



(19) **United States**

(12) **Patent Application Publication**  
**Plowman et al.**

(10) **Pub. No.: US 2013/0021504 A1**

(43) **Pub. Date: Jan. 24, 2013**

(54) **MULTIPLE IMAGE PROCESSING**

**Publication Classification**

(75) Inventors: **David Plowman**, Great Chesterfield (GB); **Naushir Patuck**, Cambridge (GB); **Benjamin Sewell**, Truro (GB)

(51) **Int. Cl.**  
**H04N 5/228** (2006.01)  
**H04N 5/217** (2011.01)  
(52) **U.S. Cl.** ..... **348/241; 348/222.1; 348/E05.031**

(73) Assignee: **BROADCOM CORPORATION**, Irvine, CA (US)

(57) **ABSTRACT**

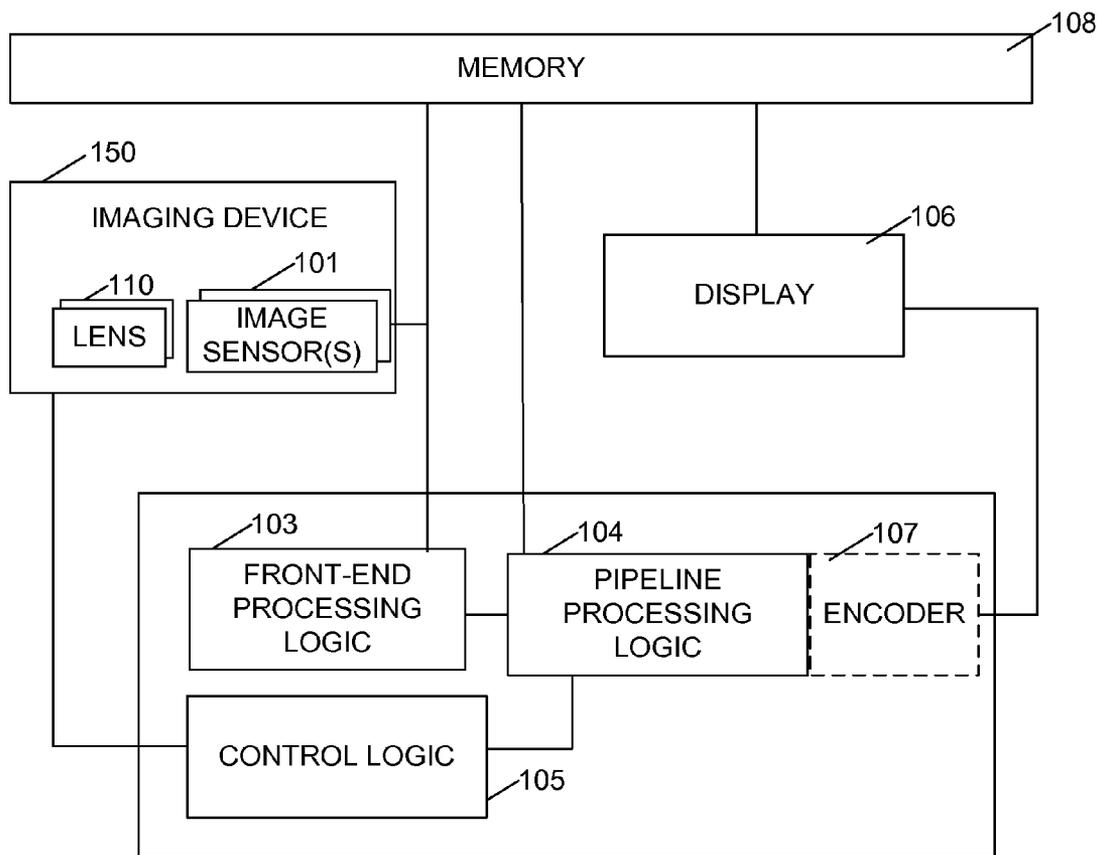
(21) Appl. No.: **13/235,975**

(22) Filed: **Sep. 19, 2011**

**Related U.S. Application Data**

(60) Provisional application No. 61/509,747, filed on Jul. 20, 2011.

Embodiments of the present application automatically utilize parallel image captures in an image processing pipeline. In one embodiment, image processing circuitry concurrently receives first image data to be processed and second image data to be processed, wherein the second image data is processed to aid in enhancement of the first image data.



100

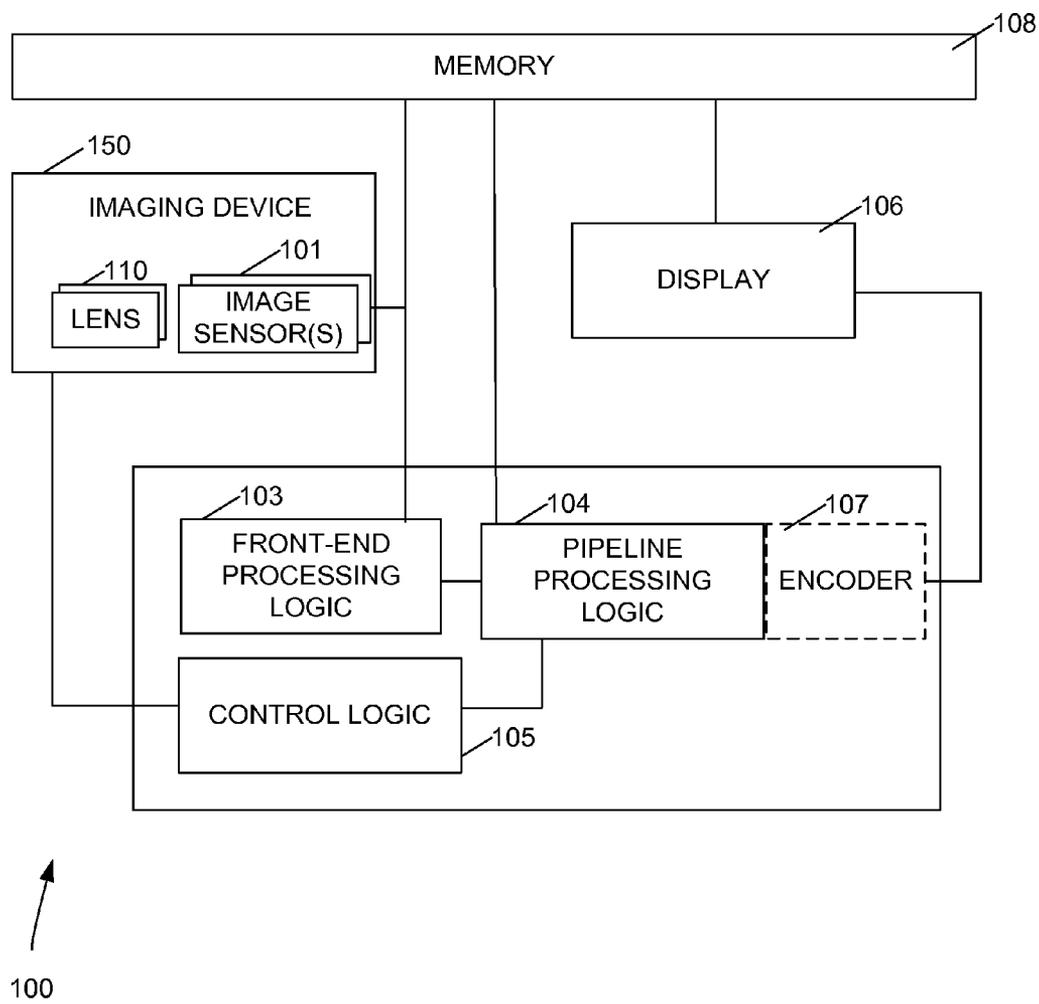


FIG. 1

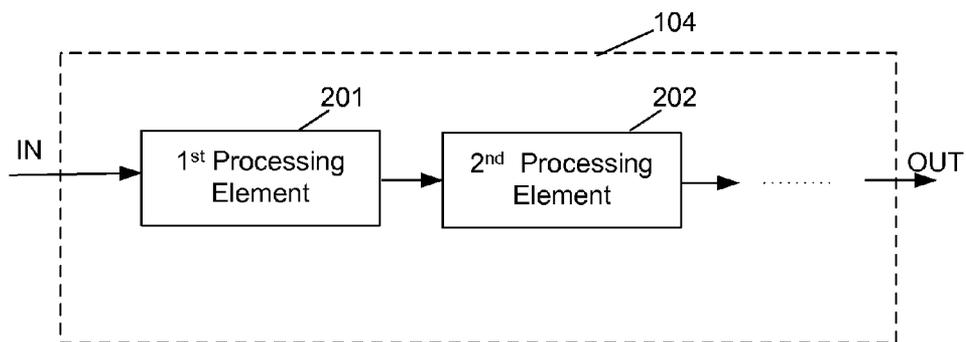


FIG. 2

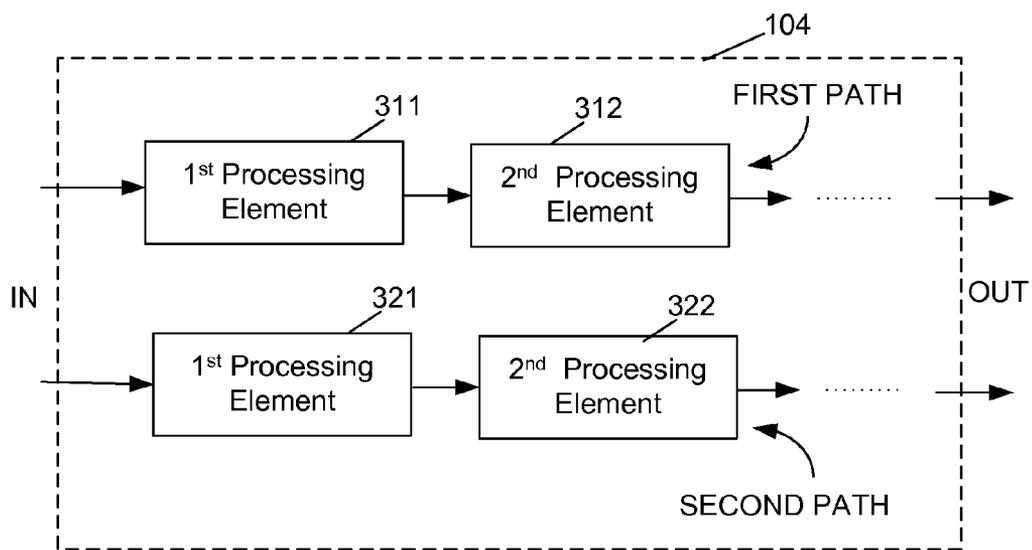


FIG. 3

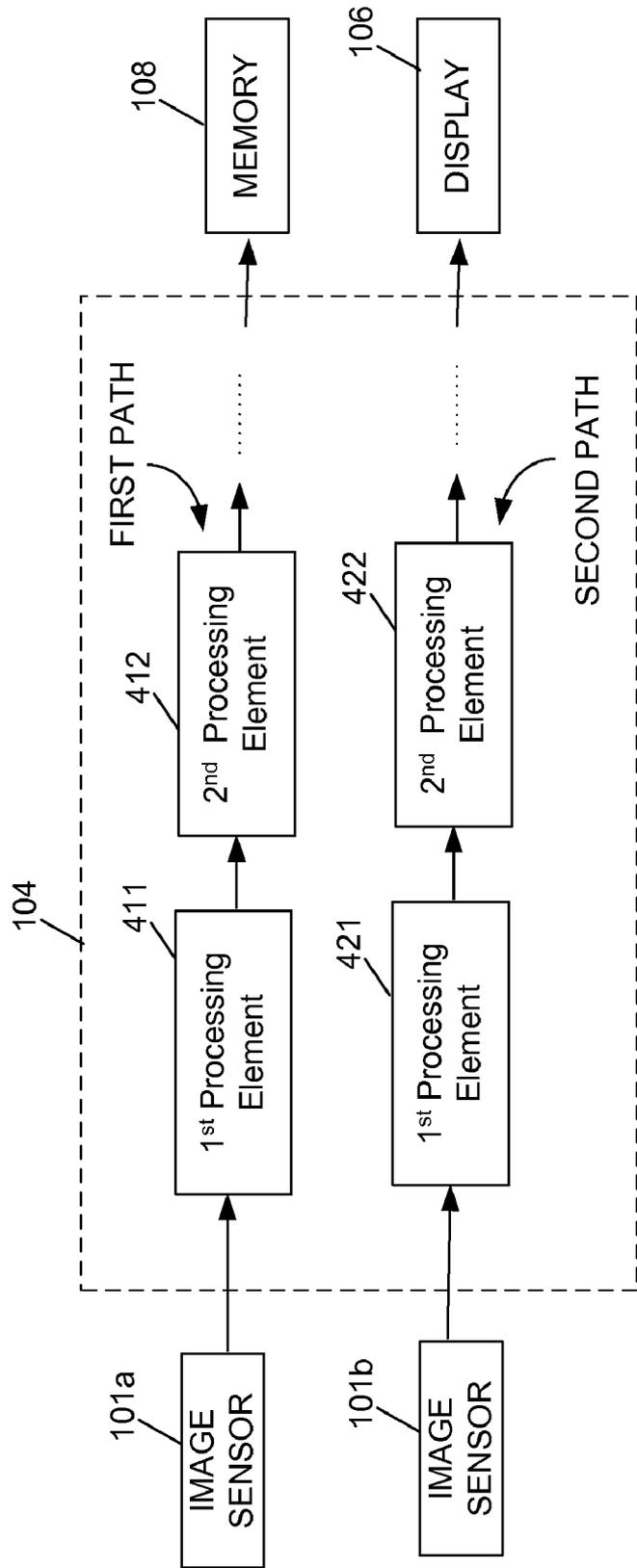


FIG. 4

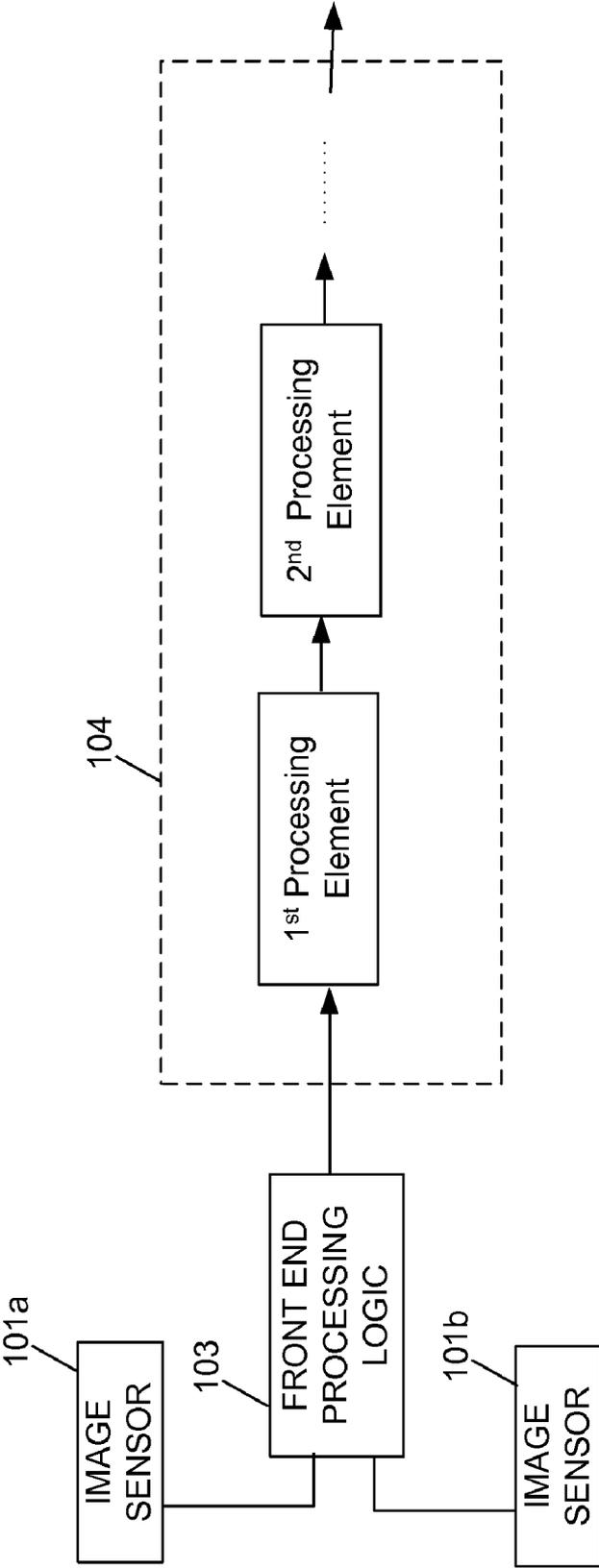


FIG. 5

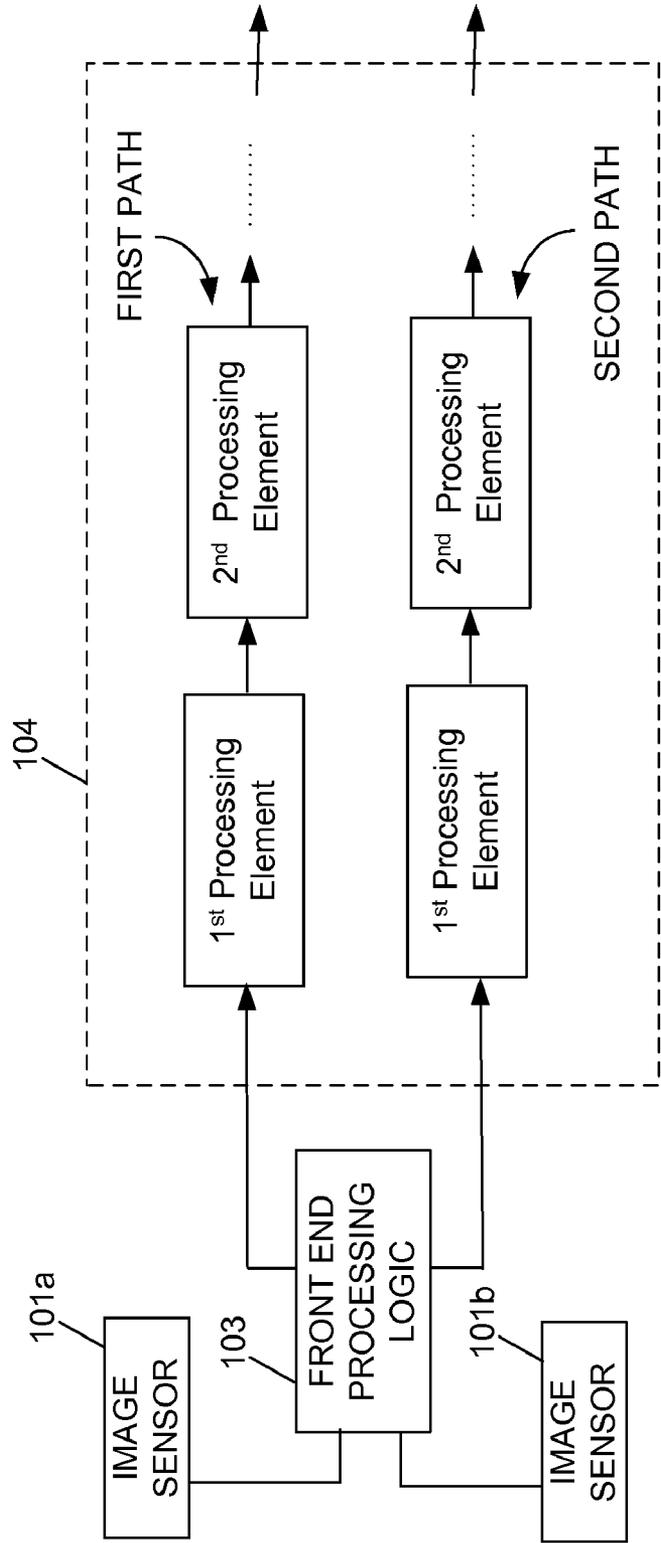


FIG. 6

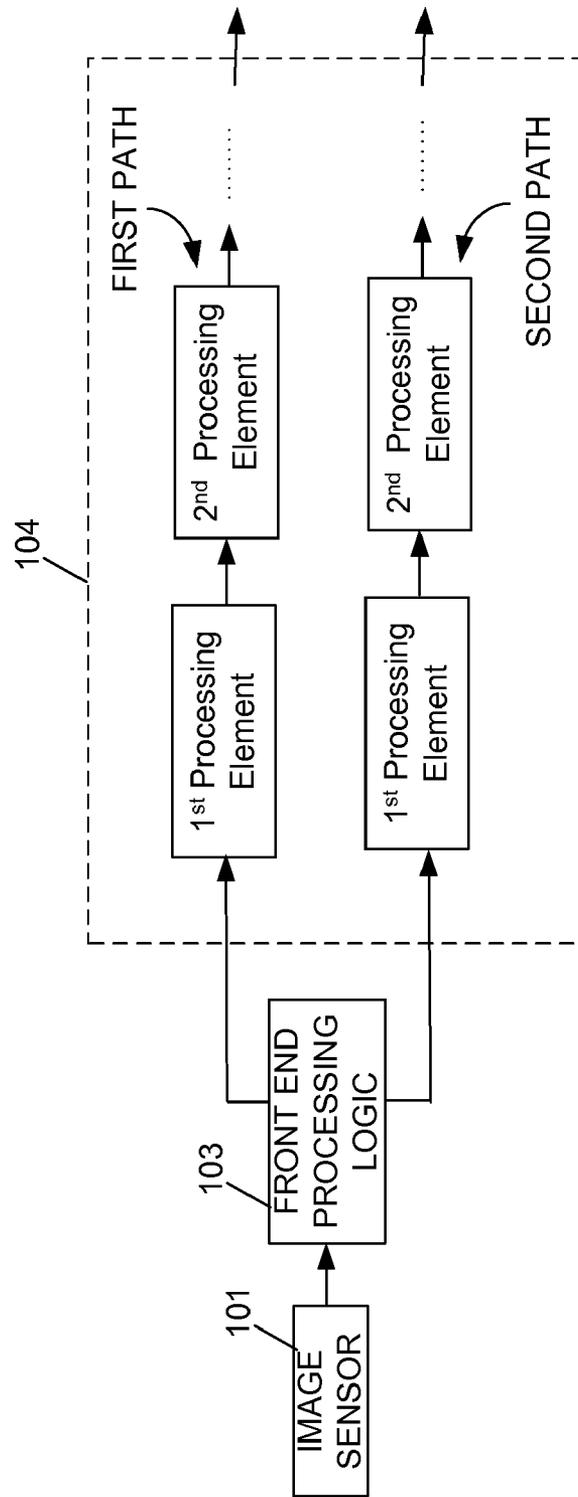


FIG. 7

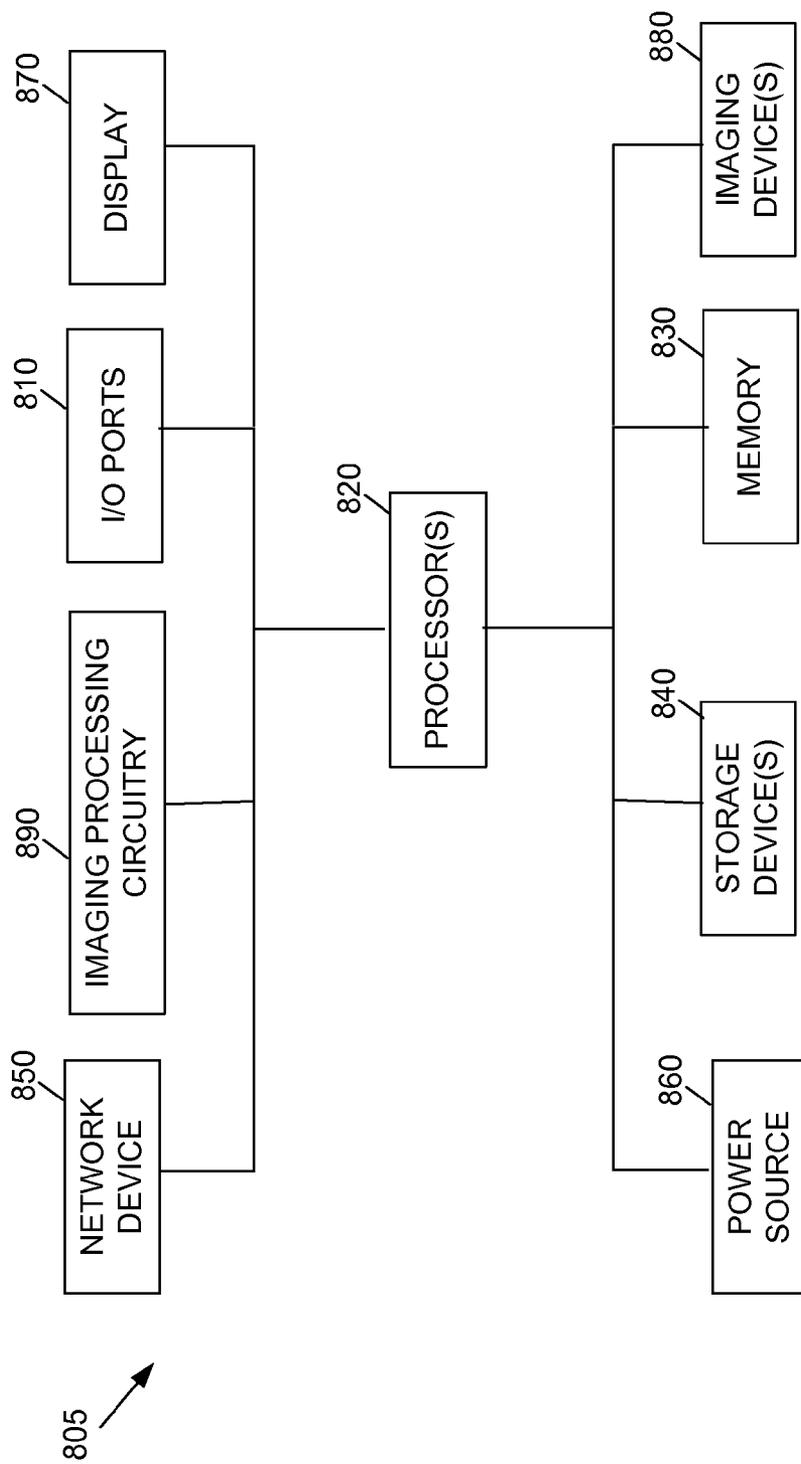


FIG. 8

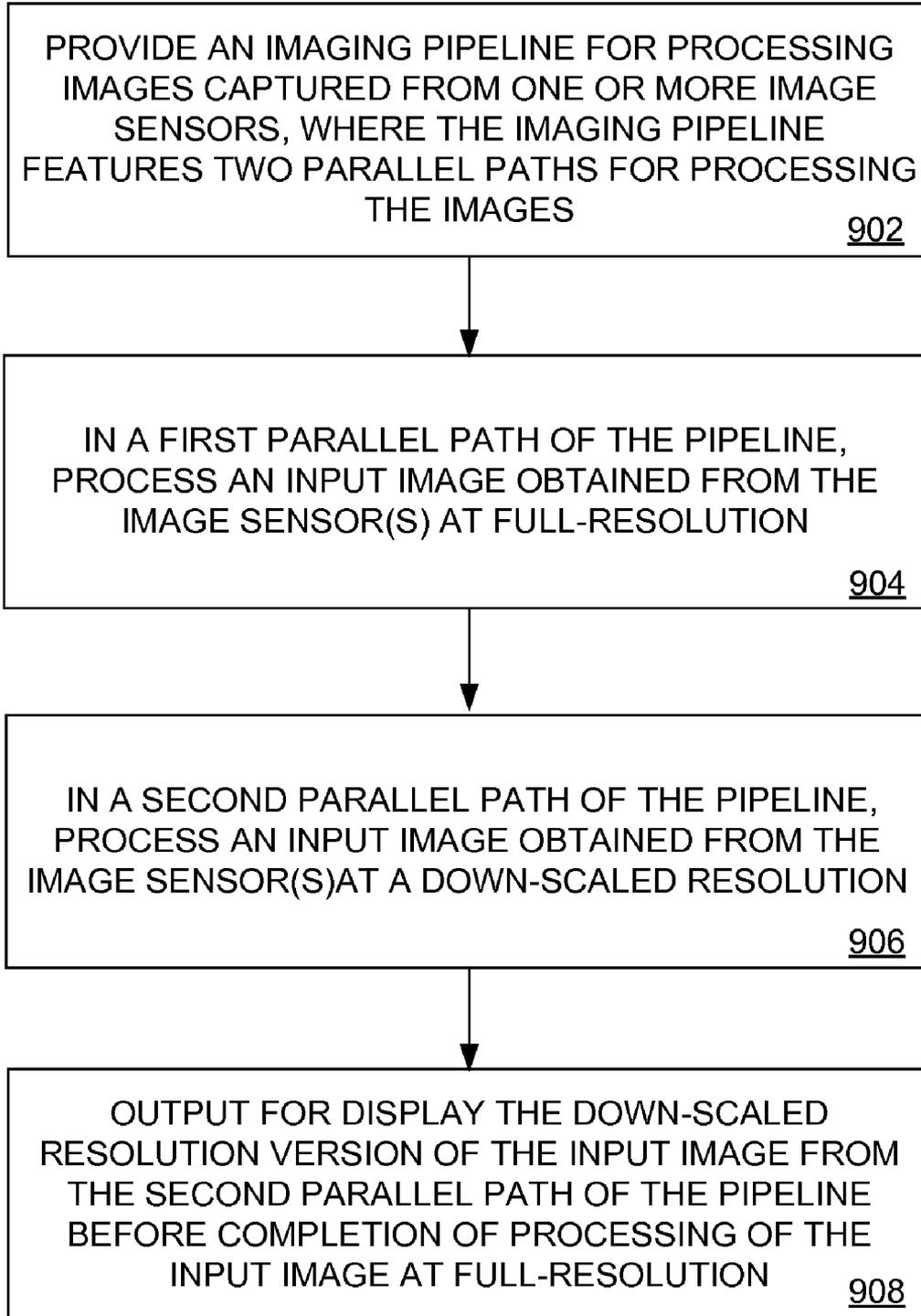


FIG. 9

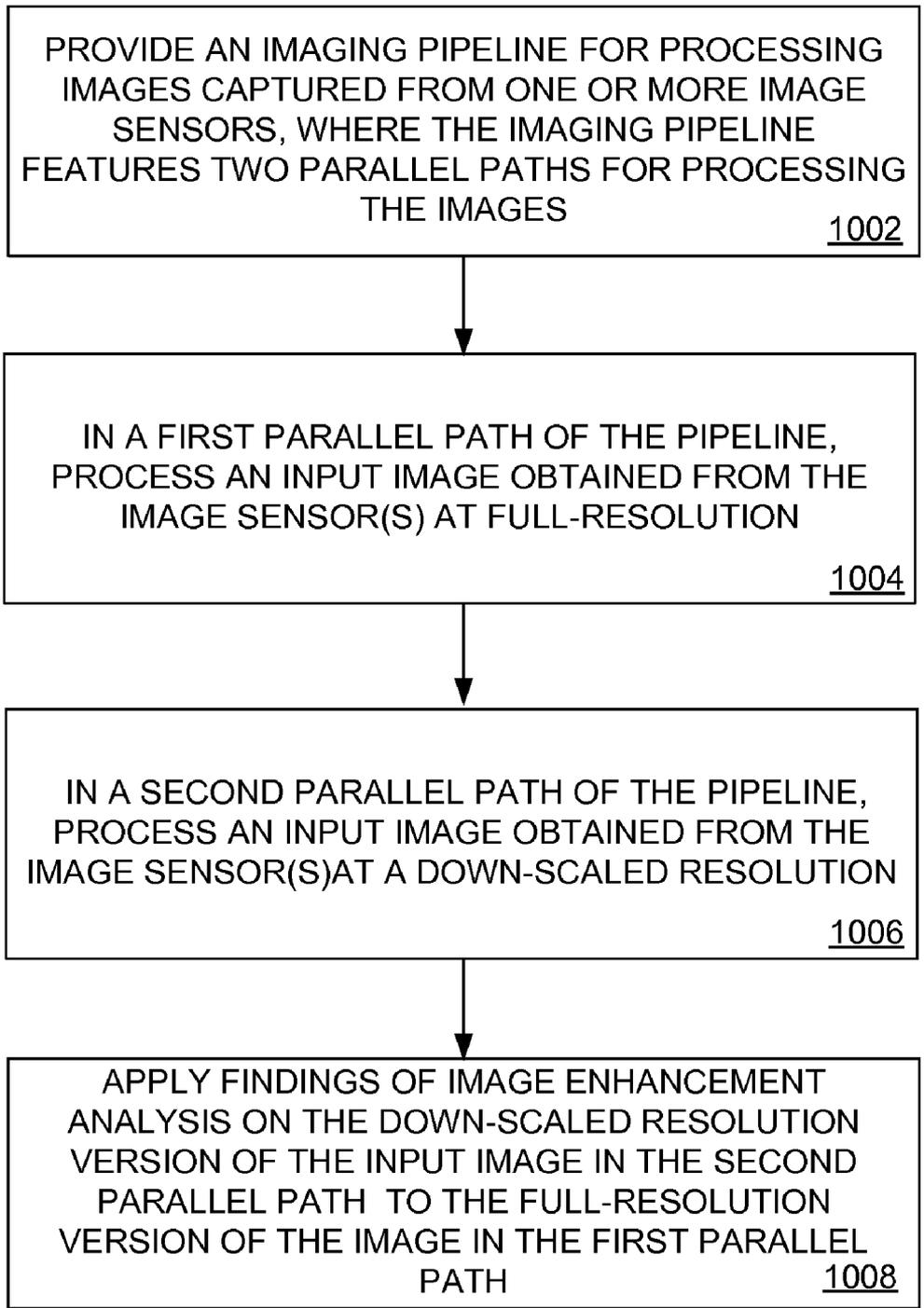


FIG. 10

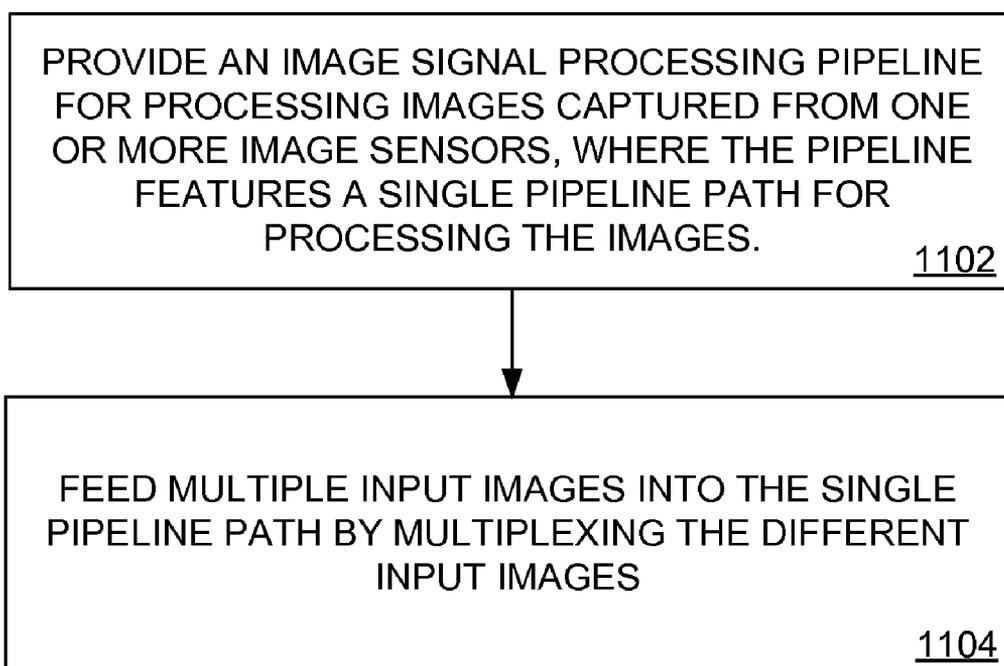


FIG. 11

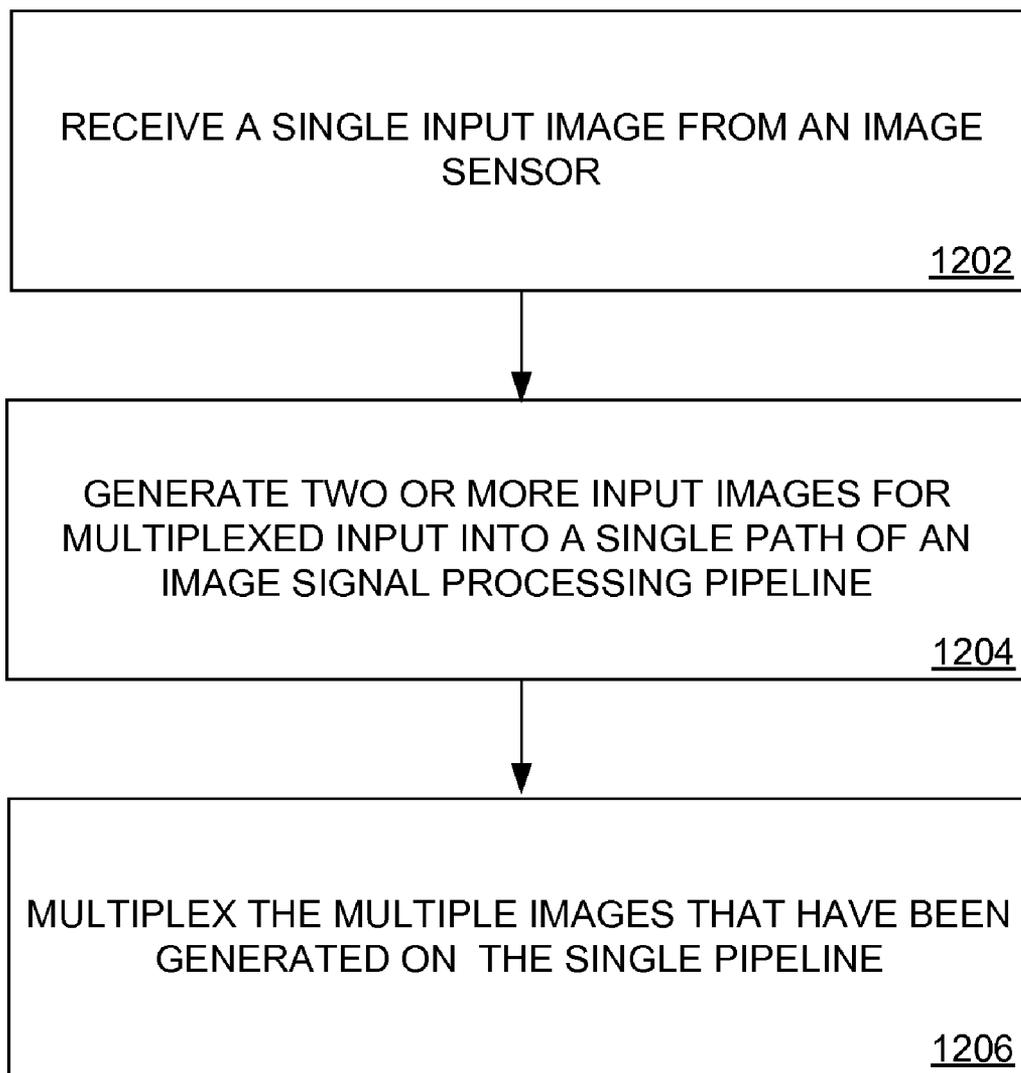


FIG. 12

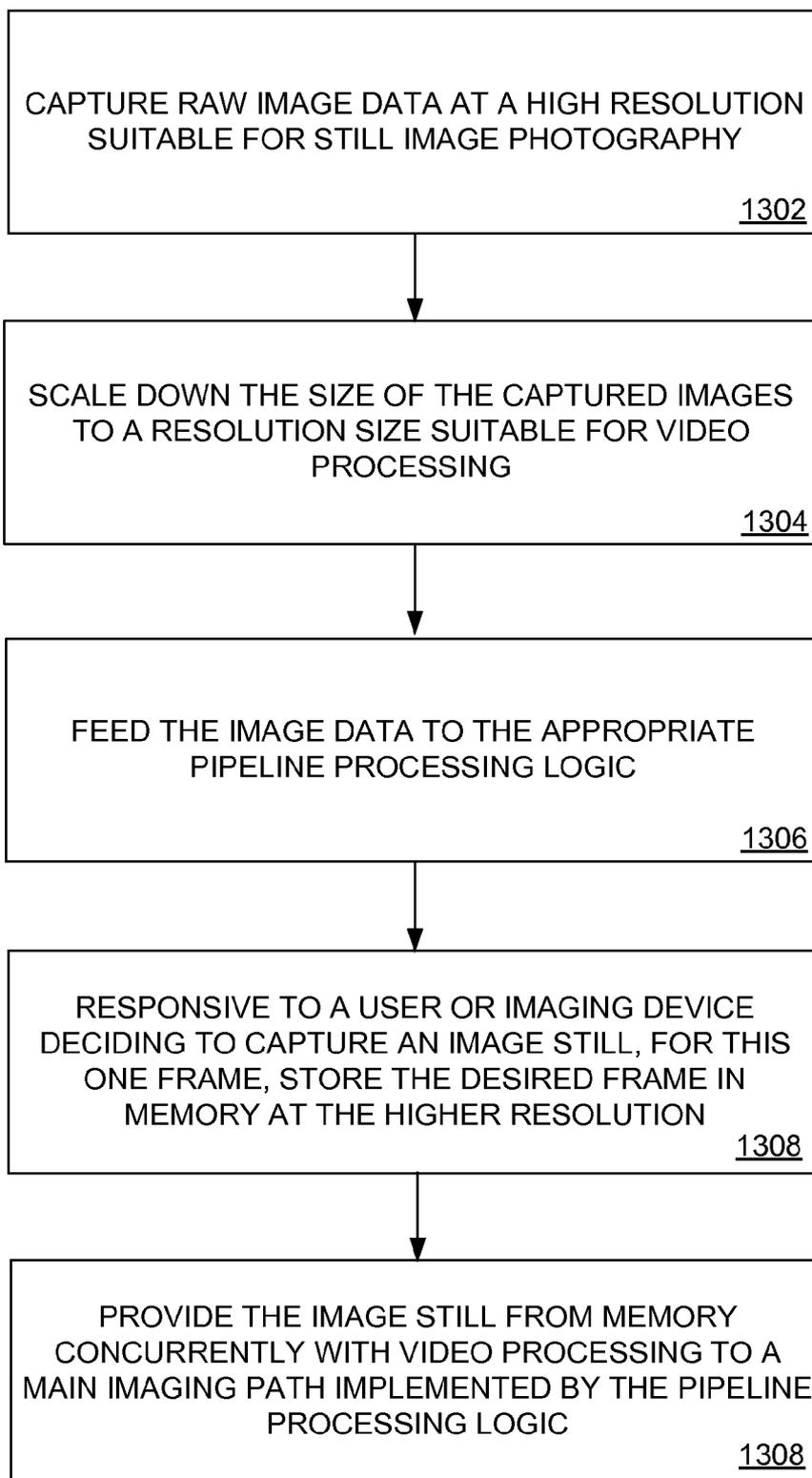


FIG. 13

## MULTIPLE IMAGE PROCESSING

### CROSS-REFERENCE TO RELATED APPLICATION

**[0001]** This application claims priority to copending U.S. provisional application entitled, "Image Capture Device Systems and Methods," having Ser. No. 61/509,747, filed Jul. 20, 2011, which is entirely incorporated herein by reference.

### BACKGROUND

**[0002]** With current cameras, there is a significant delay between the capture of an image and the subsequent display of a framed image to the user via a viewfinder. Accordingly, advances in image processing may allow for improvements, such as shorter latency.

### BRIEF DESCRIPTION OF THE DRAWINGS

**[0003]** Many aspects of the present disclosure can be better understood with reference to the following drawings. The components in the drawings are not necessarily to scale, emphasis instead being placed upon clearly illustrating the principles of the present disclosure. Moreover, in the drawings, like reference numerals designate corresponding parts throughout the several views.

**[0004]** FIG. 1 is a block diagram of one embodiment of an image processing circuitry according to the present disclosure.

**[0005]** FIGS. 2-7 are block diagrams of embodiments of an image signal processing pipeline implemented by the pipeline processing logic from the image processing circuitry of FIG. 1.

**[0006]** FIG. 8 is a block diagram illustrating an embodiment of an electronic device employing the image processing circuitry of FIG. 1.

**[0007]** FIGS. 9-13 are flow chart diagrams depicting various functionalities of embodiments of image processing circuitry of FIG. 1.

### DETAILED DESCRIPTION

**[0008]** This disclosure pertains to a device, method, computer useable medium, and processor programmed to automatically utilize parallel image captures in an image processing pipeline in a digital camera, digital video camera, or other imaging device. One of ordinary skill in the art would recognize that the techniques disclosed may also be applied to other contexts and applications as well.

**[0009]** For cameras in embedded devices, e.g., digital cameras, digital video cameras, mobile phones, personal data assistants (PDAs), tablets, portable music players, and desktop or laptop computers, to produce more visually pleasing images, techniques such as those disclosed herein can improve image quality without incurring significant computational overhead or power costs.

**[0010]** To acquire image data, a digital imaging device may include an image sensor that provides a number of light-detecting elements (e.g., photodetectors) configured to convert light detected by the image sensor into an electrical signal. An image sensor may also include a color filter array that filters light captured by the image sensor to capture color information. The image data captured by the image sensor may then be processed by an image processing pipeline circuitry, which may apply a number of various image process-

ing operations to the image data to generate a full color image that may be displayed for viewing on a display device, such as a monitor.

**[0011]** Referring to FIG. 1, a block diagram of one embodiment of an image processing circuitry 100 is shown for an imaging device 150. The illustrated imaging device 150 may be provided as a digital camera configured to acquire both still images and moving images (e.g., video). The device 150 may include lens(es) 110 and one or more image sensors 101 configured to capture and convert light into electrical signals. By way of example only, the image sensor may include a CMOS (complementary metal-oxide-semiconductor) image sensor (e.g., a CMOS active-pixel sensor (APS)) or a CCD (charge-coupled device) sensor.

**[0012]** In some embodiments, the image processing circuitry 100 may include various subcomponents and/or discrete units of logic that collectively form an image processing "pipeline" for performing each of the various image processing steps. These subcomponents may be implemented using hardware (e.g., digital signal processors or ASICs (application-specific integrated circuits)) or software, or via a combination of hardware and software components. The various image processing operations may be provided by the image processing circuitry 100.

**[0013]** The image processing circuitry 100 may include front-end processing logic 103, pipeline processing logic 104, and control logic 105, among others. The image sensor (s) 101 may include a color filter array (e.g., a Bayer filter) and may thus provide both light intensity and wavelength information captured by each imaging pixel of the image sensors 101 to provide for a set of raw image data that may be processed by the front-end processing logic 103.

**[0014]** In some embodiments, a single lens 110 and a single image sensor 101 may be employed in the image processing circuitry. While in other embodiments, multiple lens 110 and multiple image sensors 101 may be employed, such as for a stereoscopy uses, among others.

**[0015]** The front-end processing logic 103 may also receive pixel data from memory 108. For instance, the raw pixel data may be sent to memory 108 from the image sensor 101. The raw pixel data residing in the memory 108 may then be provided to the front-end processing logic 103 for processing.

**[0016]** Upon receiving the raw image data (from image sensor 101 or from memory 108), the front-end processing logic 103 may perform one or more image processing operations. The processed image data may then be provided to the pipeline processing logic 104 for additional processing prior to being displayed (e.g., on display device 106), or may be sent to the memory 108. The pipeline processing logic 104 receives the "front-end" processed data, either directly from the front-end processing logic 103 or from memory 108, and may provide for additional processing of the image data in the raw domain, as well as in the RGB and YCbCr color spaces, as the case may be. Image data processed by the pipeline processing logic 104 may then be output to the display 106 (or viewfinder) for viewing by a user and/or may be further processed by a graphics engine. Additionally, output from the pipeline processing logic 104 may be sent to memory 108 and the display 106 may read the image data from memory 108. Further, in some implementations, the pipeline processing logic 104 may also include an encoder 107, such as a compression engine, for encoding the image data prior to being read by the display 106.

[0017] The encoder 107 may be a JPEG (Joint Photographic Experts Group) compression engine for encoding still images, or an H.264 compression engine for encoding video images, or some combination thereof. Also, it should be noted that the pipeline processing logic 104 may also receive raw image data from the memory 108.

[0018] The control logic 105 may include a processor 820 (FIG. 8) and/or microcontroller configured to execute one or more routines (e.g., firmware) that may be configured to determine control parameters for the imaging device 150, as well as control parameters for the pipeline processing logic 104. By way of example only, the control parameters may include sensor control parameters, camera flash control parameters, lens control parameters (e.g., focal length for focusing or zoom), or a combination of such parameters for the image sensor(s) 101. The control parameters may also include image processing commands, such as autowhite balance, autofocus, autoexposure, and color adjustments, as well as lens shading correction parameters for the pipeline processing logic 104. The control parameters may further comprise multiplexing signals or commands for the pipeline processing logic 104.

[0019] Referring now to FIG. 2, one embodiment of the pipeline processing logic 104 may perform processes of an image signal processing pipeline by first sending image information to a first process element 201 which may take the raw data produced by the image sensor 101 (FIG. 1) and generate a digital image that will be viewed by a user or undergo further processing by a downstream process element. Accordingly, the image signal processing pipeline may be considered as a series of specialized algorithms that adjusts image data in real-time and is often implemented as an integrated component of a system-on-chip (SoC) image processor. With an image signal processing pipeline implemented in hardware, front-end image processing can be completed without placing any processing burden on the main application processor 820 (FIG. 8).

[0020] In one embodiment, the first process element 201 of an image signal processing pipeline could perform a particular image process such as noise reduction, defective pixel detection/correction, lens shading correction, lens distortion correction, demosaicing, image sharpening, color uniformity, RGB (red, green, blue) contrast, saturation boost process, etc. As discussed above, the pipeline may include a second process element 202. In one embodiment, the second process element 202 could perform a particular and different image process such as noise reduction, defective pixel detection/correction, lens shading correction, demosaicing, image sharpening, color uniformity, RGB contrast, saturation boost process etc. The image data may then be sent to additional element(s) of the pipeline as the case may be, saved to memory 108 (FIG. 1), and/or input for display 106 (FIG. 1).

[0021] Referring next to FIG. 3, in one embodiment, the image signal processing pipeline performed by pipeline processing logic 104 contains parallel paths instead of a single linear path. For example, the parallel paths may provide a first path and a second path. Further, in one embodiment, the first path comprises a main processing path and the second path comprises a supplemental processing path. Therefore, while raw image data is being processed in the first path to generate a high-resolution image output suitable for storage, the raw image data is processed in the second and parallel path to generate a lower resolution image that can be generated more quickly (as compared to the first path) and be displayed in the

camera viewfinder or display 106. It may be that the second path contains fewer stages or elements 321, 322 than the first path. Alternatively, the first path may contain the same number of or less number of stages or elements 311, 312 as compared to the second path. Further, the second path may involve resolution down-conversion of the image to lessen the amount of pixels that need to be processed during image processing, such as for image analysis, in the pipeline.

[0022] The benefits of the parallel paths may apply to still images as well as video images captured by the image sensor (s) 101 (FIG. 1). It is noted that some embodiments of the pipeline processing logic 104 utilizes a single image sensor 101 that provides raw data to the first and second paths, where the first path may process the raw data relatively carefully and more slowly than the second path that can generate an image available to be previewed more quickly.

[0023] Use of parallel paths in the image signal processing pipeline may enable processing of multiple image data simultaneously while maximizing final image quality. Additionally, each stage in the pipeline may begin processing as soon as image data is available so the entire image does not have to be received from the previous sensor or stage before processing is started.

[0024] In an alternative embodiment, multiple imagers or image sensors 101 may be utilized, as shown in FIG. 4. For example, one imager or sensor 101a may provide raw data at a lower resolution than a second image sensor 101b, where the lower resolution raw data feeds a pipeline path to the display 106 and the higher resolution data feeds a path used for encoding and/or for storage in memory 108.

[0025] Further, in some embodiments, a secondary or supplemental image may be used in image analysis that can help subsequent image analysis operations for the main image. As an example, a secondary image at a smaller size or resolution than the main image might undergo facial recognition algorithms (or other object recognition algorithm) and output of positive results may be used to identify facial structures (or other objects) in the main image. Therefore, the secondary image may be produced in a format that is more suited for some of the applicable states or processing elements in its path. Accordingly, processing elements 411, 412, may be divided up between elements that are suited for the main image and processing elements 421, 422 that are suited for the secondary image. Accordingly, a secondary image may be initially processed, such as being made smaller or scaled, for the benefit of downstream elements. As an example, the path of the secondary image may contain a noise filtering element due to a downstream element needed for the secondary image to have undergone noise reduction. The different paths or elements in the different paths may also use different imaging formats. For example, one of the paths may use an integral image format whereas a standard image format is used in the other path. Accordingly, downstream elements in the integral image path may need an integral image format as opposed to a standard image format and vice versa.

[0026] In some embodiments, the images generated by the first and second paths may be stored in memory 108 and made available for subsequent use by other procedures and elements that follow. Accordingly, in one embodiment, while a main image is being processed in a main path of the pipeline, another image which might be downsized or scaled of that image or a previous image may be read by the main path. This may enable more powerful processing in the pipeline, such as during noise filtering.

[0027] For example, during noise filtering, for any given pixel being processed, neighboring pixels are analyzed. This process of denoising the pixel may have a stronger effect with the more pixels that are able to be analyzed further away from the pixel being processed. Due to hardware constraints, such as memory buffer size used by processing logic, there is a limit in how far away from the current pixel that the process can analyze neighboring pixels.

[0028] Accordingly, in one embodiment, a downsampled version of the main image is generated in a second path and the noise filter in the main path reads the downsampled version of the image and stores those pixels for noise analysis. Since there are the same number of line buffers but the image is downsampled, this effectively allows the noise filter to see further away in the original image because the second image is at a reduced scale.

[0029] Accordingly, another embodiment utilizes a downsampled version of an image to assist in dynamic range optimization processing. By having available a downsampled version of an image alongside a full resolution image in memory 108, a dynamic range optimization process is provided a way to see further away from a current pixel than would be available by only considering the full resolution image. In a similar manner, a high dynamic range imaging process or element also reads a downsampled version of a main image to see further away from the current pixel.

[0030] Referring back to FIG. 1, in one embodiment, raw image data (from an image sensor 101) may be provided to the front-end processing logic 103 and processed on a pixel-by-pixel basis in a number of formats. For example, in one embodiment, raw pixel data received by the front-end processing logic 103 may be up-sampled for image processing purposes. In another embodiment, raw image or pixel data may be down-sampled or scaled. As will be appreciated, down-sampling of image data may reduce hardware size (e.g., area) and also reduce processing/computational complexity.

[0031] In some embodiments, the front-end processing logic 103 generates two distinct kinds of images for the pipeline processing logic 104. As an example, the imaging device 150 may be capturing video images and the user or the device itself determines to also capture a still image in addition to the video or moving images. A problem to overcome with this task in conventional cameras is that the video images are being generated at a resolution that is less than desired for still images. A potential solution would be to record the video images at the higher resolution desired for the still image, but this would require the pipeline processing logic 104 to undergo processing of the higher resolution video images. However, it is difficult to encode video at a high resolution (e.g., 8 megapixels) and it is also impractical, since video images do not necessarily require a very high resolution.

[0032] Accordingly, one embodiment of the present disclosure captures the raw image data by the sensor 101 at the higher resolution suitable for still image photography. Then, the front-end pipeline processing logic 103 scales down the size of the captured images to a resolution size suitable for video processing before feeding the image data to the appropriate pipeline processing logic 104. When the user or the imaging device 150 decides to capture an image still, for this one frame, the front-end pipeline processing logic 103 will receive instructions from the control logic 105 and store the desired frame in memory 108 at the higher resolution. Further, in one embodiment, although a main imaging path of the

pipeline is handling the video processing, as processing time allows, the main imaging path can be provided the still image from memory 108.

[0033] Accordingly, in one embodiment, the video processing is assigned a higher priority than the still image processing by the pipeline processing logic 104. In such an embodiment, the pipeline processing logic 104 features a single pipeline for processing captured images but has the capability to multiplex the single pipeline between different input images. Therefore, the single pipeline may switch from processing an image or series of images having a high priority to an image or series of images having a lower priority as processing time allows.

[0034] Multiplexing of the imaging pipeline is also implemented in an embodiment utilizing multiple image sensors 101. For example, consider a stereoscopic image device that delivers a left image and a right image of an object to a single image pipeline, as represented in FIG. 5. The single image pipeline in pipeline processing logic 104 can therefore be multiplexed between the left and right images that are being input in parallel to the image signal processing pipeline so that the pipeline is shared. Instead of processing one of the images in its entirety after the other has been processed in its entirety, the images can be processed concurrently by switching processing of the images between one another as processing time allows by front-end processing logic 103. This reduces latency by not delaying processing of an image until completion of the other image, and processing of the two images will finish more quickly.

[0035] Alternatively, one embodiment utilizes multiple image sensors 101 that produce multiple inputs for the pipeline processing logic 104. Referring now to FIG. 6, in one scenario, one of the image sensors 101b may capture a low resolution image that is fed as a preview of an image recently captured, where the other image sensor 101a captures a high resolution image of the subject of the picture that is processed in parallel. Otherwise, the low resolution image may be used for framing a shot to be captured, where the subsequent captured shot or image is at a higher resolution and may undergo additional processing. Therefore, this embodiment features an imaging device with two fully parallel image capture and processing pipeline paths.

[0036] Further, in some embodiments, a single image sensor 101 is utilized to capture image information and provide the information to the front-end processing logic 103, whereby the front-end processing logic 103 may generate two input images for parallel paths in the pipeline processing logic 104 (as represented in FIG. 7). Also, in some embodiments, a single image sensor 101 is utilized to capture image information and provide the information to the front-end processing logic 103, whereby the front-end processing logic 103 may generate two input images for multiplexed input into a single path of the pipeline processing logic 104 (as represented in FIG. 6).

[0037] Keeping the above points in mind, FIG. 8 is a block diagram illustrating an example of an electronic device 805 that may provide for the processing of image data using one or more of the image processing techniques briefly mentioned above. The electronic device 805 may be any type of electronic device, such as a laptop or desktop computer, a mobile phone, tablet, a digital media player, or the like, that is configured to receive and process image data, such as data acquired using one or more image sensing components.

[0038] Regardless of its form (e.g., portable or non-portable), it should be understood that the electronic device 805 may provide for the processing of image data using one or more of the image processing techniques briefly discussed above, among others. In some embodiments, the electronic device 805 may apply such image processing techniques to image data stored in a memory 830 of the electronic device 805. In further embodiments, the electronic device 805 may include one or more imaging devices 880, such as an integrated or external digital camera, configured to acquire image data, which may then be processed by the electronic device 805 using one or more of the above-mentioned image processing techniques.

[0039] As shown in FIG. 8, the electronic device 805 may include various internal and/or external components which contribute to the function of the device 805. Those of ordinary skill in the art will appreciate that the various functional blocks shown in FIG. 8 may comprise hardware elements (including circuitry), software elements (including computer code stored on a computer readable medium) or a combination of both hardware and software elements. For example, in the presently illustrated embodiment, the electronic device 805 may include input/output (I/O) ports 810, one or more processors 820, memory device 830, non-volatile storage 840, networking device 850, power source 860, and display 870. Additionally, the electronic device 10 may include one or more imaging devices 880, such as a digital camera, and image processing circuitry 890. As will be discussed further below, the image processing circuitry 890 may be configured to implement one or more of the above-discussed image processing techniques when processing image data. As can be appreciated, image data processed by image processing circuitry 890 may be retrieved from the memory 830 and/or the non-volatile storage device(s) 840, or may be acquired using the imaging device 880.

[0040] Before continuing, it should be understood that the system block diagram of the device 805 shown in FIG. 8 is intended to be a high-level control diagram depicting various components that may be included in such a device 805. That is, the connection lines between each individual component shown in FIG. 1 may not necessarily represent paths or directions through which data flows or is transmitted between various components of the device 805. Indeed, as discussed below, the depicted processor(s) 820 may, in some embodiments, include multiple processors, such as a main processor (e.g., CPU), and dedicated image and/or video processors. In such embodiments, the processing of image data may be primarily handled by these dedicated processors, thus effectively offloading such tasks from a main processor (CPU).

[0041] Referring next to FIG. 9, shown is a flowchart that provides one example of the operation of a portion of the image processing circuitry 100 according to various embodiments. It is understood that the flowchart of FIG. 9 provides merely an example of the many different types of functional arrangements that may be employed to implement the operation of the portion of the image processing circuitry 100 as described herein. As an alternative, the flowchart of FIG. 9 may be viewed as depicting an example of steps of a method implemented in the electronic device 805 (FIG. 8) according to one or more embodiments.

[0042] Beginning in step 902, imaging processing circuitry 100 provides an imaging pipeline for processing images captured from one or more image sensors 101, where the imaging single processing pipeline features two parallel paths for pro-

cessing the images. As described in step 904, in a first parallel path of the pipeline, an input image obtained from the image sensor(s) 101 is processed at full-resolution. Additionally, in a second parallel path of the pipeline, an input image obtained from the image sensor(s) 101 is processed at a down-scaled resolution, as depicted in step 906. The down-scaled resolution version of the input image is output from the second parallel path of the pipeline before completion of processing of the input image at full-resolution and is provided for display, in step 908.

[0043] Next, referring to FIG. 10, shown is a flowchart that provides an additional example of the operation of a portion of the image processing circuitry 100 according to various embodiments. Beginning in step 1002, imaging processing circuitry 100 provides an imaging pipeline for processing images captured from one or more image sensors 101, where the image signal processing pipeline features two parallel paths for processing the images. As described in step 1004, in a first parallel path of the pipeline, an input image obtained from the image sensor(s) 101 is processed at full-resolution. Additionally, in a second parallel path of the pipeline, an input image obtained from the image sensor(s) 101 is processed at a down-scaled resolution, as depicted in step 1006. The down-scaled resolution version of the input image undergoes image enhancement analysis in the second parallel path that is applied to the full-resolution version of the image in the first parallel path, in step 1008. In particular, pixels are able to be analyzed in the down-scaled resolution version of the input image that may not be able to be analyzed as efficiently in the full-resolution version of the input image due to buffer limitations or other hardware restraints. In various embodiments, the type of image enhancement analysis may include noise filtering, dynamic range optimization, high dynamic range imaging, facial or object recognition, among others.

[0044] In FIG. 11, a flow chart is shown that provides an additional example of the operation of a portion of the image processing circuitry 100 according to various embodiments. Beginning in step 1102, imaging processing circuitry 100 provides an image signal processing pipeline for processing images captured from one or more image sensors 101, where the pipeline features a single pipeline path for processing the images. As described in step 1104, multiple input images may be fed into the single pipeline path by multiplexing the different images by front-end circuitry (e.g., front-end processing logic 103). For example, consider a stereoscopic image device that delivers a left image and a right image of an object to a single image pipeline, as represented in FIG. 5. The single image pipeline in pipeline processing logic 104 can therefore be multiplexed between the left and right images that are being input in parallel to the pipeline via the front-end circuitry. Instead of processing one of the images in its entirety after the other has been processed in its entirety, the images can be processed concurrently by switching processing of the images between one another as processing time allows by front-end processing circuitry.

[0045] Further, in FIG. 12, a flow chart is shown that provides an additional example of the operation of a portion of the image processing circuitry 100 according to various embodiments. Beginning in step 1202, front-end processing circuitry can receive a single input image from an image sensor 101. In step 1204, the front-end processing circuitry may then generate two or more input images for multiplexed input into a single path of an image signal processing pipeline of pipeline processing logic 104 (as represented in FIG. 6).

The single pipeline in pipeline processing logic 104 can therefore be multiplexed between the multiple images that have been generated by the front-end circuitry, in step 1206. Instead of processing one of the images in its entirety after the other has been processed in its entirety, the images can be processed concurrently by switching processing of the images between one another as processing time allows by front-end processing circuitry.

[0046] Next, in FIG. 13, a flow chart is shown that provides an additional example of the operation of a portion of the image processing circuitry 100 according to various embodiments. Accordingly, one embodiment of the present disclosure captures the raw image data by the sensor 101 at a high resolution suitable for still image photography, in step 1302. Then, the front-end pipeline processing logic 103 scales down the size of the captured images to a resolution size suitable for video processing, in step 1304, before feeding the image data to the appropriate pipeline processing logic 104, in step 1306. When the user or the imaging device 150 decides to capture an image still, for this one frame, the front-end pipeline processing logic 103 will receive instructions from the control logic 105 and store the desired frame in memory 108 at the higher resolution, in step 1308. Further, in one embodiment, although a main imaging path of the pipeline is handling the video processing, as processing time allows, the main imaging path can be provided the still image from memory 108, in step 1310.

[0047] Any process descriptions or blocks in flow charts should be understood as representing modules, segments, or portions of code which include one or more executable instructions for implementing specific logical functions or steps in the process, and alternate implementations are included within the scope of embodiments of the present disclosure in which functions may be executed out of order from that shown or discussed, including substantially concurrently or in reverse order, depending on the functionality involved, as would be understood by those reasonably skilled in the art.

[0048] In the context of this document, a “computer readable medium” can be any means that can contain, store, communicate, or transport the program for use by or in connection with the instruction execution system, apparatus, or device. The computer readable medium can be, for example but not limited to, an electronic, magnetic, optical, electromagnetic, infrared, or semiconductor system, apparatus, or device. More specific examples (a nonexhaustive list) of the computer readable medium would include the following: an electrical connection (electronic) having one or more wires, a portable computer diskette (magnetic), a random access memory (RAM) (electronic), a read-only memory (ROM) (electronic), an erasable programmable read-only memory (EPROM or Flash memory) (electronic), an optical fiber (optical), and a portable compact disc read-only memory (CDROM) (optical). In addition, the scope of certain embodiments includes embodying the functionality of the embodiments in logic embodied in hardware or software-configured mediums.

[0049] It should be emphasized that the above-described embodiments are merely possible examples of implementations, merely set forth for a clear understanding of the principles of the disclosure. Many variations and modifications may be made to the above-described embodiment(s) without departing substantially from the spirit and principles of the disclosure. All such modifications and variations are intended

to be included herein within the scope of this disclosure and protected by the following claims.

Therefore, having thus described various embodiments, at least the following is claimed:

1. An image capture device, comprising:
  - a hardware processor; and
  - an image processing circuitry that concurrently receives first image data to be processed and second image data to be processed, wherein the second image data is processed to aid in enhancement of the first image data.
2. The image capture device of claim 1, wherein the first image data comprises video.
3. The image capture device of claim 1, wherein the image processing circuitry comprises an imaging pipeline for processing a plurality of image data captured from at least one image sensor, where the imaging pipeline features two parallel paths for processing the plurality of image data.
4. The image capture device of claim 3, wherein the first image data is processed at full-resolution on a first parallel path and the second image data is processed at a down-scaled resolution on a second parallel path.
5. The image capture device of claim 1, further comprising:
  - a viewfinder display that displays output of the second image data from the image processing circuitry before completion of processing of the first image data by the image processing circuitry.
6. The image capture device of claim 1, wherein the enhancement of the first image data comprises at least one of noise filtering, dynamic range optimization, and high dynamic range imaging.
7. The image capture device of claim 1, wherein the image processing circuitry comprises an imaging pipeline for processing a plurality of image data captured from at least one image sensor, where the imaging pipeline features a singular path for processing the plurality of image data by multiplexing the plurality of image data between the singular path.
8. The image capture device of claim 1, further comprising:
  - a first image sensor that recorded the first image data; and
  - a second image sensor that recorded the second image data.
9. An image processing method, comprising:
  - receiving a first image data to be processed and a second image data to be concurrently processed; and
  - processing the second image data to aid in enhancement of the first image data.
10. The image processing method of claim 9, wherein the first image data comprises video.
11. The image processing method of claim 9, wherein the first image data and the second image data are processed in a singular pipeline path, the method further comprising:
  - multiplexing the first image data and the second image data between the singular path.
12. The image processing method of claim 9, wherein the first image data and the second image data are processed in parallel pipeline paths, wherein the first image data is processed at full-resolution on a first parallel path and the second image data is processed at a down-scaled resolution on a second parallel path.
13. The image processing method of claim 12, further comprising:
  - displaying output of the second image data from the second parallel path before completion of processing of the first image data at the first parallel path.

**14.** The image processing method of claim **9**, wherein the enhancement of the first image data comprises at least one of noise filtering, and dynamic range optimization, and high dynamic range imaging.

**15.** A computer readable medium having an image processing program, when executed by a hardware processor, causes the hardware processor to:

receive a first image data to be processed and a second image data to be concurrently processed; and  
process the second image data to aid in enhancement of the first image data.

**16.** The computer readable medium of claim **15**, wherein the first image data comprises video.

**17.** The computer readable medium of claim **15**, wherein the first image data and the second image data are processed in a singular pipeline path, the image processing program further causing the hardware processor to:

multiplex the first image data and the second image data between the singular path.

**18.** The computer readable medium of claim **15**, wherein the first image data and the second image data are processed in parallel pipeline paths, wherein the first image data is processed at full-resolution on a first parallel path and the second image data is processed at a down-scaled resolution on a second parallel path.

**19.** The computer readable medium of claim **18**, the image processing program further causing the hardware processor to:

display output of the second image data from the second parallel path before completion of processing of the first image data at the first parallel path.

**20.** The computer readable medium of claim **15**, wherein the enhancement of the first image data comprises at least one of noise filtering, and dynamic range optimization, and high dynamic range imaging.

\* \* \* \* \*