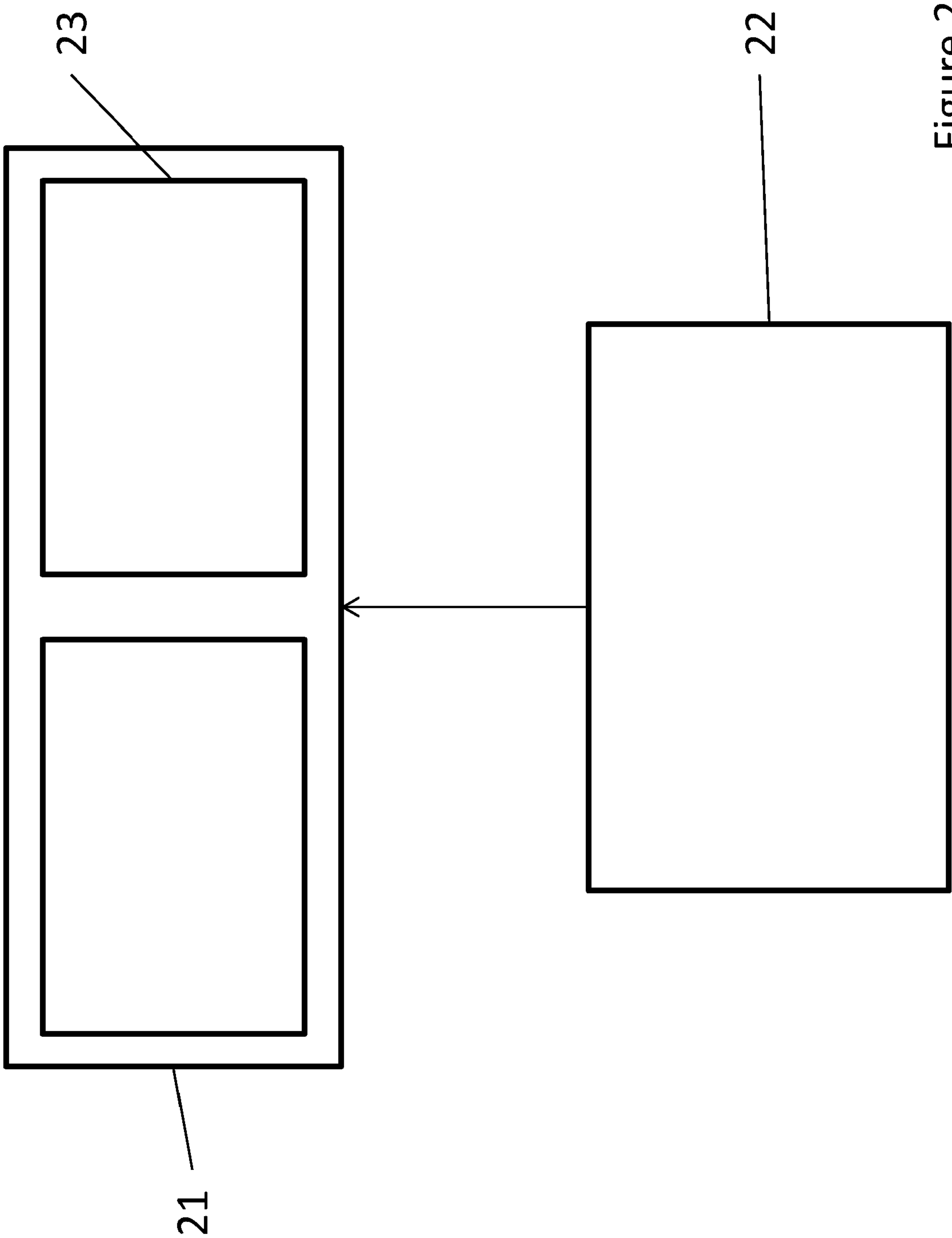

Figure 2A

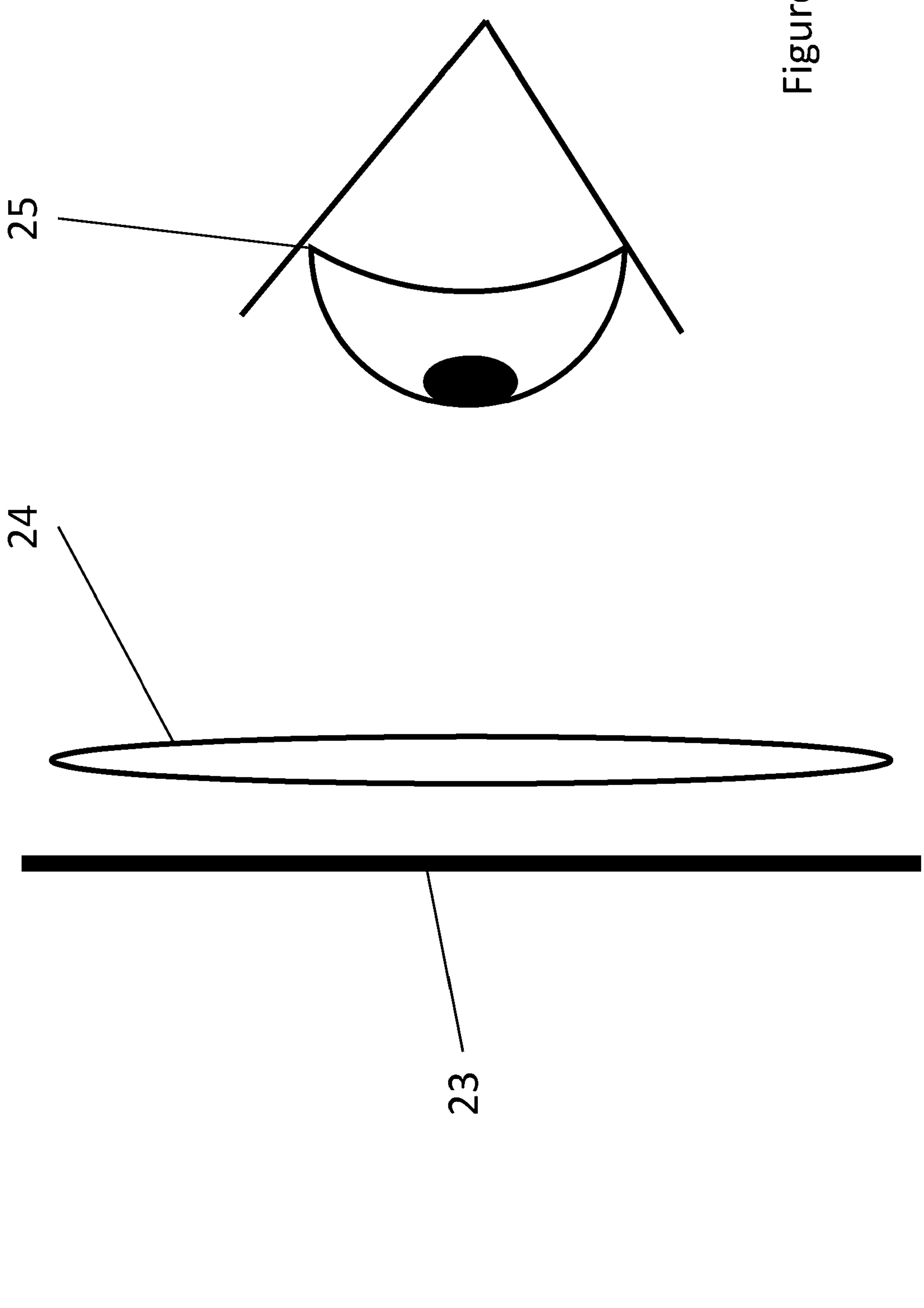

Figure 2B

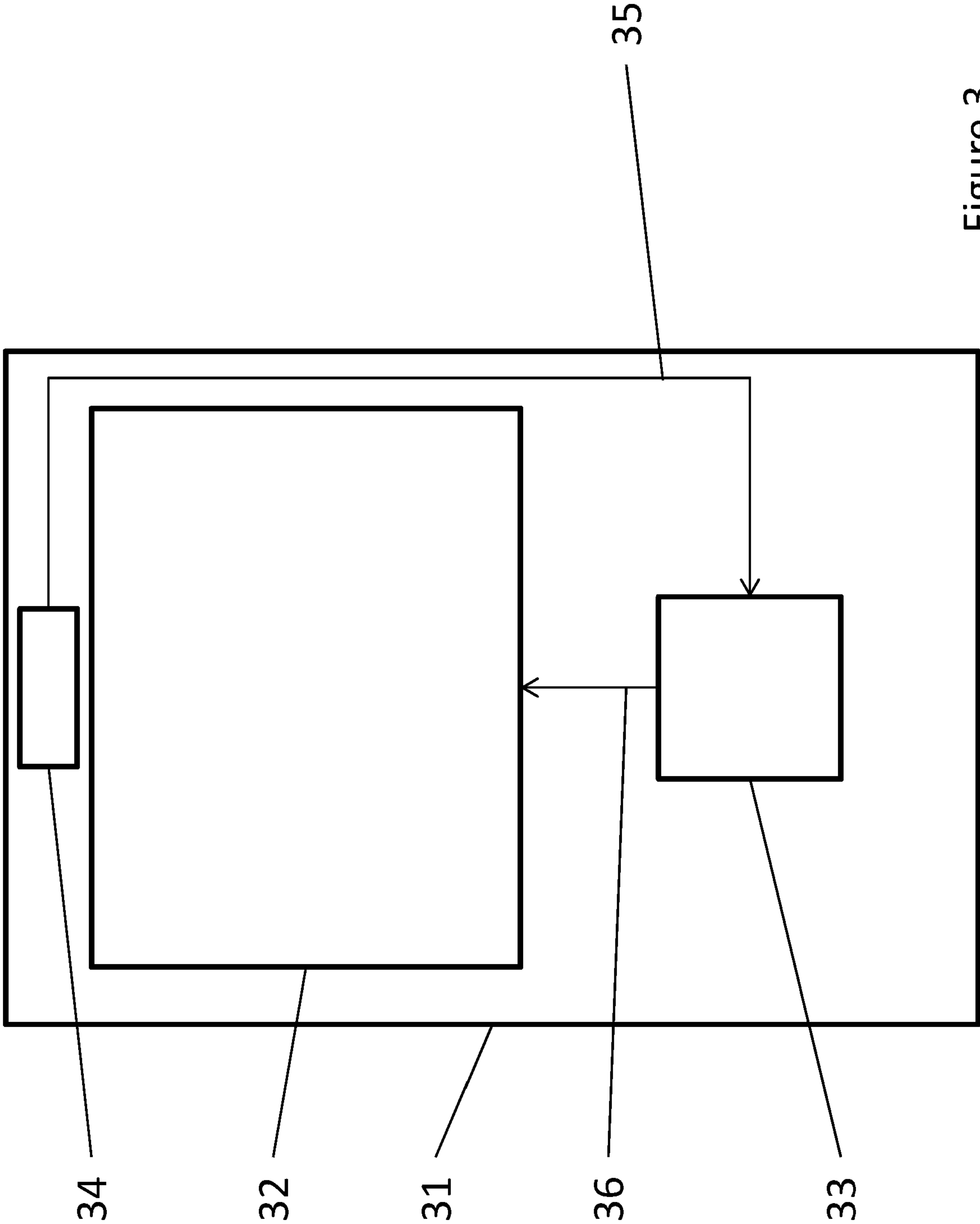

Figure 3

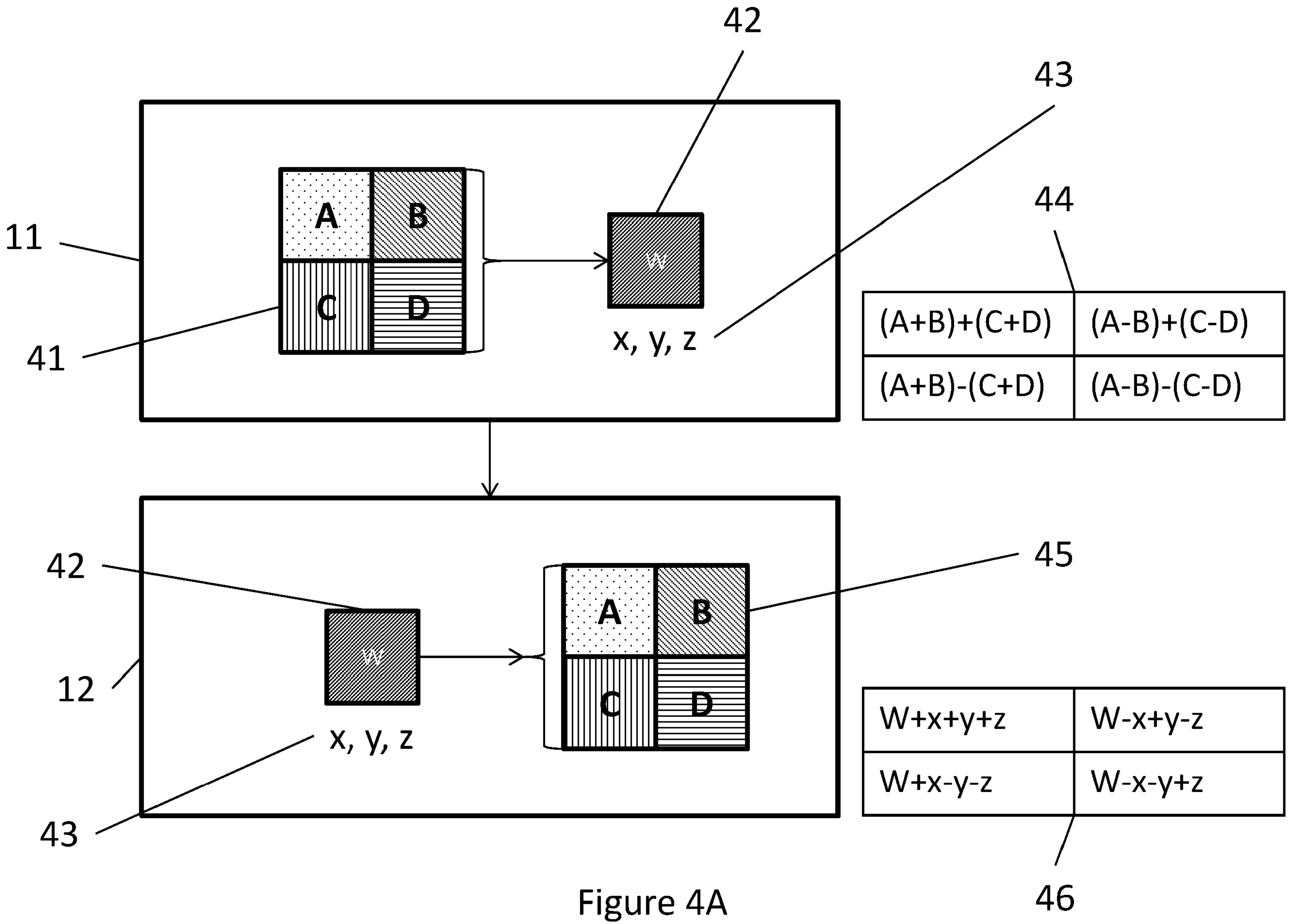
11
41
A
B
C
D
42
W
43
x, y, z
44
(A+B)+(C+D)
(A-B)+(C-D)
(A+B)-(C+D)
(A-B)-(C-D)
12
42
43
45
46
W+x+y+z
W-x+y-z
W+x-y-z
W-x-y+z

Figure 4A

Figure 4B

ENCODE

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

S31: PASS 1

18	-2	26	-2	34	-2	42	-2
-16	0	-16	0	-16	0	-16	0
82	-2	90	-2	98	-2	106	-2
-16	0	-16	0	-16	0	-16	0
146	-2	154	-2	162	-2	170	-2
-16	0	-16	0	-16	0	-16	0
210	-2	218	-2	226	-2	234	-2
-16	0	-16	0	-16	0	-16	0

S32: PASS 2

216	-2	-16	-2	280	-2	-16	-2
-16	0	-16	0	-16	0	-16	0
-128	-2	0	-2	-128	-2	0	-2
-16	0	-16	0	-16	0	-16	0
728	-2	-16	-2	792	-2	-16	-2
-16	0	-16	0	-16	0	-16	0
-128	-2	0	-2	-128	-2	0	-2
-16	0	-16	0	-16	0	-16	0

S33: PASS 3

2016	-2	-16	-2	-128	-2	-16	-2
-16	0	-16	0	-16	0	-16	0
-128	-2	0	-2	-128	-2	0	-2
-16	0	-16	0	-16	0	-16	0
-1024	-2	-16	-2	0	-2	-16	-2
-16	0	-16	0	-16	0	-16	0
-128	-2	0	-2	-128	-2	0	-2
-16	0	-16	0	-16	0	-16	0

S34

DECODE

2016	-2	-16	-2	-128	-2	-16	-2
-16	0	-16	0	-16	0	-16	0
-128	-2	0	-2	-128	-2	0	-2
-16	0	-16	0	-16	0	-16	0
-1024	-2	-16	-2	0	-2	-16	-2
-16	0	-16	0	-16	0	-16	0
-128	-2	0	-2	-128	-2	0	-2
-16	0	-16	0	-16	0	-16	0

S35: PASS 1

216	-2	-16	-2	280	-2	-16	-2
-16	0	-16	0	-16	0	-16	0
-128	-2	0	-2	-128	-2	0	-2
-16	0	-16	0	-16	0	-16	0
728	-2	-16	-2	792	-2	-16	-2
-16	0	-16	0	-16	0	-16	0
-128	-2	0	-2	-128	-2	0	-2
-16	0	-16	0	-16	0	-16	0

S36: PASS 2

18	-2	26	-2	34	-2	42	-2
-16	0	-16	0	-16	0	-16	0
82	-2	90	-2	98	-2	106	-2
-16	0	-16	0	-16	0	-16	0
146	-2	154	-2	162	-2	170	-2
-16	0	-16	0	-16	0	-16	0
210	-2	218	-2	226	-2	234	-2
-16	0	-16	0	-16	0	-16	0

S37: PASS 3

0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31
32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

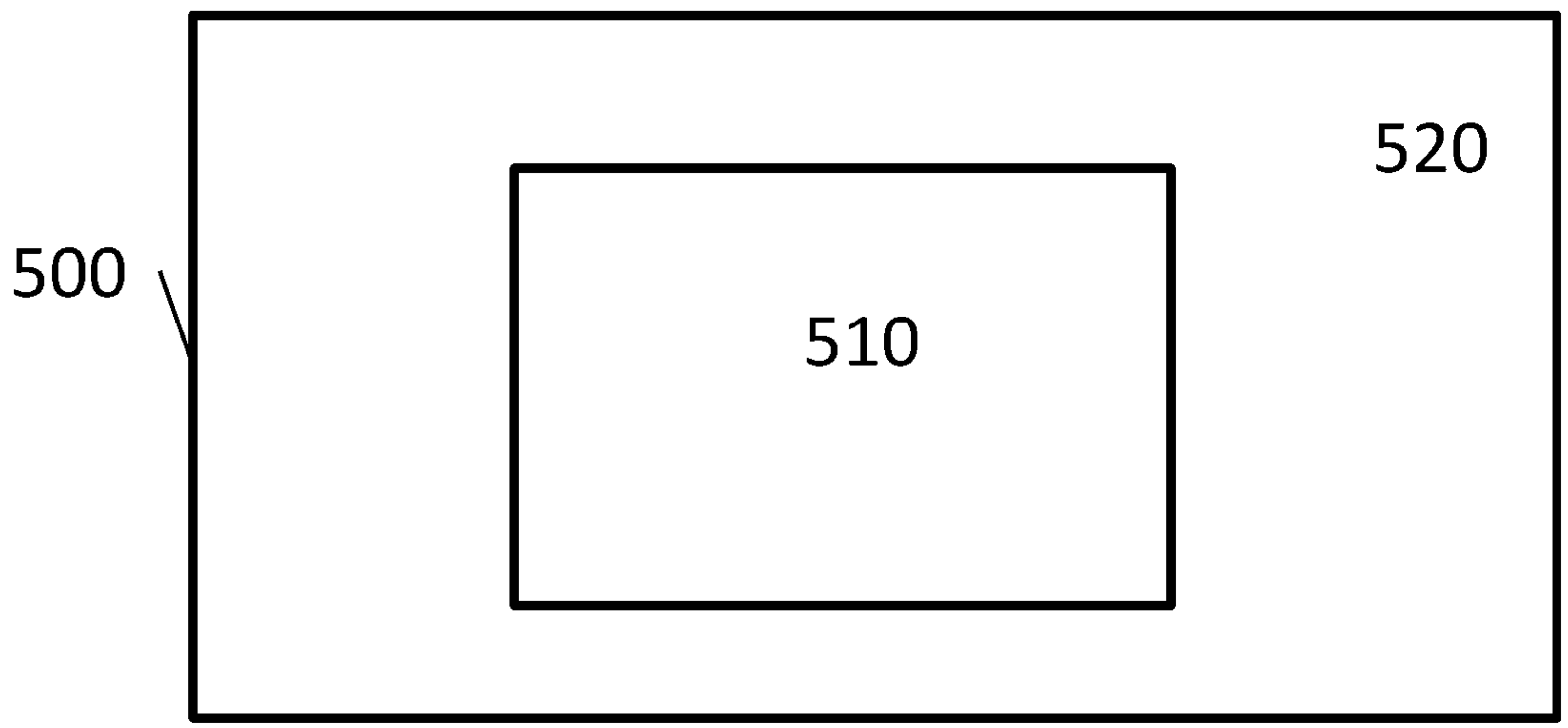

Figure 5A

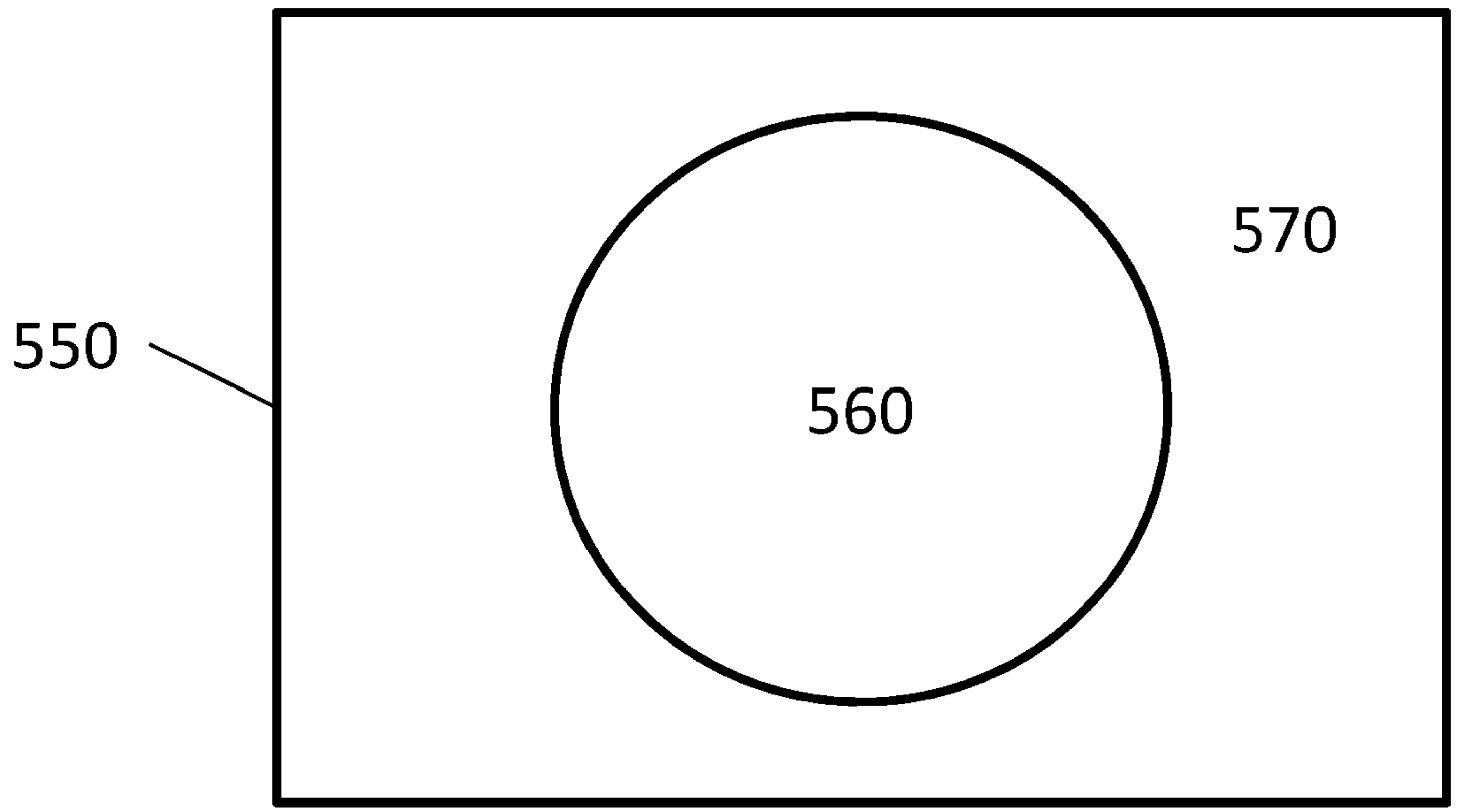

Figure 5B

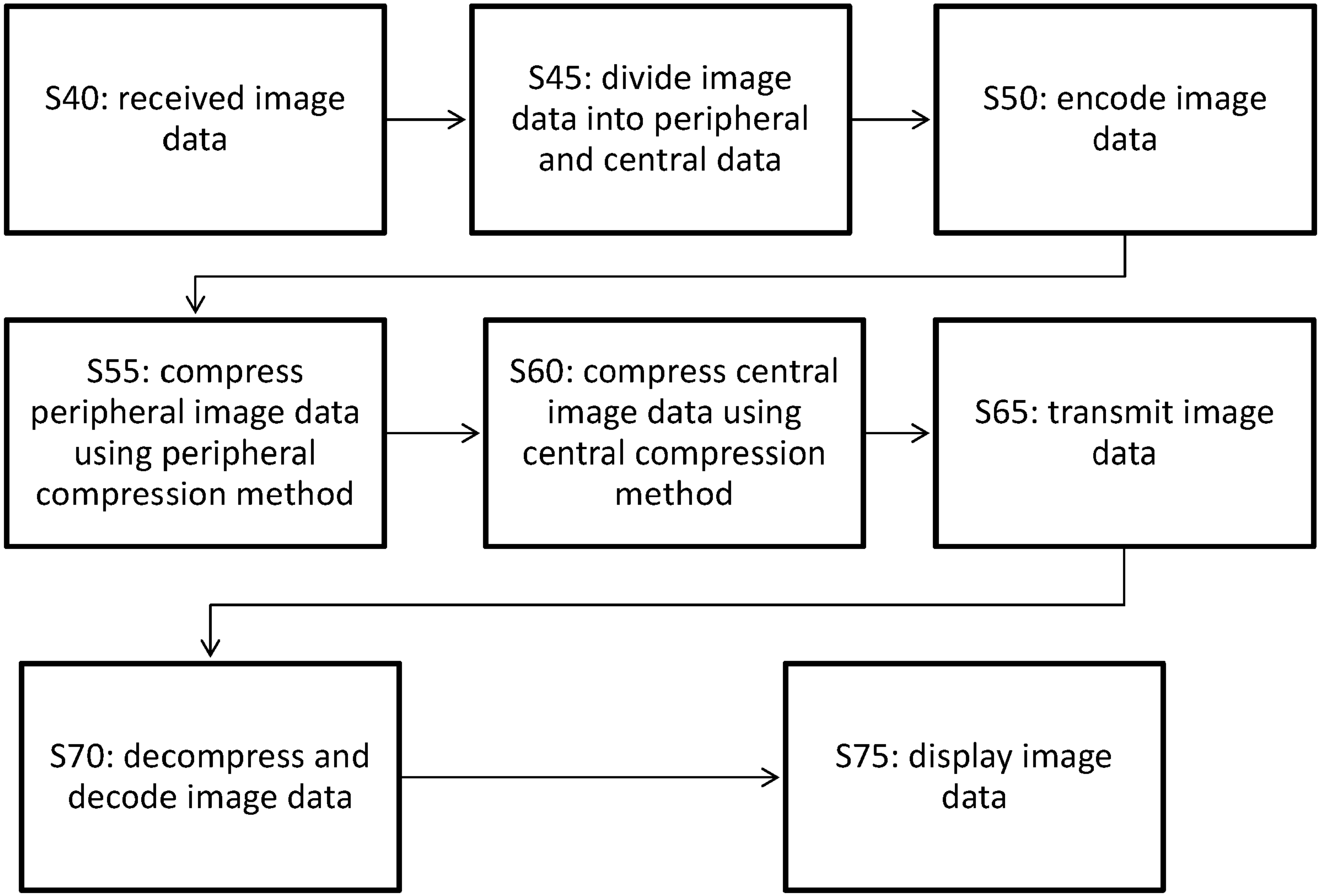
S40: received image data
S45: divide image data into peripheral and central data
S50: encode image data
S55: compress peripheral image data using peripheral compression method
S60: compress central image data using central compression method
S65: transmit image data
S70: decompress and decode image data
S75: display image data

Figure 6

VIDEO IMAGE PROCESSING

FIELD OF THE INVENTION

The present invention relates to processing for video images. More particularly, the present invention relates to improvements in processing the peripheral areas of video images, which has particular applications in high-quality video imaging, such as for virtual reality (VR).

BACKGROUND OF THE INVENTION

Processing images for VR and other similar applications requires high quality video streams to be processed and transmitted rapidly, potentially over wireless links. Current development of display technology, especially in virtual reality, aims to match the performance of computer displays to the performance of the human eye. However, as more and more display data is required in more complicated scenarios, and it is desirable to use wireless technology, rather than wires, the amount of data to be sent becomes very large. It would be desirable to try to minimise the amount of data that actually needs to be sent, while maintaining a good user experience.

Accordingly, the present invention tries to reduce the amount of display data that needs to be sent for display to try to mitigate the above problems.

SUMMARY OF THE INVENTION

Aspects of the invention are set out in the independent claims and preferred features are set out in the dependent claims.

There is described herein a method for processing data for display on a screen, the method comprising the steps of: receiving image data for display on the screen; designating the received image data into peripheral image data and central image data; encoding the image data; compressing the encoded image data for transmission using a peripheral compression method for the peripheral image data and using a central compression method for the central image data, the central compression method being different from the peripheral compression method; and transmitting the compressed image data over a link for display on the screen.

The inventors have observed that the amount of data to be transmitted can be reduced without negatively affecting user experience by compressing image data in the peripheral zones of an image, e.g. by reducing the amount of information transmitted about peripheral areas of the image. Thus it is possible to provide acceptable video images that can be transmitted more efficiently, for example more quickly and/or at lower bandwidth. In particular, user perception of colour in peripheral areas is generally reduced so in some embodiments colour information for the peripherals can be compressed. In general edges and movement in peripheral areas are more noticeable than shades of colour and remain important for user perception, therefore information about movement in peripheral zones may be maintained, or at least not compressed as much as colour data.

Peripheral areas of the image are, for example, those intended to be displayed on peripheral areas of the screen, or to be viewed in peripheral areas of the user's vision. In some examples, image data to be displayed at the periphery of the screen (or display panel) may in fact be in centre of a user's vision and thus be treated as central image data, for example where the image data is intended to be displayed on multiple displays (e.g. in the case of goggles having two displays, one for each eye).

Such a method may be performed on a video encoder device. The method may in some cases be performed at a video recording device.

In preferred embodiments the peripheral compression method reduces image data resolution more than the central compression method.

Each compression method may be described by a set of compression rules, such as peripheral compression rules and central compression rules.

Preferably, the peripheral compression method reduces image resolution, or the quantity of peripheral image data, while retaining information relating to edges in the image data preferentially (e.g. retaining data describing edges over retaining non-edge data and/or over colour or tone data). Additionally, or alternatively, movement data may be retained in the peripheral region, e.g. by comparing image data describing consecutive frames to identify the movement. The peripheral compression method may retain (more) visible edges in the data preferentially, e.g. those edges that are visible, or more visible, to the human eye, such as due to a contrast in luminance or colour or tone. The data may be compressed by a quantization method.

Preferably, the peripheral compression method favours retaining information relating to edges in the image data over information relating to colour.

In some embodiments, the peripheral compression method comprises applying edge detection; preferably by applying a high pass filter. Conventional methods to reduce data quantity have involved compression, using e.g. quantization, to emphasise low frequency components in comparison to high frequency components. However the inventors have found that in fact emphasising high frequency components, e.g. using a high pass filter, especially in the peripheral zones, may be more beneficial and provide a better image for viewing by a user.

Optionally, the peripheral compression method comprises reducing chrominance information in the peripheral image data; preferably by chroma subsampling, or severely chroma subsampling. For example, the chrominance information in the peripheral image data may be compressed more than luminance information, or components. This may be achieved, e.g. by quantization. In some embodiments the quantisation may involve quantising (or reducing) the chrominance AC values significantly (or removing AC chrominance values completely), whilst retaining the chrominance DC values. However, at the same time, in some embodiments the luminance data AC values may be favoured (or reduced less) in order to sharpen the edges. In some examples, encoding the image data comprises encoding using a colour space having at least one chrominance component and at least one luminance component. For example, there may be two chrominance components and one luminance component. The colour space may be based on YUV, or modified versions thereof.

In preferred embodiments, the colour space is based on HSL (hue, saturation, luminance) or HSV (hue, saturation, value), or modified versions thereof. Using HSL or HSV colour space may provide better separation of luminance data compared to other colour spaces (e.g. YUV). Thus by using HSL or HSV it may be possible to retain the luminance data (and reduce chrominance data) more effectively and thus achieve "sharper" edges.

In the central zone it would be best to retain more detail in the colour / chrominance component(s) compared to the peripheral area. In one example, this may be achieved by changing the A-B-C-D terms in the HAAR transform. The peripheral compression method may comprise reducing or compressing the chrominance component(s) more than the central compression method. Additionally or alternatively, the peripheral compression method may comprise reducing or compressing the chrominance component(s) more than luminance component(s). However the central compression method may comprise reducing or compressing the chrominance component(s) less than, or to a similar degree to, the luminance component(s).

When using HSL or HSV, for example, the peripheral compression method may comprise reducing or compressing the hue/saturation components more than the luminance or brightness components.

In some examples, encoding the image data comprises encoding using a reversible colour transform (RCT).

Preferably, encoding the image data comprises performing a transform to obtain, for each frame block, or tile, in each frame of the image data, a general term describing image data in the block as a whole and a plurality of specific terms describing image data for individual pixels in the block. When a DCT is used to encode data, the general term is a DC term and the specific terms are AC terms. In some examples the general term is the A component of a HAAR transform and the specific terms are the B-C-D terms of the HAAR transform.

Optionally, where encoding the image data comprises encoding using a colour space having chrominance and luminance components, the peripheral compression method may comprise discarding the specific terms for at least the chrominance components. Preferably the specific terms for at least or only the chrominance components will be discarded.

Optionally, encoding the image data comprises performing a discrete cosine transform (DCT) to obtain a DC term and a plurality of AC terms for each frame block, or tile, of the image data; and the peripheral compression method comprises discarding the AC terms for at least the chrominance components. Preferably the AC terms are discarded for at least the chrominance components, or only for the chrominance components (e.g. AC terms are retained for the luminance components). This can retain edge information which results in improved user perception of images.

In some embodiments, the peripheral compression method comprises performing quantisation which favours the specific terms more than the general terms. Preferably, the central compression method comprises performing quantisation which favours the general terms more than the specific terms.

Generally, the central image data is data relating to a central portion of the image intended to be displayed on a central zone of the screen, preferably the central zone is expected to be viewed in a central portion of a viewer's vision; and/or the peripheral image data is data relating to a peripheral portion of the image intended to be displayed on a peripheral zone of the screen, preferably wherein the peripheral zone is expected to be viewed in a peripheral portion of the viewer's vision. The central portion of a viewer's vision may be the foveal portion, while the peripheral portion may be the non-foveal area or zone.

Preferably, the method further comprises: providing a model of the sensitivity of a viewer to colour and/or movement in images displayed at different locations on the screen; and designating the received image data into peripheral image data and central image data based on the model of the sensitivity of the viewer. The model of the sensitivity of the viewer may be a model of a typical, average viewer, for example obtained from taking measurements of a group of people. The model may be based on particular characteristics of a user's eye, or group of users, e.g. for users with an astigmatism. In some embodiments the model may be an individual model, tailored to the viewer.

Optionally, the model of the sensitivity of a viewer comprises a measure of the sensitivity of the viewer to at least one of: chrominance, luminance, contrast, edges, and movement.

Preferably, the method further comprises designating the received image data by:

comparing the modelled sensitivity of the viewer across the screen to a threshold sensitivity; defining an area of the screen for which the modelled sensitivity is above the threshold sensitivity as the central zone of the screen; and defining an area of the screen for which the user sensitivity is below the threshold colour sensitivity as the peripheral zone of the screen.

In some examples, the model of the sensitivity of the viewer is based on an expected sensitivity of the viewer across the viewer's visual field and which locations of the screen are expected to be in the viewer's visual field, or in particular parts of the viewer's visual field (e.g. foveal/non-foveal areas), in use. For example a model of which parts of a screen on a headset may be in the viewer's visual field when the headset is worn by the user may be used. Expected sensitivity may be based on a model developed by measuring vision of a population of people and/or based on configuration and shape of viewing equipment, e.g. screen and/or headset.

Preferably, the model of the sensitivity of a viewer to light displayed at different locations on the screen is based on the screen equipment and the expected position of the viewer in use.

Preferably, the image data comprises a set of tiles and all the data in a single tile is compressed using the same compression method. Thus each tile may be part of either peripheral image data or central image data, but not both. Dividing the received image data into peripheral image data and central image data may comprise allocating each tile as either a peripheral tile or a central tile.

In some embodiments, the peripheral image data and central image data relate to predetermined peripheral and central areas of the image or screen. For example, the peripheral image data and central image data may relate to data expected or intended to be displayed on the peripheral and central areas of a screen or display, respectively.

In other embodiments, the peripheral image data and central image data relate to peripheral and central areas of the image or screen that are dynamically defined. For example, the peripheral image data and central image data may relate to data expected or intended to be displayed on the peripheral and central areas of a screen or display, respectively. The dynamic allocation may, for example, be dependent on the particular image, user, group of images and/or transmission rate/available bandwidth. For example, as transmission rate/bandwidth decreases, the amount of data selected or designated as peripheral image data may increase and the amount of data selected or designated as central image data may decrease.

Preferably, the peripheral and central areas are defined based on one or more of: a detection of the position or orientation of the eye of a viewer; a model of the vision of an individual user or general model; image content; image data; and a measured data speed or latency in the link.

There is also described herein a method for processing data for display on a screen comprising the steps of: receiving compressed image data over a link, wherein the compressed image data comprises: peripheral image data, compressed using a peripheral compression method; and central image data, compressed using a central compression, the central compression method being different from the peripheral compression method; decompressing the compressed image data; decoding the image data; providing the central image data for display on a central area of the screen; and providing the peripheral image data for display on a peripheral area of the screen.

For example, such a method may be performed at a video decoder device and/or a video display device.

The method may further comprise displaying the central image data on a central area of the screen; and displaying the peripheral image data on peripheral area of the screen.

The screen may be located in a remote device, such as a headset, configured to be worn on a viewer's head, preferably a virtual reality headset or an augmented reality set of glasses. The screen may be a display panel of a headset display, preferably the headset display has two display panels, for example one corresponding to each eye of a user.

The methods described above may work when the link is a wireless link.

In some embodiments, the display data comprises a set of frames for display on the screen at a frame rate of at least 50 frames per second, preferably at least 60 frames per second, more preferably at least or about 90 frames per second, more preferably about 120 frames per second.

In some embodiments, the central image data and/or the peripheral image data make up at least 10% of the total image data, for example 10% of the total area image display area on a screen. In some embodiments the central image data and/or the peripheral image data make up at least at least 15%, preferably at least 20% or at least 30%.

There is also described herein a method for use in image data processing, comprising the steps of: receiving image data for display on the screen; designating the received image data into peripheral image data and central image data; encoding the image data; compressing the encoded image data for transmission using a peripheral compression method for the peripheral image data and using a central compression method for the central image data, the central compression method being different from the peripheral compression method; and transmitting the compressed image data over a link for display on the screen; receiving compressed image data over a link, wherein the compressed image data comprises: peripheral image data, compressed using a peripheral compression method; and central image data, compressed using a central compression, the central compression method being different from the peripheral compression method; decompressing the compressed image data; decoding the image data; providing the central image data for display on a central area of the screen; and providing the peripheral image data for display on a peripheral area of the screen

There is also described herein an encoder device for processing data for display on a screen, the encoder device comprising: a processor; a memory; and an interface for communicating with a decoder device over a link; wherein the encoder device is arranged to: receive image data for display on the screen; designate the received image data into peripheral image data and central image data; encode the image data; compress the encoded image data for transmission using a peripheral compression method for the peripheral image data and using a central compression method for the central image data, the central compression method being different from the peripheral compression method; and transmit the compressed image data over the link for display on the screen.

The encoder device may be further arranged to perform some of the methods described above.

There is also described herein an decoder device for processing data for display on a screen, the decoder device comprising: a processor; a memory; and an interface for communicating with an encoder device over a link; wherein the encoder device is arranged to: receive compressed image data over the link, wherein the compressed image data comprises: peripheral image data, compressed using a peripheral compression method; and central image data, compressed using a central compression method, the central compression method being different from the peripheral compression method; decompress the compressed image data; decode the image data; provide the peripheral image data for display on a peripheral area of the screen; and provide the central image data for display on a central area of the screen

The decoder device may be further arranged to perform some of the methods described above.

There is also described herein a system for displaying image data, the system comprising: an encoder device as described above; a decoder device as described above; and a screen for displaying the image data.

Preferably, the screen is mounted in a remote device, such as a headset configured to be worn on a viewer's head, preferably a virtual reality headset or an augmented reality set of glasses.

In some embodiments, the decoder device is located in the remote device.

In some embodiments, the encoder device and the decoder device are in communication via a wireless link.

BRIEF DESCRIPTION OF THE DRAWINGS

Embodiments will now be described, by way of example only and with reference to the accompanying drawings, in which:

Figure 1 illustrates an exemplary system for displaying images;

Figure 2A illustrates an exemplary display device;

Figure 2B illustrates an exemplary display device and user's eye;

Figure 3 illustrates another exemplary display device system;

Figure 4A illustrates an exemplary method for processing data for display using a Haar transform;

Figure 4B illustrates an exemplary method for processing data for display using a Haar transform;

Figure 5A illustrates an exemplary screen for displaying images;

Figure 5B illustrates another exemplary screen for displaying images; and

Figure 6 illustrates an exemplary method for processing image data.

DETAILED DESCRIPTION OF THE INVENTION

Example System Configuration

Figure 1 shows a block diagram overview of a system according to the current art. A host computer [11] is connected to a display control device [12], which is in turn connected to a display device [13]. The host [11] contains an application [14], which produces display, or image, data. The display data may be produced and sent for compression either as complete frames or as canvasses, which may, for example, be separate application windows. In either case, they are made up of tiles of pixels, where each tile is a geometrically-shaped collection of one or more pixels. Generally tiles will be rectangular, preferably square, in shape.

The display data is sent to a compression engine [15], which may comprise software running in a processor or an appropriate hardware engine. The compression engine [15] first performs an encoding of the data, for example using a Haar transformation, to convert the data into a format that may then be further compressed, minimising data loss.

The compression engine [15] may then further compress the data and thereafter sends the compressed data to an output engine [16]. The output engine [16] manages the connection with the display control device [12] and may, for example, include a socket for a cable to be plugged into for a wired connection or a radio transmitter for a wireless connection. In either case, it is connected to a corresponding input engine [17] on the display control device [12].

The input engine [17] is connected to a decompression engine [18]. When it receives compressed data it sends it to the decompression engine [18] or to a memory from which the decompression engine [18] can fetch it according to the operation of a decompression algorithm. In any case, the decompression engine [18] may decompress the data, if necessary, and performs a decoding operation, optionally using a reverse Haar transform. In the illustrated system, the decompressed data is then sent to a scaler [19]. In the case where the display data was produced and compressed as multiple canvasses, it may be composed into a frame at this point.

If scaling is necessary, it is preferable for it to be carried out on a display control device [12] as this minimises the volume of data to be transmitted from the host [11] to the display control device [12], and the scaler [19] operates to convert the received display data to the correct dimensions for display on the display device [13]. In some embodiments, the scaler may be omitted or may be implemented as part of the decompression engine. The data is then sent to an output engine [110] for transmission to the display device [13]. This may include, for example, converting the display data to a display-specific format such as VGA, HDMI, etc.

In one embodiment, the display device is a virtual reality headset [21], as illustrated in Fig. 2A, connected to a host device [22], which may be a computing device, gaming station, etc. The virtual reality headset [21] incorporates two display panels [23], which may be embodied as a single panel split by optical elements. In use, one display is presented to each of a viewer's eyes. The host device [22] generates image data for display on these panels [23] and transmits the image data to the virtual reality headset [21].

In another embodiment, the headset is a set of augmented reality glasses. As in the virtual reality headset [21] shown in Figure 2A, there are two display panels, each associated with one of the user's eyes, but in this example the display panels are translucent.

The host device [22] may be a static computing device such as a computer, gaming console, etc., or may be a mobile computing device such as a smartphone or smartwatch. As previously described, it generates image data and transmits it to the augmented reality glasses or virtual reality headset [21] for display.

Figure 2B shows a cross-section of part of the headset [21], showing a display panel [23] and a lens [24] in front of it, together with a user's eye [25] when the system is in use. The display panel [23] is similar to the display panel shown in Figure 2A, and the lens [24] is of a standard type.

Display data, or image data, is produced by the application [14] on the host [11] and transmitted to the headset [21], where it is received by the display control device [12]. It is processed for display on the display panels [23]. It is then sent to the display panels [23] for display. The user views it through the lenses [24], which allow focussing at the very close range required. However, they distort the image, in particular by chromatic aberration, as a lens [24] may introduce a colour shift to the image as observed by the viewer. For example, thick glass may introduce a slight blue or green tint. Chromatic aberration tends to be more pronounced at the edges or periphery of the image, where the angle is greatest. This may be known and accounted for by the application [14], for example, the application [14] may introduce a slight red tint to the images prior to transmission, which may vary across the image.

The display device may be connected to the host device [11, 22] or display control device [12] if one is present by a wired or wireless connection. While a wired connection minimises latency in transmission of data from the host to the display, wireless connections give the user much greater freedom of movement within range of the wireless connection and are therefore preferable. A balance must be struck between high compression of data, in particular video data, which can be used to enable larger amounts of data (e.g. higher resolution video) to be transmitted between the host and display, and the latency that will be introduced by processing of the data.

Ideally, the end-to-end latency between sensing a user's head movement, generating the pixels in the next frame of the VR (virtual reality) scene and streaming the video should be kept below 20ms, preferably below 10ms, further preferably below 5ms.

The wireless link should be implemented as a high bandwidth short-range wireless link, for example at least 1 Gbps, preferably at least 2 Gbps, preferably at least 3 Gbps. An “extremely high frequency (EHF)” radio connection, such as a 60GHz radio connection is suitable for providing such high-bandwidth connections over short-range links. Such a radio connection can implement the WiFi standard IEEE 802.11ad.

The 71-76, 81-86 and 92–95 GHz bands may also be used in some implementations.

The wireless links described above can provide transmission between the host and the display of more than 50 frames per second (fps), preferably more than 60fps, further preferably more than 90fps, more preferably more than 120fps.

Figure 3 shows a system which is similar in operation to the embodiment shown in Figures 2A and 2B. In this case, however, there is no separate host device [22]. The entire system is contained in a single casing [31], for example in a smartphone or other such mobile computing device. The device contains a processor [33], which generates display data for display on the integral display panel [32]. The mobile computing device may be mounted such that the screen is held in front of the user’s eyes as if it were the screen of a virtual reality headset.

Haar Encoding

A Haar transformation process may be implemented in conjunction with the present system, such as those shown in Figures 1 to 3. The Haar transform takes place on the host [11], specifically in the compression engine [15]. Decompression takes place on the display control device [12], specifically in the decompression engine [18], where the data is put through an inverse Haar transform to return it to its original form, or an approximation thereof.

In the example shown in Figure 4A, a group of four tiles [41] has been produced by the application [14] and passed to the compression engine [15]. In this example, each tile [41] comprises one pixel, but may be larger. Each pixel [41] has a value indicating its colour, here represented by the pattern of hatching. The first pixel [41A] is marked with dots and considered to have the lightest colour. The second pixel [41B] is marked with diagonal hatching and is considered to have the darkest colour. The third pixel [41C] is marked with vertical hatching and is considered to have a light colour, and the fourth pixel [41D] is marked with horizontal hatching and is considered to have a dark colour. The values of the four pixels [41] are combined using the formulae [44] shown to the right of Figure 4A to produce a single pixel value [42], referred to as "W", which is shaded in grey to indicate that its value is derived from the original four pixels [41], as well as a set of coefficients [43] referred to in Figure 4A as "x, y, z". The pixel value [42] is generated from a sum of the values of all four pixels: ((A+B)+(C+D)). The three coefficients [43] are generated using the other three formulae [44] as follows:

- x: (A-B)+(C-D)
- y: (A+B)-(C+D)
- z: (A-B)-(C-D)

Any or all of these values may then be quantised, e.g. divided by a constant in order to produce a smaller number which will be less accurate but can be more effectively compressed and rounded. In general a Haar transform will sharpen up the edges of an image, but the quantization can be targeted to preferentially sharpen edges, whilst allowing other data (e.g. data that does not contribute to visible edges) to be reduced. For example by favouring AC values (e.g. in comparison to DC values), edges can be sharpened. Conversely, by favouring DC values, edges could be dulled.

The reverse transform process is carried out on the single pixel value [42] and coefficients [43] produced in the transform as described above. This process will be carried out after a decompression process, which might involve, for example, multiplying quantised coefficients to restore an approximation of their original values.

The decompression engine [18] combines the coefficients [43] with the value of the pixel value [42] transmitted by the host [11] to recreate the original four pixels [45] (or at least approximations) using the formulae [46] shown to the right of Figure 4A.

- A: W+x+y+z
- B: W-x+y-z
- C: W+x-y-z
- D: W-x-y+z

This is repeated the same number of times that the data was transformed. These pixels [45] are then transmitted to the scaler [19] if a scaler is used.

Figure 4B shows an example of the actual process of a Haar transform. The top part of the encode section shows 64 tiles, each numbered in order from 0. These numbers are used to indicate the values of the tiles as previously mentioned. The tiles are divided into groups of four: for example, the top-left group comprises tiles 0, 1, 8, and 9.

At each pass, the same calculations are performed on a larger range of tile groups to produce combined pixel values and coefficients. In the first pass, Step S31, the tiles in each group are processed using the previously-mentioned formulae [44]. This converts the values in the circled first group to 18, -2, 16, 0 and these values are stored and used in place of the original values. In this example, 18 is the pixel value “W” [42] described in Figure 4A and -2, -16, and 0 are the coefficients “x”, “y”, and “z” [43] described in Figure 4A. The same process is carried out on all the groups. These results are shown in the second section of the process, after Step S31.

The second pass, Step S32, applies the same formulae [44] to the top-left tiles in each set of four tile groups. The values to be used in the top-left quarter of the frame in the second pass are shown circled: 18, from the top-left group, 26 from the group to the immediate right, 82 from the group below, and 90 from the final group in the upper-left quarter. The same formulae [44] are then applied to these values to produce 216, -16, -128, and 0, which are put in the places of the original values. Again, these values correspond to W [42], x, y, and z [43] as described in Figure 4A. The same process is carried out on all four quarters, and all other values are unchanged: for example, in the top-left group the three values not used in the second pass of the transform and not circled are unchanged from -2, -16, and 0.

The third pass, Step S33, is carried out on one value from each quarter, as shown circled in Figure 4B: 216 from the top-left quadrant, 280 from the top-right quadrant, 728 from the bottom-left quadrant, and 792 from the bottom-right quadrant. This produces the final results shown at the bottom of the encode section: 2016 (W), -128 (x), -1024 (y), and 0 (z). Once again, all the other values are unchanged.

The values can then be rearranged so that the different coefficients are grouped together. The pixel values at each level are transmitted first, prioritising the results of later passes, followed by the coefficients. This will result in many small numbers, including many identical numbers: for example, there is a 0 in the same position in each group after the third pass, and these can be grouped and sent as a single number. The values may also be quantised: divided by a constant to produce smaller coefficients and rounded, if desired.

At Step S34, the data is transmitted from the host [11] to the display control device [12], where it is decompressed, de-quantised and re-ordered as appropriate prior to decoding. In this example, these processes produce the same data as was generated by the initial transform, and this table is shown at the beginning of the Decode section. A similar process is then performed to reverse the transform process.

At Step S35, the first pass is performed and the formulae [46] described in Figure 4A are applied to the circled top-left tile from each quadrant: as mentioned after the third pass of the encode stage, in this pass, and this example, the figures are: 2016 (W [42]), -128 (x [43]), -1024 (y [43]), and 0 (z [43]). This produces a new W value [42] for each quadrant: 216, 280, 728, and 792.

At Step S36, the second pass is carried out. It takes the top-left value from each group in each quadrant (W:216 [42], x: -16, y: -128, z: 0 [43]) and applies the same formulae [46] to them. Finally, the same formulae [46] are applied to every value in each group in the third pass: Step S37. This produces the same values as were input at the beginning of the encode section.

Such a transform is useful because not only does it allow the host [11] to transmit a smaller number of pixels than are present in the full image data, combined with a collection of coefficients, but the coefficients can be compressed more efficiently than pixel data, with less loss of data; they are small numbers and so can be transmitted in fewer bits without any further compression being applied.

Encoding of Colour Images

There are several different types of colour space used in imaging. A colour space allows for a representation of colours to be encoded in digital data. Some colour spaces are based on the RGB colour model, which uses additive primary colours (red, blue, green).

The term "colour space" may be used to describe a particular combination of a colour model and mapping function. A mapping function for a colour space defines a link between the colour model and a reference colour space, thereby defining a colour space. In some cases the term "colour space" is also used (generally more informally) to identify a colour model. However, although identifying a colour space may also automatically identify the associated colour model, the converse is not necessarily true as each colour model may have several associated colour spaces. For example, several specific colour spaces are based on the RGB colour model.

CMYK is another colour model which uses subtractive colours (cyan, magenta, yellow and black). Another way of representing colours is by using an HSL or HSV colour model (hue, saturation, luminance/brightness/value), CIE (International Commission on Illumination, in French Commission internationale de l'éclairage) colour model or YUV.

In YUV colour space, the Y component determines the brightness, or luminance, of colour and the U and V components define the chrominance, or the colour itself. Generally Y values range from 0 to 1 (or 0 to 255 in digital formats) and U and V values range from -0.5 to 0.5 (or -128 to 127 in signed digital form, or 0 to 255 in unsigned form). Variations on YUV colour space exist, such as Y'UV, YUV, YCbCr, YPbPr. In some scenarios encoding image data based on YUV colour space (or a variant thereof, having luminance and chrominance components) can reduce the amount of data required to transmit an image. For example, the chrominance components can be reduced or compressed without having a huge impact on human perception because the brightness information encoded by the luminance channel has comparatively greater impact on discerning the image detail. In particular, in the peripheral areas of vision the human perception of chrominance may be greatly reduced, while perception of luminance may be higher.

The representation of a colour can be translated from one basis to another by colour space conversion or transformation.

Image Filtering

Various image filters may be used in image processing, depending on the desired effect on the image or the image data. A filter may be defined by a small filter array applied to each pixel and its neighbours within an image. The process used to apply filters to an image is sometimes known as convolution, and may be applied in the spatial or frequency domain. Such filters may be used to compress data through lossy compression.

In one exemplary filtering process in the spatial domain, the first part of the convolution process multiplies the elements of the filter array by the matching pixel values. The elements of the resulting array are then averaged, and the original pixel value is replaced with this result.

In another example in the frequency domain, convolution can be performed by multiplying the FFT (Fast Fourier Transform) of the image by the FFT of the array, and then transforming back into the spatial domain.

One example filter is a low pass filter, which can be used to smooth an image, e.g. by decreasing the disparity between pixel values by averaging nearby pixels. Using a low pass filter tends to retain the low frequency information within an image while reducing the high frequency information.

Conversely, a high pass filter may be used to sharpen an image, e.g. by enhancing contrast between adjoining areas with little variation in brightness or darkness. A high pass filter tends to retain the high frequency information within an image while reducing the low frequency information. A high pass filter can for example increase the brightness of the centre pixel relative to neighbouring pixels. Thus a high pass filter array usually contains a single positive value at its centre, and is surrounded by negative values. High pass filters may be used to accentuate edges in images.

Another type of filter is a directional filter, which can be used in edge detection methods. An edge within an image is visible when a large difference (or a steep gradient) occurs between adjacent pixel values. This change in values can be measured by calculating the first derivatives of an image. For example, for images x-and y-directional filters may be used to compute derivatives in their respective directions.

Alternatively (or additionally), a Laplacian filter may be used for detecting edges. A Laplacian filter can be used to compute the second derivatives of an image, which helps to determine if a change in adjacent pixel values is an edge or a continuous progression.

Dividing image data

Figure 5A shows an exemplary screen [500] on which image data can be shown. The screen [500] may be used as the display panels [23] described above, for example as display panels in a VR headset. E.g. one screen [500] may be provided for each eye, or the screen [500] may be split and display images to be viewed by both eyes.

The screen [500] is divided into two areas. The first area, or region, [510] is designated as a central region of the screen. The second area, or region, [520] is designated as a peripheral area of the screen [500]. Image data for display on the screen may be divided into central data and peripheral data. A first portion of display data which codes parts of the image to be displayed on the first area, or central region, [510] is designated as central image data. A second portion of display data for display on the second area, or peripheral region, [520] of the screen is designated as peripheral image data.

The peripheral image data is then compressed using a peripheral compression method which differs from a central compression method used for the central image data. Thus the peripheral image data may thus be more compressed than the central image data. Whilst the more compressed peripheral image data may not produce such a good display for the viewer, since this will generally be in the peripheral vision of the user this is of minor significance. The first and second areas [510], [520] of the screen [500] are rectangular, in line with the shape of the entire screen.

Figure 5B shows an exemplary screen [550] on which image data can be shown. As with the screen [500], the screen [550] may be used as the display panels [23] described above, e.g. in a VR headset. The screen [550] is divided into two areas. The first area, or region, [560] is a central region of the screen [550]. The second area, or region, [570] is a peripheral area of the screen [550]. The first region [560] is circular since this generally better reflects the region of the screen which will appear in a viewer's central, or foveal, vision (e.g. where the viewer's vision is more sensitive to chrominance/colour). A first portion of display data which codes parts of the image to be displayed on the first area, or central region, [560] is designated as central image data. A second portion of display data for display on the second, or peripheral, region [570] of the screen is encoded according to a different, second colour space.

The regions [510, 520, 560, 570] may be selected to coincide with tile boundaries for simpler encoding and compression. Where the boundary is curved (e.g. between region 560 and 570) the image data may be divided along steps generally in-line with the curve, so that the boundary still coincides with the edges of tiles.

Image data processing method

Figure 6 illustrates an exemplary method for processing image data. At step S40 image, or display, data is received. This data may be in the form of a frame, which may be part of a set of frames making up moving, or video, images. The frames may be comprised of tiles.

At step S45 the data is divided into peripheral data and central data. For example this division may be based on where on a screen the data is intended to be displayed, or where in the image the data is. In some cases the division of the data may also be determined by the type or content of the image data. For example, where image data has more visible boundaries/edges and/or movement (e.g. transitions between consecutive frames) and/or less colour variation, the data selected as peripheral image data may correspond to a larger proportion of image/screen than in cases where image data has fewer boundaries/edges, movement and/or more colour variation. This is because generally the peripheral compression method selected reduces colour (e.g. chrominance) information more than edge information.

In some embodiments, the division of image data into peripheral and central data is performed based on the transmission rate of the link over which the data is to be transferred. For example, where transmission speed is higher the peripheral area may be reduced and the central area may be increased. Where transmission speed is lower the peripheral area may be increased (and central area reduced) to increase the amount data which receives greater compression and thus overall reduce the amount of data to be sent over the link. Thus the amount of data classed as peripheral and central data may vary between consecutive images, e.g. it may change dynamically.

In some embodiments the peripheral and central data may be determined based on a user's eye orientation or eye movement. For example, if it is detected that the user has moved their eyes to look in a certain direction, different parts of the screen may now appear in their central (or foveal) and peripheral vision.

At step S50 the image information is encoded. For example, it may be encoded in a particular colour space, according to a particular colour model. For example, the received image information may be in a colour space based on a colour model having three chrominance components, such as an RGB colour model, and the encoding process may comprise translating to another colour space, such as a colour space having a chrominance and two luminance components, e.g. based on a YUV colour model.

Encoding may also (additionally or alternatively) comprise performing a discrete cosine transformation (DCT) on the data. A DCT generally produces, for each tile having a number of pixel values, a DC term, which corresponds to the average of the values of the pixels in a tile, such as the average colour or average brightness, and a number of AC terms, which represent difference values in the colour or brightness for the pixels in the tile. The AC values are generally stored in order of increasing value. Thus for a tile size of *N* pixels, performing a DCT transform results in generation of a single DC value and also in generation of *N-1* AC values, one for each of the remaining pixels in the frame, these AC values representing the colour change between the pixels, across the frame.

At step S55 the peripheral image data is compressed using a peripheral compression method. Such a peripheral compression method would generally involve lossy compression. Visual features or information which is less likely to be noticed by a user in their peripheral vision may be compressed/reduced more than features that the user would notice. For example, edges and movement are more noticeable in peripheral vision than colour shades. Thus the peripheral compression method may reduce colour information significantly more than edge or contrast information.

For example, colour information may be reduced by chroma subsampling. One exemplary method of compressing for the peripheral zone is to severely chroma subsample the peripheral image data, e.g. in the chrominance components to remove AC values and leave just DC values of the DCT. In other methods, the AC values may be significantly reduced without removing them completely. Reducing or removing the AC values may create artefacts in the peripheral areas of the image shown on the screen, however this does not significantly impact the user experience since the user does not generally concentrate on details shown in the periphery. Reducing or quantising more (e.g. increasing the quantisation value, thus reducing the transformed value) the AC values may, for example, dull the edges. The quantisation value may be dependent on the input data, e.g. on the number of possible values for each pixel, $z = 2^{(\text{number of bits})}-1$.

The peripheral compression method may include quantisation on the Haar transform coefficients to reduce the size of the data generally, but to retain (or emphasise) edge detail.

The peripheral compression method may additionally or alternatively include passing the peripheral image data using an edge detection filter, e.g. a high pass filter, as described above.

In step S60 the central image data is compressed using a central compression method. The central compression method is different from the peripheral compression method used at step S55. In general the central compression method will reduce/compress image data less than the peripheral compression method. In particular, the central compression method may retain more colour/chrominance information than the peripheral compression method.

Thus quantisation may be used, but the quantisation may be different from that used for the peripheral compression method. In some embodiments, different quantisation tables may be used for the peripheral compression and the central compression.

At step S65 the compressed image data is transmitted. For example, the data may be transmitted over a link to a display device, or a decoder connected to the display device, such as display control device [12]. The link may be a wireless link.

In some embodiments, further coding bits or information may be sent to identify which data is peripheral data and which is central data. This may allow the decoder to decode correctly.

At step S70 the display data is decompressed and decoded. The peripheral area data may be decompressed differently from the central area data, e.g. because the compression method was different a different decompression method is required. In such situations, the data may be labelled (e.g. coding bits or other information) with whether it is central or peripheral data and/or what compression method has been used to compress it.

The data is displayed on the screen or display (such as a VR headset display) at step S75.

Any system feature as described herein may also be provided as a method feature, and vice versa. As used herein, means plus function features may be expressed alternatively in terms of their corresponding structure.

Any feature in one aspect of the invention may be applied to other aspects of the invention, in any appropriate combination. In particular, method aspects may be applied to system aspects, and vice versa. Furthermore, any, some and/or all features in one aspect can be applied to any, some and/or all features in any other aspect, in any appropriate combination.

It should also be appreciated that particular combinations of the various features described and defined in any aspects of the invention can be implemented and/or supplied and/or used independently.

The above embodiments and examples are to be understood as illustrative examples. Further embodiments, aspects or examples are envisaged. It is to be understood that any feature described in relation to any one embodiment, aspect or example may be used alone, or in combination with other features described, and may also be used in combination with one or more features of any other of the embodiments, aspects or examples, or any combination of any other of the embodiments, aspects or examples. Furthermore, equivalents and modifications not described above may also be employed without departing from the scope of the invention, which is defined in the accompanying claims.

CLAIMS:

1. A method for processing data for display on a screen, the method comprising the steps of:

    receiving image data for display on the screen;

    designating the received image data into peripheral image data and central image data;

    encoding the image data;

    compressing the encoded image data for transmission using a peripheral compression method for the peripheral image data and using a central compression method for the central image data, the central compression method being different from the peripheral compression method, wherein the peripheral compression method reduces resolution and favours retaining information relating to edges in the peripheral image data preferentially over information relating to colour; and

    transmitting the compressed image data over a link for display on the screen.

2. A method according to claim 1, wherein the peripheral compression method comprises applying edge detection.

3. A method according to either claim 1 or claim 2, wherein the peripheral compression method comprises reducing chrominance information in the peripheral image data.

4. A method according to claim 3, wherein encoding the image data comprises encoding using a colour space having at least one chrominance component and at least one luminance component.

5. A method according to claim 4, wherein the colour space is based on YUV, or modified versions thereof.

6. A method according to claim 4, wherein the colour space is based on HSL (hue, saturation, luminance) or HSV (hue, saturation, value), or modified versions thereof.

7. A method according to any preceding claim, wherein:

    encoding the image data comprises performing a transform to obtain, for each frame block in each frame of the image data, a general term describing image data in the block as a whole and a plurality of specific terms describing image data for individual pixels in the block.

8. A method according to claim 7, wherein encoding the image data comprises encoding using a colour space having at least one chrominance component and at least one luminance component; and wherein the peripheral compression method comprises discarding the specific terms for at least the at least one chrominance component.

9. A method according to claim 8, wherein:

encoding the image data comprises performing a discrete cosine transform (DCT) to obtain a DC term and a plurality of AC terms for each frame block of the image data; and

the peripheral compression method comprises discarding the AC terms for at least the chrominance components.

10. A method according to any of claims 7 to 9, wherein:

the peripheral compression method comprises performing quantisation which favours the specific terms more than the general terms.

11. A method according to claim 10, wherein the central compression method comprises performing quantisation which favours the general terms more than the specific terms.

12. A method according to any preceding claim, wherein:

the central image data is data relating to a central portion of the image intended to be displayed on a central zone of the screen, wherein the central zone is expected to be viewed in a central portion of a viewer’s vision; and/or

the peripheral image data is data relating to a peripheral portion of the image intended to be displayed on a peripheral zone of the screen, wherein the peripheral zone is expected to be viewed in a peripheral portion of the viewer’s vision.

13. A method according to any preceding claim, further comprising:

providing a model of the sensitivity of a viewer to colour and/or movement in images displayed at different locations on the screen; and

designating the received image data into peripheral image data and central image data based on the model of the sensitivity of the viewer.

14. A method according to claim 13, wherein the model of the sensitivity of a viewer comprises a measure of the sensitivity of the viewer to at least one of: chrominance,

luminance, contrast, edges, and movement.

15. A method according to claim 13 or 14, further comprising designating the received image data by:

comparing the modelled sensitivity of the viewer across the screen to a threshold sensitivity;

defining an area of the screen for which the modelled sensitivity is above the threshold sensitivity as the central zone of the screen; and

defining an area of the screen for which the user sensitivity is below the threshold colour sensitivity as the peripheral zone of the screen.

16. A method according to any of claims 13 to 15, wherein the model of the sensitivity of the viewer is based on an expected sensitivity of the viewer across the viewer's visual field and which locations of the screen are expected to be in the viewer's visual field in use.

17. A method according to any of claims 13 to 16, wherein the model of the sensitivity of a viewer to light displayed at different locations on the screen is based on the screen equipment and the expected position of the viewer in use.

18. A method according to any preceding claim, wherein the image data comprises a set of tiles and all the data in a single tile is compressed using the same compression method.

19. A method according to any preceding claim, wherein the peripheral image data and central image data relate to predetermined peripheral and central areas of the image or screen.

20. A method according to any preceding claim, wherein the peripheral image data and central image data relate to peripheral and central areas of the image or screen that are dynamically defined.

21. A method according to claim 20, wherein the peripheral and central areas are defined based on one or more of:

a detection of the position or orientation of the eye of a viewer;

a model of the vision of an individual user or general model;

image content;

image data; and

a measured data speed or latency in the link.

22. A method for processing data for display on a screen, the method comprising the steps of:

receiving compressed image data over a link, wherein the compressed image data comprises: peripheral image data, compressed using a peripheral compression method; and central image data, compressed using a central compression, the central compression method being different from the peripheral compression method, wherein the peripheral compression method reduces resolution and favours retaining information relating to edges in the peripheral image data preferentially over information relating to colour;

decompressing the compressed image data;

decoding the image data;

providing the central image data for display on a central area of the screen; and

providing the peripheral image data for display on a peripheral area of the screen.

23. A method for processing data for display on a screen according to claim 22, further comprising:

displaying the central image data on a central area of the screen; and

displaying the peripheral image data on peripheral area of the screen.

24. A method according to any preceding claim, wherein the screen is in a remote device, such as a headset, configured to be worn on a viewer’s head, the headset being a virtual reality headset or an augmented reality set of glasses.

25. A method according to any preceding claim, wherein the link is a wireless link.

26. A method according to any preceding claim, wherein the display data comprises a set of frames for display on the screen at a frame rate of at least 50 frames per second.

27. A method according to any preceding claim, wherein each of the central image data and peripheral image data make up at least 10% of the total image data.

28. An encoder device for processing data for display on a screen, the encoder device comprising:

a processor;

a memory; and

an interface for communicating with a decoder device over a link; wherein the encoder device is arranged to:

receive image data for display on the screen;

designate the received image data into peripheral image data and central image data;

encode the image data;

compress the encoded image data for transmission using a peripheral compression method for the peripheral image data and using a central compression method for the central image data, the central compression method being different from the peripheral compression method, wherein the peripheral compression method reduces resolution and favours retaining information relating to edges in the peripheral image data preferentially over information relating to colour; and

transmit the compressed image data over the link for display on the screen.

29. An encoder device according to claim 28, further arranged to perform the method of any of claims 2 to 21 or 23 to 26 (when dependent on claims 1 to 21).

30. A decoder device for processing data for display on a screen, the decoder device comprising:

a processor;

a memory; and

an interface for communicating with an encoder device over a link; wherein the encoder device is arranged to:

receive compressed image data over the link, wherein the compressed image data comprises: peripheral image data, compressed using a peripheral compression method; and central image data, compressed using a central compression method, the central compression method being different from the peripheral compression method, wherein the peripheral compression method reduces resolution and favours retaining information relating to edges in the peripheral image data preferentially over information relating to colour;

decompress the compressed image data;

decode the image data;

provide the peripheral image data for display on a peripheral area of the screen; and provide the central image data for display on a central area of the screen.

31. A decoder device according to claim 30, further arranged to perform the method of any of claims 22 to 26 (for claims 24 to 26, when dependent on claims 22 or 23).

32. A system for displaying image data, the system comprising:
an encoder device according to claim 28 or 29;
a decoder device according to claim 30 or 31; and
a screen for displaying the image data.

33. A system according to claim 32, wherein the screen is mounted in a remote device, such as a headset configured to be worn on a viewer's head, the headset being a virtual reality headset or an augmented reality set of glasses.

34. A system according to claim 33, wherein the decoder device is located in the remote device.

35. A system according to any of claims 32 to 34, wherein the encoder device and the decoder device are in communication via a wireless link.