(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
24 June 2021 (24.06.2021)

WIPO | PCT

(10) International Publication Number
WO 2021/126268 A1

(51) International Patent Classification:
G06K 9/62 (2006.01) *G06N 3/02* (2006.01)

(21) International Application Number:
PCT/US2019/068047

(22) International Filing Date:
20 December 2019 (20.12.2019)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicants: **HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.** [US/US]; 10300 Energy Drive, Spring, Texas 77389 (US). **UNIVERSIDADE FEDERAL DE PERNAMBUCO** [BR/BR]; Av. Prof. Moraes, Rêgo, 1235-Cidade Universitária, 50740-560 Recife (BR).

(72) Inventors: **VINICIUS MACHADO DE SOUZA, Thiago**; Centro de Informática (CIn) da UFPE, Av. Jornalista Anibal Fernandes, s/n, Cidade Universitária (Campus Recife), 50740-560 Recife (BR). **PAULO MAGALHÃES SIMÕES, Francisco**; Centro de Informática (CIn) da UFPE, Av. Jornalista Anibal Fernandes, s/n, Cidade Universitária (Campus Recife), 50740-560 Recife (BR). **DE MENEZES CHAVES, Thiago**; Centro de Informática (CIn) da UFPE, Av. Jornalista Anibal Fernandes, s/n, Cidade Universitária (Campus Recife), 50740-560 Recife (BR). **DE CASTRO FELIX, Heitor**; Centro de Informática (CIn) da UFPE, Av. Jornalista Anibal Fernandes, s/n, Cidade Universitária (Campus Recife), 50740-560 Recife (BR). **VALENÇA ROCHA MARTINS DE ALBUQUERQUE, Lucas**; Centro de Informática (CIn) da UFPE, Av. Jornalista Anibal Fernandes, s/n, Cidade Universitária (Campus Recife), 50740-560 Recife (BR). **BATISTA DA CUNHA, Kelvin**; Centro de Informática (CIn) da UFPE, Av. Jornalista Anibal Fernandes, s/n, Cidade Universitária (Campus Recife), 50740-560 Recife (BR). **ALVES ROBERTO, Rafael**; Centro de Informática (CIn) da UFPE, Av. Jornalista Anibal Fernandes, s/n, Cidade Universitária (Campus Recife), 50740-560 Recife (BR). **TEIXEIRA, João Marcelo Xavier Natário**; Centro de Informática (CIn) da UFPE, Av. Jornalista Anibal Fernandes, s/n, Cidade Universitária (Campus Recife), 50740-560 Recife (BR). **SILVA DO MONTE LIMA, João Paulo**; Centro de Informática (CIn) da UFPE, Av. Jornalista Anibal Fernandes, s/n, Cidade Universitária (Campus Recife), 50740-560 Recife (BR). **TEICHRIEB, Veronica**; Centro de Informática (CIn) da UFPE, Av. Jornalista Anibal Fernandes, s/n, Cidade Universitária (Campus Recife), 50740-560 Recife (BR). **POLESE COSSIO, Lucio**; Av. Ipiranga, 6681, Bld. 45C, Predios 5-6, 90619-900 Porto Alegre (BR).

(54) **Title:** NEURAL NETWORKS TO PROVIDE IMAGES TO RECOGNITION ENGINES

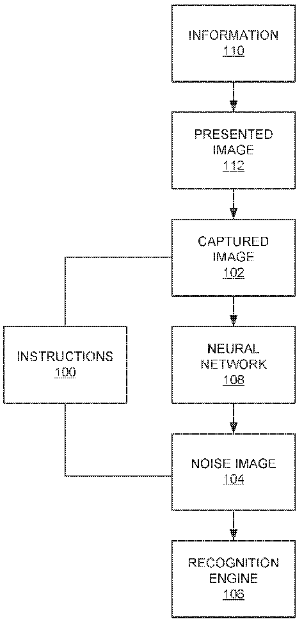

FIG. 1

(57) **Abstract:** An example adapter device includes a host-side connector to connect to a host device, the host-side connector including a host-side electrical contact to connect to a corresponding electrical contact of the host device. The adapter device further includes a storage-side connector to connect to storage devices operable under different protocols, the storage-side connector including a storage-side electrical contact to connect to a connected storage device. The adapter device further includes a circuit to apply a bias voltage to the host-side electrical contact. The host-side electrical contact is to provide a protocol-indicating voltage to indicate to the host device a protocol of the connected storage device. The protocol-indicating voltage is dependent on the connected storage device's influence on the bias voltage.

[Continued on next page]

(74) **Agent: SORENSEN, C. Blake** et al.; HP Inc., 3390 E. Harmony Road, Mail Stop 35, Fort Collins, Colorado 80528-9544 (US).

(81) **Designated States** *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

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

Declarations under Rule 4.17:

— *as to the identity of the inventor (Rule 4.17(i))*

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

Published:

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

NEURAL NETWORKS TO PROVIDE IMAGES TO RECOGNITION ENGINES

BACKGROUND

[0001] Electronic devices are capable of capturing a processing images, such as digital photographs. Many mobile phones or smartphones available today contain cameras and digital photographs are used by many people to record useful information.

BRIEF DESCRIPTION OF THE DRAWINGS

[0002] FIG. 1 is a block diagram of example instructions to obtain a noise image from a neural network and provide the noise image to a recognition engine.

[0003] FIG. 2 is a block diagram of example instructions to obtain a noise image from application of a captured image to a neural network and to provide the noise image and the captured image to a recognition engine.

[0004] FIG. 3 is a block diagram of an example electronic device to provide a noise image obtained from a captured image to a recognition engine.

[0005] FIG. 4 is a flowchart of an example method to obtain a noise image from a neural network and provide the noise image to a recognition engine.

[0006] FIG. 5 is a flowchart of an example method to obtain a noise image from a neural network and provide the noise image and a captured image to a recognition engine.

[0007] FIG. 6 is a graph of results of a test of using captured images and noise image to recover hidden information with a recognition engine.

DETAILED DESCRIPTION

[0008] Hidden information may be applied to images for various uses, such as copyright, data protection/integrity, and automation features, such as automatic hyperlink following. Such information may be referred to as a mark or watermark. Examples of such information include a text string that is hidden within an image. Digimarc™ provides example tools to encode and decode information from presented images.

[0009] A recognition engine may be used to decode information hidden in presented images. For example, a user may see a picture in a magazine and may use a mobile phone device to digitally capture the picture in the magazine. A processor of the phone may perform recognition that identifies concealed information using a steganographic technique or an invisible or nearly invisible watermark application technique to identify specific pixel or image attributes that encode the information. Successful detection and decoding of the information become increasingly difficult as imperceptibility of the information is increased. However, imperceptibility of the information is often useful, so that the information remains hidden to the human observer.

[0010] Images may be preprocessed to increase recognition and recovery of such information. Examples of preprocessing discussed herein include generating a noise image using a trained neural network, where the noise image is an enhanced representation of the hidden information that is normally outside human perception or nearly so in the presented image.

[0011] A neural network may be trained using publicly available images. The neural network may be trained to extract a secret image that has been steganographically encoded within a container image. Both the image encoded into the container image and the container image itself may be human intelligible, *e.g.*, digital photographs. After training, the neural network is put into operation.

[0012] Subsequently, an image may be captured, for example, by a camera of a mobile phone. The captured image may be a digital photograph of a subject such as printed document, an advertisement, an object, or similar instance of a presented image. Information, such as a hyperlink, message, or similar, may be encoded within the imagery of the subject and therefore obtainable from the captured image by a recognition engine.

[0013] Prior to recognition, the captured image may be preprocessed by the neural network to extract a representation of the encoded information from the captured image. The representation may be considered a noise image, which is generally not intelligible to humans. The noise image may be provided to the recognition engine to decode the information hidden in the presented image. Providing the noise image to the recognition engine may increase the likelihood that the recognition engine can recover the information.

[0014] As such, rather than merely passing a captured image to the recognition engine, a noise image is generated by a neural network and the noise image is passed to the recognition engine. Further, the neural network may be trained to generally extract a steganographically encoded image from a container image and need not be trained to the specific technique used to encode information for recognition by the recognition engine. The neural network may be trained for image-to-image steganography without regard to the encoding of text, such as messages or hyperlinks, into container images.

[0015] FIG. 1 shows example instructions 100 to process a captured image 102 to obtain a noise image 104 that may be passed to a recognition engine 106. A neural network 108 is used to generate the noise image 104 from the captured image 102. The instructions 100 are executable by a processor and may be stored in a non-transitory computer readable medium.

[0016] The instructions 100 provide the captured image 102 to the neural network 108. The captured image 102 may be obtained from a camera. The captured image 102 may be a representation of a presented image 112 that was steganographically encoded with information 110. For example, a presented image 112 may be a digital image that is printed, and the captured image 102 may be a digital photograph of the printed version of the presented image 112. In another example, the presented image 112 is displayed on a display device, and the captured image 102 may be a digital photograph of the display device.

[0017] The information 110 may be encoded into the presented image 112 using a steganographic technique that does not use an image as a source of the information 110. That is, image-to-image steganography is not used. For example, the information 110 may include text that may be encoded into the presented image 112 by modifying pixel information of the presented image 112. Examples include modifying the a bit or bits of pixels of an image, modifying a brightness, chrominance, and/or luminance of pixels or regions of the image, modifying color-channel information of an image or region thereof, introducing imperceptible patterns into pixels or regions of an image, or similar.

[0018] The neural network 108 is trained to extract steganographically encoded images from container images, *i.e.*, image-to-image steganography. The neural network 108 may be trained with human-intelligible images, such as images from public sources or a set of standard training images. Human-intelligible images include digital photographs. For example, a “secret” digital photograph may be steganographically encoded into a container digital photograph, so only the container digital photograph is readily observable. The neural network 108 may be trained using numerous secret and container images to extract secret images from modified container images. Examples of such a neural network include that described by Baluja in “Hiding Images in Plain Sight: Deep Steganography.”

[0019] The neural network 108 may be based on an autoencoder architecture is used to embed a full-size color image (secret image) in another image (container or cover image). In such a model, a secret image may be hidden across all available bits of a container image.

[0020] The instructions 100 further obtain the noise image 104 from the neural network 108. This may be performed automatically and without user intervention in response to the captured image 102 being received by the neural network 108.

[0021] The noise image 104 is representative of the information 110 encoded within the presented image 112 of which the captured image 102 is a copy. The noise image 104 is not a reconstruction of a secret image encoded into a container image. That is, the noise image 104 is an image representation of encoded information 110 that may have not originally been an image. For example, the information 110 may contain text. As such, the noise image 104 may be human-unintelligible and not readily understandable by an observer.

[0022] The instructions 100 further output the noise image 104 to the recognition engine 106 to decode the information 110 represented by the noise image 104. The recognition engine 106 may expose to the instructions 100 functionality that is limited to receiving images and outputting corresponding decoded information. For example, the recognition engine 106 may be proprietary, *e.g.*, Digimarc™, and detailed information about its implementation and operational methodology may be not readily available. The recognition engine 106 may be designed to process captured images 102 to obtain the information 110. Providing noise images 104 to the recognition engine 106 may increase the likelihood that the recognition engine 106 can recognize the information 110, as opposed to merely providing the captured image 102 as would be expected by the recognition engine 106.

[0023] In an example of operation, information 110 includes a textual copyright notification and the presented image 112 is a photograph to which the copyright notification pertains. The copyright notification is encoded within the digital photograph using a steganographic technique or an invisible or nearly invisible watermark application technique. The resulting image is printed. At a later time, a user takes a digital photograph of the printed image using, for example, a smartphone that executes the instructions 100. The smartphone provides the captured image 102 to the neural network 108 and passes the resulting noise image 104 to the recognition engine 106. The recognition engine

106 may then output the copyright notification as obtained from the noise image 104. The smartphone may then display the copyright notification to the user.

[0024] In another example of operation, information 110 includes a hyperlink and the presented image 112 is a digital advertisement. The hyperlink is encoded within the advertisement using a steganographic technique or an invisible or nearly invisible watermark application technique. The resulting image is displayed on a display device. At a later time, a user takes a digital photograph of the encoded advertisement using, for example, a smartphone that executes the instructions 100. The smartphone provides the captured image 102 to the neural network 108 and passes the resulting noise image 104 to the recognition engine 106. The recognition engine 106 may then output the hyperlink as obtained from the noise image 104. The smartphone may then open the hyperlink in a user agent, such as a web browser.

[0025] The neural network 108 is trained to extract secret images from container images. However, the information 110 may be encoded into the presented image 112 using any suitable steganographic technique or an invisible or nearly invisible watermark application technique. As such, the neural network 108 may generate a noise image 104 that is not human-intelligible. For example, the noise image 104 representative of the encoded copyright notification or hyperlink may be unreadable by a person. Rather, such a noise image 104 may appear to an observer as noise. Providing the noise image 104 to the recognition engine 106 may increase the likelihood of the recognition engine 106 obtaining the copyright notification, hyperlink, or other information 110. This may provide a more satisfying user experience, in that proper detection of the information 110 may occur more frequently. Alternatively, a parameter of the steganographic technique or an invisible or nearly invisible watermark application technique used to encode the information 110 may be adjusted to reduce the apparentness of the encoded information 110 in the container image. That is, the modification of the presented image 112 may be made more subtle, so that the appearance of the container image is closer to the presented image 112.

[0026] FIG. 2 shows example instructions 200 to process a captured image 102 to obtain a noise image 104 that may be passed to a recognition engine 106. Reference to the example sets of instructions and devices discussed herein may be made for details not repeated here.

[0027] A neural network 108 is used to generate the noise image 104 from the captured image 102. The captured image 102 may also be passed to the recognition engine 106, so that the recognition engine 106 may process one or both of the noise image 104 and the captured image 102. The instructions 200 are executable by a processor and may be stored in a non-transitory computer readable medium.

[0028] As discussed elsewhere the noise image 104 is generated by the neural network 108. In addition, to providing the noise image 104 to the recognition engine 106 to decode the information 110, the captured image 102 may also be outputted to the recognition engine to decode the information 110. This may increase the likelihood if obtaining successfully decoded information 202.

[0029] The recognition engine 106 may provide a success/fail indication concerning decoding information from an image. A success may be indicated by output of the successfully decoded information 202. A failure may be indicated by a failure message or code. Providing both the captured image 102 and the noise image 104 to the recognition engine 106 gives the recognition engine 106 two attempts to obtain one success, as the same information 110 is encoded in both the captured image 102 and the noise image 104.

[0030] Also shown in FIG. 2 are a container image 204 and an encoder 206 that encodes the information 110 as steganographic information within the container image 204. Implementation details of the encoder 206 need not be provided when training the neural network 108.

[0031] Further, a parameter, such as strength, of the encoder 206 may be decreased to achieve a target rate of successfully decoded information 202 with tolerable disruption of the presented image 112 as compared to the container image 204. That is, the encoder 206 modifies the container image 204 to steganographically encode the information 110, and a greater strength may increase human-perceptible changes in the resulting presented image 112. Hence, the strength, or other parameter, may be decreased to reduce human perceptibility of the information 110 in the presented image 112 because decreased decoding performance by the recognition engine 106 may be compensated by the provision of the noise image 104. In other words, use of the noise image 104 at recognition may allow for higher quality presented images 112.

[0032] The encoder 206 may implement a steganographic technique or an invisible or nearly invisible watermark application technique that uses a relationship between channels to conceal information. For example, the technique may encode the information 110 in the container image 204 using a U chrominance channel and a V chrominance channel. The technique may generally use channels with a complementary relationship to modify pixel data based on the information 110. In a multi-channel example, first input color channel data is associated with a first input color channel that is correlated to a second input color channel associated with second input color channel data based on a chrominance emphasis. The combined input color channel may be used to represent the information 110. An example channel includes an array of values where three arrays of values represent data for a color component for red green blue (RGB) components of RGB color space. For another example, a channel may refer to an array of luminance values, such as in a YUV color space. A color space is a mapping of a description of colors registerable by a sensor (*e.g.*, a human eye) to physically producible colors. Example color spaces include RGB, YUV, XYZ, cyan magenta yellow black (CMYK), hue saturation value (HSV), lightness with A and B color-opponent dimensions (LAB), and the like.

[0033] The recognition engine 106 may implement the same or similar steganographic technique or invisible or nearly invisible watermark application technique as the encoder 206 to detect and decode information 110 hidden in container images 204.

[0034] The neural network 108 may be trained without regard to the steganographic technique or invisible or nearly invisible watermark application technique used by the encoder 206 and recognition engine 106. Image-to-image steganography or other invisible or nearly invisible watermark application technique may be used to train the neural network 108 with readily available image sets. As such, the neural network 108 may generate noise images 104 that are not specific to the steganographic technique or other invisible or nearly invisible watermark application technique used by the encoder 206 and recognition engine 106 but that increase the detection and decoding performed by recognition engine 106.

[0035] FIG. 3 shows an example electronic device 300, such as a mobile phone, smartphone, or other portable computing device. The electronic device 300 includes a camera 302, memory 304, and a processor 306 coupled to the camera 302 and the memory 304. Reference to the example sets of instructions and devices discussed herein may be made for details not repeated here.

[0036] The processor 306 may be coupled to the memory 304 to communicate instructions and data therebetween to provide for execution of instructions by the processor 306. The processor 306 may include a central processing units (CPU), a microcontroller, a microprocessor, a processing core, a field-programmable gate array (FPGA), or similar device capable of executing instructions.

[0037] The memory 304 may include a non-transitory computer-readable storage medium that may be any electronic, magnetic, optical, or other physical storage device that stores executable instructions. The computer-readable storage medium may include, for example, random access memory (RAM), read-only memory (ROM), electrically-erasable programmable read-only memory (EEPROM), flash memory, a storage drive, an optical disc, and the like. The computer-readable storage medium may be encoded with executable instructions. The memory 304 may store instructions 308 that are executable by the processor 306.

[0038] The instructions 308 when executed cause the processor 306 to control the camera 302 to obtain a captured image 102, provide the captured image 102 to a neural network 108, obtain an enhanced image 310 from the neural network 108 in response to providing the captured image 102, and provide the enhanced image 310 to a recognition engine 106 to decode information 314 hidden within the captured image 102.

[0039] The neural network 108 is trained to extract steganographically encoded images from container images, as discussed elsewhere herein. The neural network 108 is trained without regard to the specific type of information 314 expected to be hidden within captured images 102 and without regard to the recovery techniques implemented at the recognition engine 106.

[0040] The enhanced image 310 is representative of information encoded within a presented image that is captured by the camera 302 as the captured image 102. The enhanced image 310 may include a noise image, as discussed elsewhere herein. The captured image 102 may also be passed to the recognition engine 106. Passing a noise image or a noise image and the captured image 102 to the recognition engine 106 increases the likelihood of successful decoding of the information 314.

[0041] The recognition engine 106 may provide a response to receiving the enhanced image 310. The response may include the information 314 as decoded from the enhanced image 310 or a message indicating that the information 314 is not decodable from the enhanced image 310. The message may be presented to the user or may trigger the processor 306 to take an action. If the captured image 102 is also passed to the recognition engine 106, then the response may include the information 314 as decoded from the captured image 102 or a message indicating that the information 314 is not decodable from the captured image 102. The information 314 may be provided if either or both of the enhanced image 310 and captured image 102 are successfully decoded. The message may be provided if none of the enhanced image 310 and captured image 102 are successfully decoded.

[0042] The processor 306 may perform various operations automatically without user intervention. In various examples, the processor 306 provides the captured image 102 to the neural network 108, obtains the enhanced image 310 from the neural network, and provides the enhanced image 310 to the recognition engine 106 automatically and without user intervention. These operations may be performed in response to capturing an image. In various examples, all images captured by the camera 302 undergo these operations automatically and without user intervention and, when information 314 is detected as hidden within a particular captured image 102, then the processor 306 notifies the user of the electronic device 300 or takes other action.

[0043] The electronic device 300 may further include a network interface 316 connected to the processor 306.

[0044] Information steganographically encoded into images or encoded using other invisible or nearly invisible watermark application technique may include hyperlinks. For example, a printed or displayed advertisement may include an image that steganographically encodes a hyperlink to the advertiser’s web site. As such, a user of the electronic device 300 may observe the image without noticing or being distracted by the hyperlink and the electronic device 300 may detect and decode the hyperlink.

[0045] As such, the processor 306 may further obtain a host address from the information 314 and control a user agent 318, such as a web browser, to access a host device 320 at the host address via a network 322, such as the internet. Accordingly, a web browser may be pointed to a web address to obtain further information.

[0046] In other examples, the neural network 108 and/or recognition engine 106 may be located at a server or similar electronic device and accessible to the electronic device 300 via the network 322. That is, processing may be offloaded to a server rather than being performed at the device 300.

[0047] FIG. 4 shows an example method 400 to obtain a noise image from a neural network and provide the noise image to a recognition engine. The method 400 may be used with any of the example sets of instructions and devices discussed herein. Reference to the example sets of instructions and devices discussed herein may be made for details not repeated here. The method starts at block 402.

[0048] At block 404, a captured image is provided to a neural network. The captured image may be a digital photograph of a presented image, which may be a printed or displayed image that steganographically or using other invisible or nearly invisible watermark application technique that encodes information, such as text, according to an arbitrary technique. The neural network may be trained to extract steganographically encoded images from container images. The neural network may be trained based on image-to-image steganography using images from available sources.

[0049] At block 406, a noise image is obtained from the neural network in response to providing the captured image. The noise image is not a reconstruction of a secret image encoded into a container image. Rather, the noise image is representative of information, such as text, encoded within a presented image of which the captured image is a copy. Such information may be encoded according to a technique that may be unavailable or proprietary or may be governed by unavailable or proprietary parameters.

[0050] At block 408, the noise image is outputted to a recognition engine to decode the information. The recognition engine operates according to the same technique as the encoding. Instead of operating on the capture image, the recognition engine operates on the noise image, which may thereby increase the likelihood of successful recognition and decoding.

[0051] The method 400 ends at block 410. The method 400 may be repeated, for example, continuously for a sequence of captured images, such as video frames.

[0052] FIG. 5 shows an example method 500 to obtain a noise image from a neural network and provide the noise image and a captured image, on which the noise image is based, to a recognition engine. The method 500 may be used with any of the example sets of instructions and devices discussed herein. Reference to the example sets of instructions and devices discussed herein may be made for details not repeated here. The method starts at block 502.

[0053] At blocks 404, 406, a captured image is provided to a neural network and a noise image is obtained.

[0054] At block 504, the noise image and the captured image are outputted to a recognition engine to decode the information. Instead of operating on the capture image alone, the recognition engine operates on both the noise image and the captured image, which may thereby increase the likelihood of successful recognition and decoding of the information.

[0055] If either of both of the noise image and the captured image result in successful recognition by the recognition engine, at block 506, then the information originally hidden in the presented image is provided in a response, at block 508. If neither the noise image nor the captured image result in successful recognition, then a message is provided as a response, at block 510. The message may be user message or programmatic message to indicate to a processor that no information was obtained.

[0056] The method 500 ends at block 512. The method 500 may be repeated, for example, continuously for a sequence of captured images, such as video frames.

[0057] A test of the above described techniques was made using a neural network model proposed by Baluja. The model was trained with image-to-image stenography and was not trained with Digimarc™ watermark examples. Two smartphone models, a Samsung J2 Prime™ ("Device 1") and a Samsung Galaxy S8™ ("Device 2"), were used to recognize the watermark at distances of 12.2 inches, 6.1 inches, and with free movement ("F").

[0058] Digimarc™ watermarks were applied with a low watermark perception configuration (watermark per inch or WPI of 150 and strength 3) in four colorful RGB images from the LIVE1 dataset. The images were displayed on a screen during the test and each image was tested separately. In addition, the noise images were also extracted from the watermarked images and displayed on the screen for recognition. Recognition accuracy results are shown in FIG. 6 for application of the Digimarc™ recognition engine directly on captured images and on noise images, which may be considered reconstructions of the applied watermark. As can be seen, use of noise images resulted in a larger number of successful recognitions. Further, it was observed that use of noise images resulted in faster detection. As such, it was unexpectedly apparent that faster and more accurate processing of watermarks was possible with a neural network that was trained on general images and not images specific to the watermarking technique applied.

[0059] Another set of tests was performed on printed media.

[0060] A Digimarc™ watermark was applied to two digital images varying only the strength ("STR") of application between the values 1, 3, 5, 7, and 10, generating 10 examples. These examples were printed by different printers: an HP Inkjet 8610™ and an HP Sprocket 2.0™. The same WPI value of 75 and the same hidden information, a hyperlink, was used in all tests.

[0061] The comparison between the recognition results before and after the use of the techniques discussed herein are shown in Tables 1 and 2 below. In the tables, "X" denotes not recognized and "V" denotes recognized. As can be seen, accuracy of recognition improved overall. It is contemplated that variance in accuracy among different printer models can be compensated for by providing both the captured image and the noise image to the recognition engine, as discussed herein.

[0062] Table 1:

PRINTER 1					
CAPTURED IMAGE 1			NOISE IMAGE OF CAPTURED IMAGE 1		
STR	DEVICE 1	DEVICE 2	STR	DEVICE 1	DEVICE 2
1	X	X	1	X	X
3	V	X	3	V	V
5	V	V	5	V	V
7	V	V	7	V	V
10	V	V	10	V	V
CAPTURE IMAGE 2			NOISE IMAGE OF CAPTURED IMAGE 2		
STR	DEVICE 1	DEVICE 2	STR	DEVICE 1	DEVICE 2
1	X	X	1	X	X
3	X	X	3	X	X
5	X	X	5	V	V
7	X	V	7	V	V
10	V	V	10	V	V

[0063] Table 2:

PRINTER 2					
CAPTURED IMAGE 1			NOISE IMAGE OF CAPTURED IMAGE 1		
STR	DEVICE 1	DEVICE 2	STR	DEVICE 1	DEVICE 2
1	X	X	1	X	X
3	X	X	3	X	X
5	X	X	5	X	X
7	X	V	7	X	X
10	V	V	10	V	V
CAPTURE IMAGE 2			NOISE IMAGE OF CAPTURED IMAGE 2		
STR	DEVICE 1	DEVICE 2	STR	DEVICE 1	DEVICE 2
1	X	X	1	X	X
3	X	X	3	X	X
5	X	X	5	X	X
7	X	V	7	X	X
10	V	V	10	V	V

[0064] In view of the above, it should be apparent that a neural network trained for image-to-image steganography may be used to extract an image representation of steganographically encoded information, and particularly text information, from a presented image without undue regard to the steganographic technique used to hide that information. Training of the neural network may be based on readily available images, such as digital photographs, instead of examples of text information, such as copyright notices, hyperlinks, ownership information, or similar. As such, a recognition engine may be provided with a noise image representative of the hidden information that increases the likelihood of successfully recognizing and decoding the information.

[0065] It should be recognized that features and aspects of the various examples provided above can be combined into further examples that also fall within the scope of the present disclosure. In addition, the figures are not to scale and may have size and shape exaggerated for illustrative purposes.

CLAIMS

1. A non-transitory computer-readable medium comprising instructions executable by a processor to:

provide a captured image to a neural network, the neural network being trained to extract steganographically encoded images from container images;

obtain a noise image from the neural network in response to providing the captured image, the noise image being representative of information encoded within a presented image of which the captured image is a copy; and

output the noise image to a recognition engine to decode the information.

2. The non-transitory computer-readable medium of claim 1, wherein the instructions are further to:

output the captured image with the noise image to the recognition engine to decode the information.

3. The non-transitory computer-readable medium of claim 1, wherein the neural network is trained with human-intelligible images, and wherein the information as encoded within the presented image and as represented in the noise image is human-unintelligible.

4. The non-transitory computer-readable medium of claim 1, wherein the instructions are further to control a camera of an electronic device to capture the captured image.

5. The non-transitory computer-readable medium of claim 1, wherein the instructions are further to obtain a host address from the information and control a user agent to access a host device at the host address via a network.

6. The non-transitory computer-readable medium of claim 1, wherein the recognition engine exposes to the instructions functionality that is limited to receiving images and outputting corresponding decoded information.

7. An electronic device comprising:

a camera; and

a processor connected to the camera, the processor to:

control the camera to obtain a captured image;

provide the captured image to a neural network trained to extract steganographically encoded images from container images;

obtain an enhanced image from the neural network in response to providing the captured image, the enhanced image being representative of information encoded within a presented image; and

provide the enhanced image to a recognition engine to decode the information.

8. The electronic device of claim 7, wherein enhanced image contains a representation of noise encoded into the presented image to represent the information.

9. The electronic device of claim 7, wherein the processor is further to:

provide the captured image to the recognition engine to decode the information.

10. The electronic device of claim 9, wherein the processor is further to receive a response from the recognition engine, wherein the response contains one or both of:

the information as decoded from the enhanced image or a message indicating that the information is not decodable; and

the information as decoded from the captured image or a message indicating that the information is not decodable.

11. The electronic device of claim 10, wherein the processor is further to output the information if the information is contained in the response.

12. The electronic device of claim 7, wherein the neural network is trained with human-intelligible secret images steganographically encoded within human-intelligible images container images.

13. The electronic device of claim 7, further comprising a network interface connected to the processor, wherein the processor is further to obtain a host address from the information and control a user agent to access a host device at the host address via a network.

14. The electronic device of claim 7, wherein the processor is to provide the captured image to the neural network, obtain the enhanced image from the neural network, and provide the enhanced image to the recognition engine automatically and without user intervention.

15. A method comprising:

providing a captured image to a neural network, the neural network being trained to extract steganographically encoded images from container images;

obtaining a noise image from the neural network in response to providing the captured image, the noise image being representative of information encoded within a presented image of which the captured image is a copy; and

outputting the noise image or the noise image and the captured image to a recognition engine to decode the information.

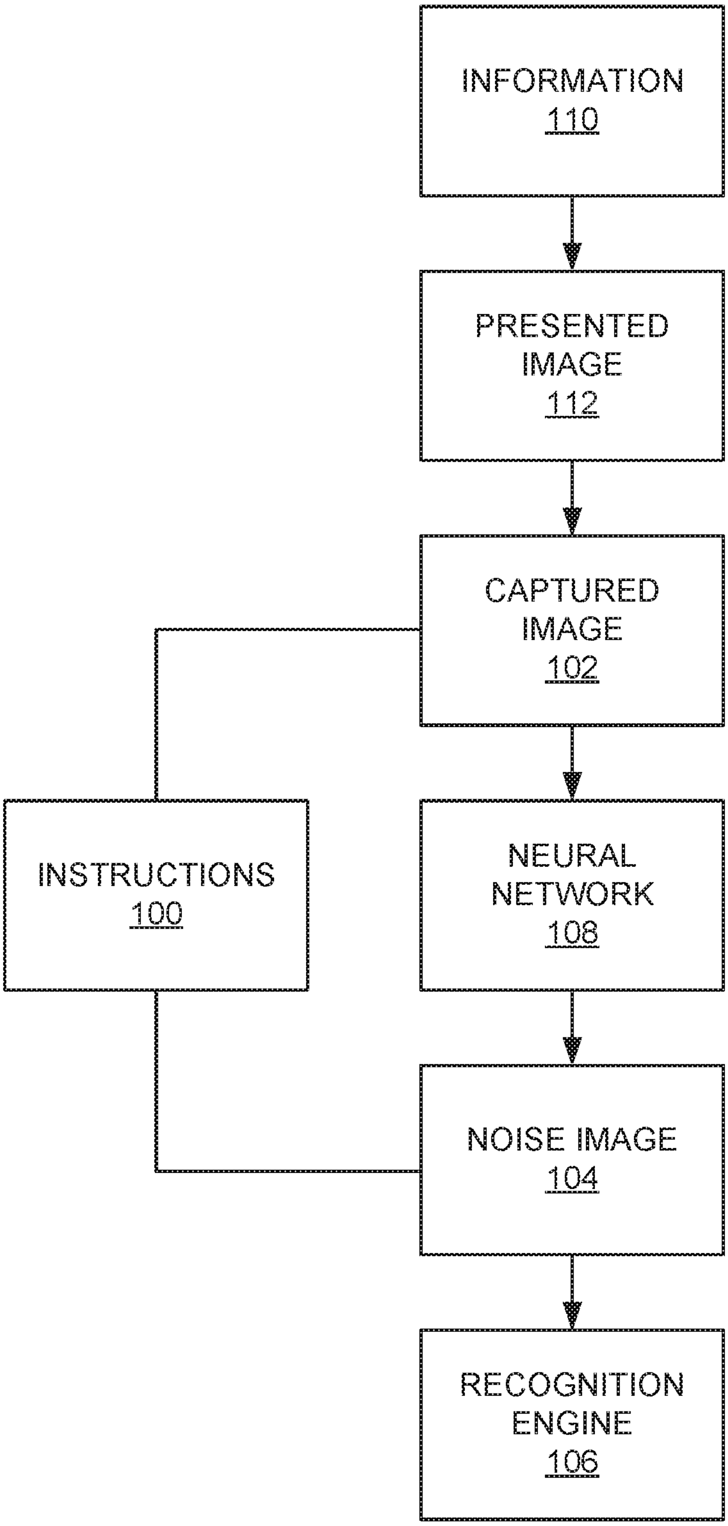
INFORMATION
110
PRESENTED IMAGE
112
CAPTURED IMAGE
102
INSTRUCTIONS
100
NEURAL NETWORK
108
NOISE IMAGE
104
RECOGNITION ENGINE
106

FIG. 1

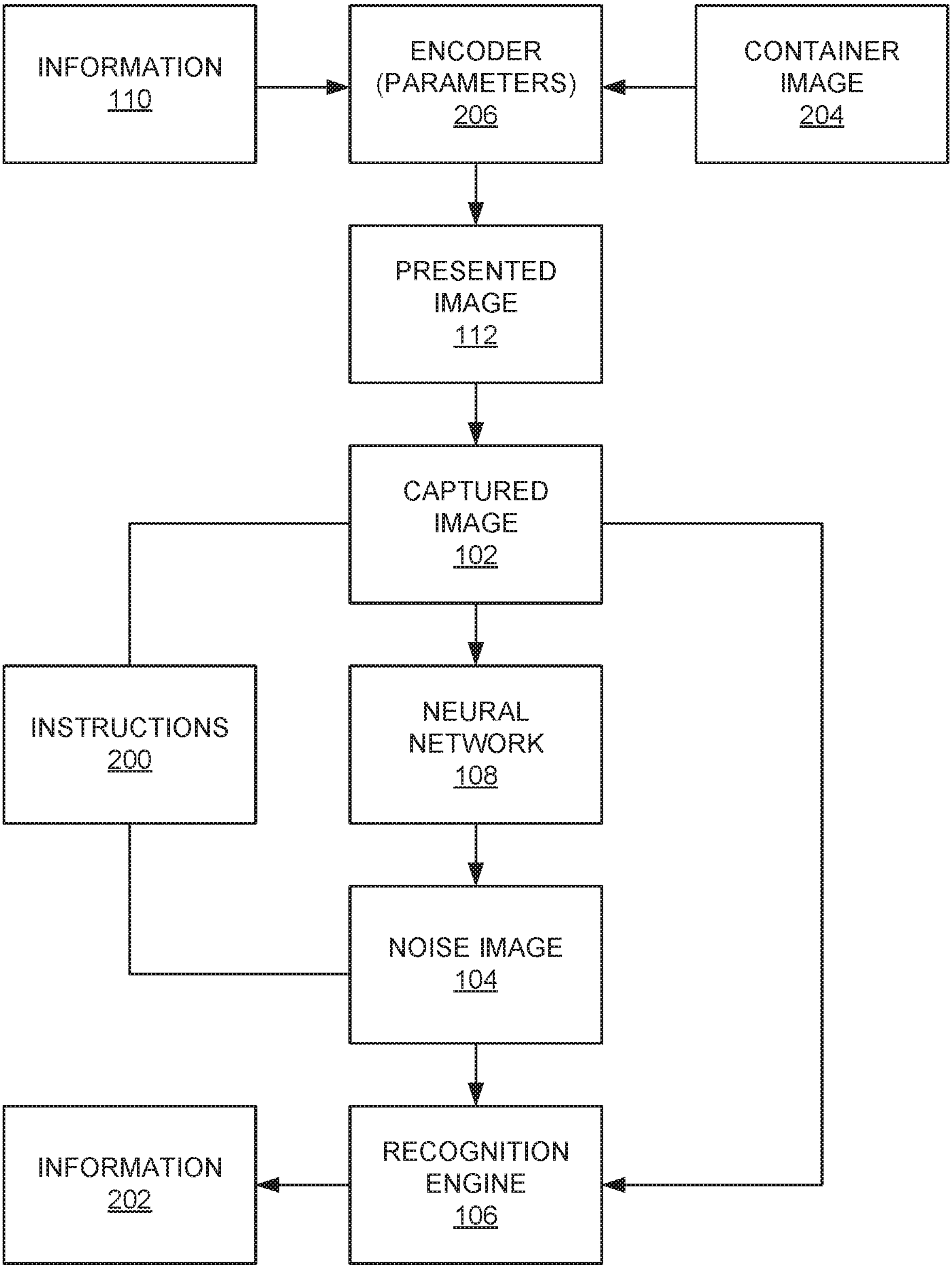
INFORMATION
110
ENCODER
(PARAMETERS)
206
CONTAINER
IMAGE
204
PRESENTED
IMAGE
112
CAPTURED
IMAGE
102
INSTRUCTIONS
200
NEURAL
NETWORK
108
NOISE IMAGE
104
INFORMATION
202
RECOGNITION
ENGINE
106

FIG. 2

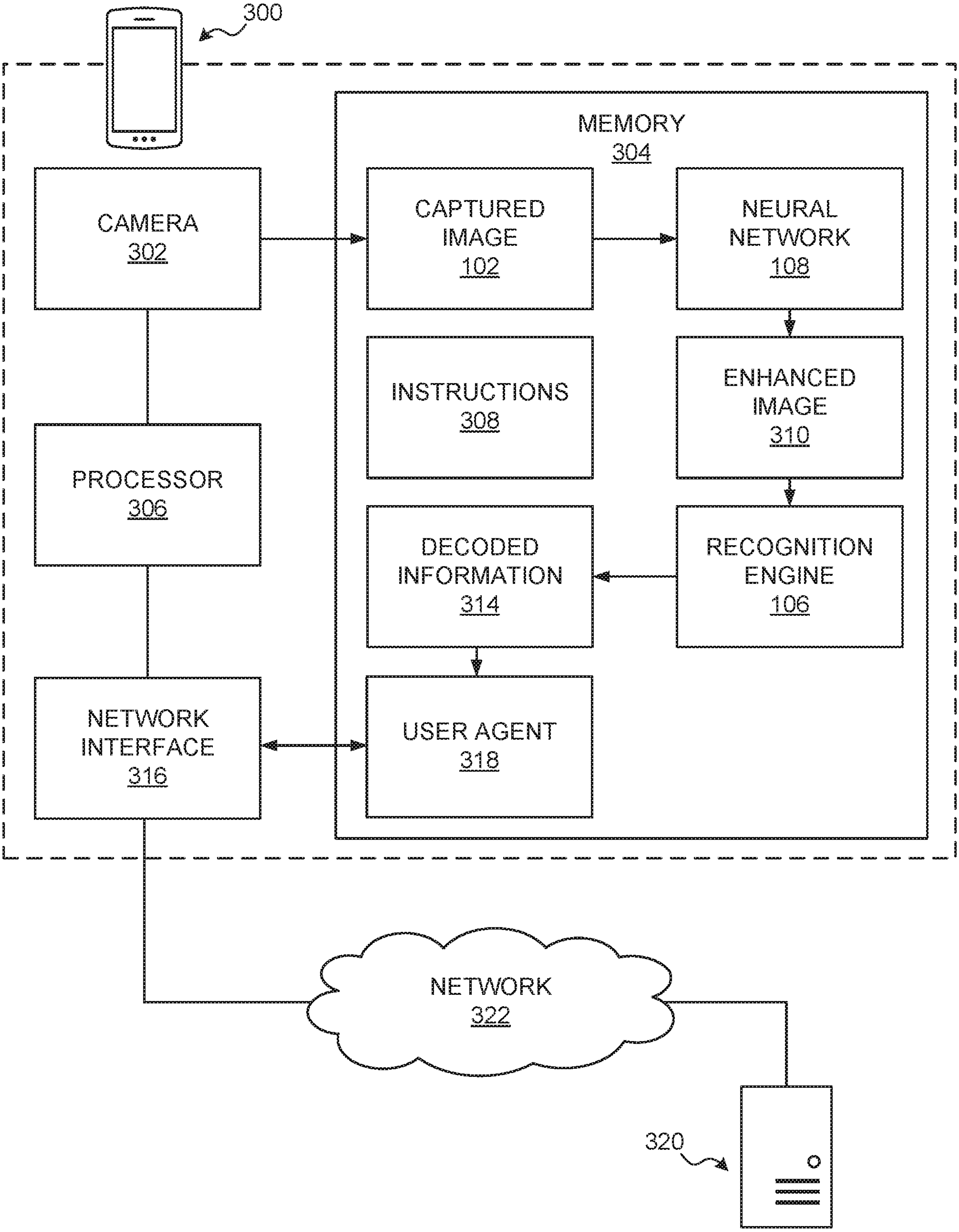
300
CAMERA
302
MEMORY
304
CAPTURED
IMAGE
102
NEURAL
NETWORK
108
INSTRUCTIONS
308
ENHANCED
IMAGE
310
PROCESSOR
306
DECODED
INFORMATION
314
RECOGNITION
ENGINE
106
NETWORK
INTERFACE
316
USER AGENT
318
NETWORK
322
320

FIG. 3

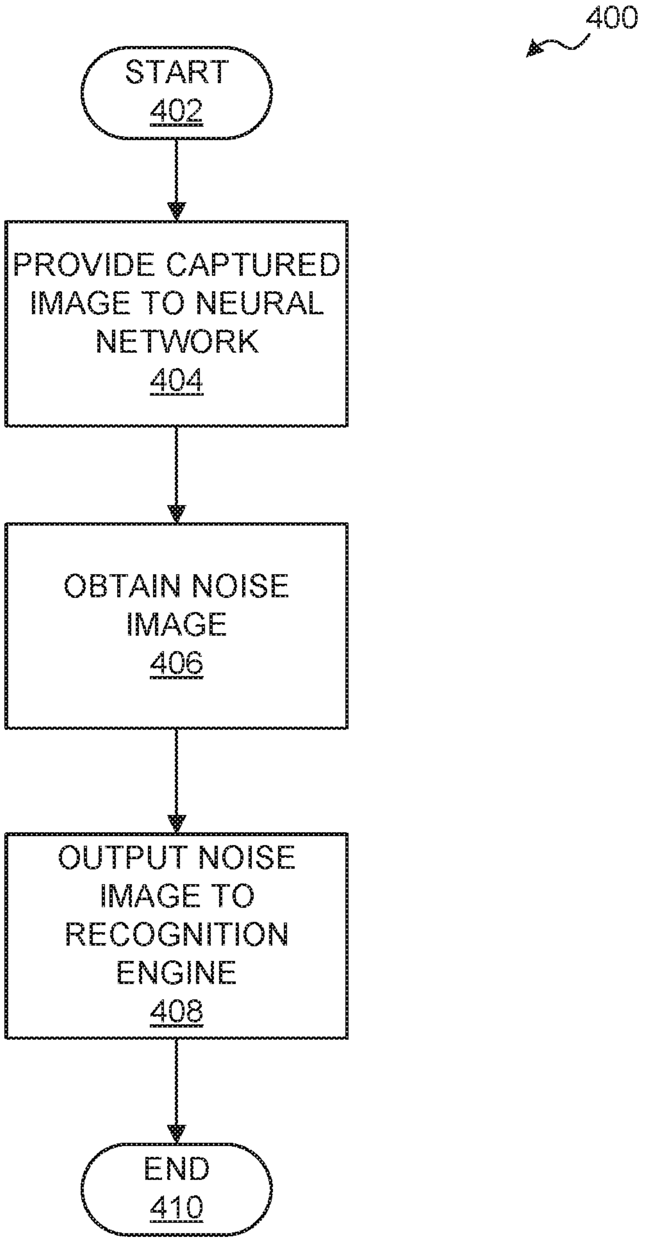
400
START
402
PROVIDE CAPTURED IMAGE TO NEURAL NETWORK
404
OBTAIN NOISE IMAGE
406
OUTPUT NOISE IMAGE TO RECOGNITION ENGINE
408
END
410

FIG. 4

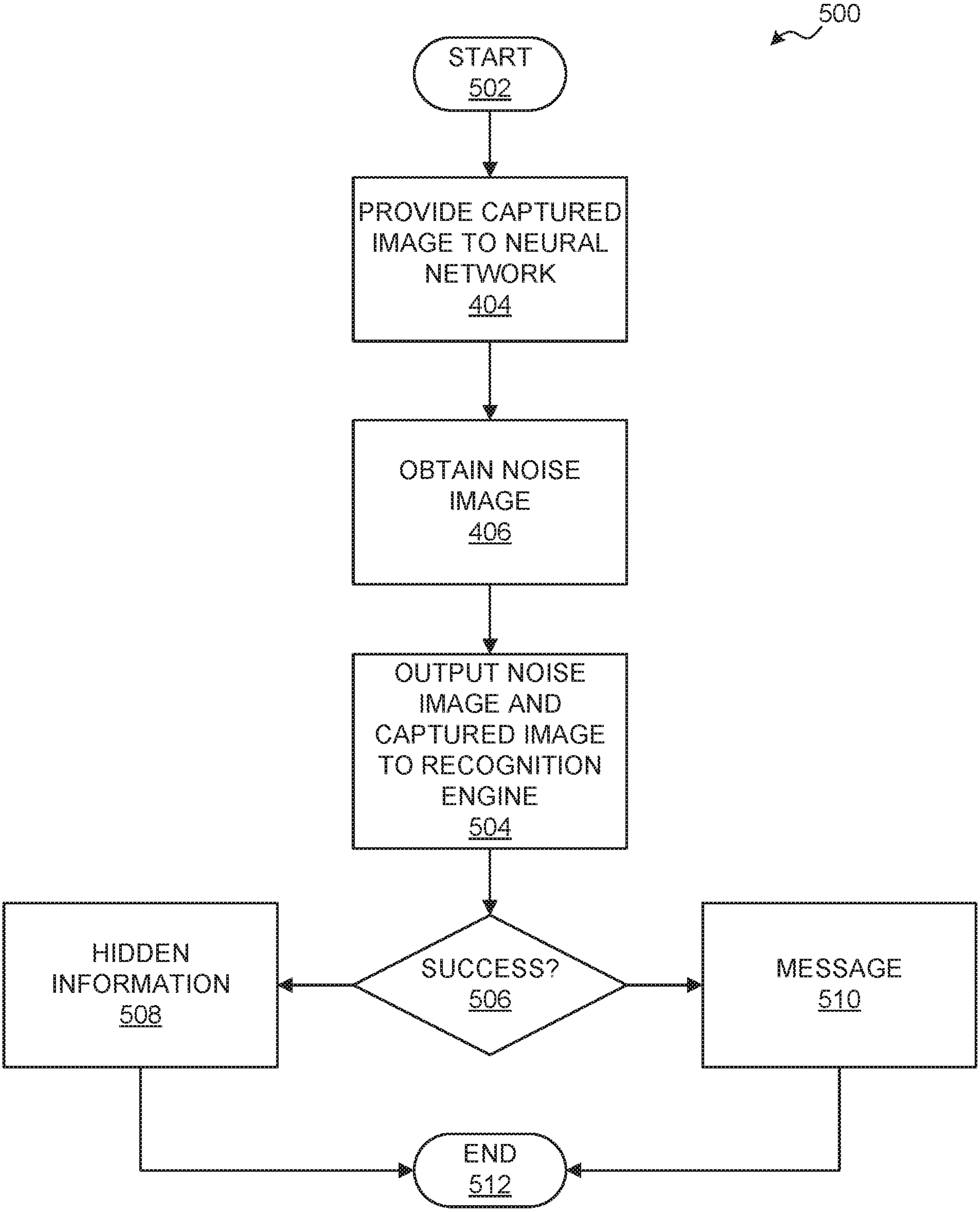
500
START
502
PROVIDE CAPTURED IMAGE TO NEURAL NETWORK
404
OBTAIN NOISE IMAGE
406
OUTPUT NOISE IMAGE AND CAPTURED IMAGE TO RECOGNITION ENGINE
504
HIDDEN INFORMATION
508
SUCCESS?
506
MESSAGE
510
END
512

FIG. 5

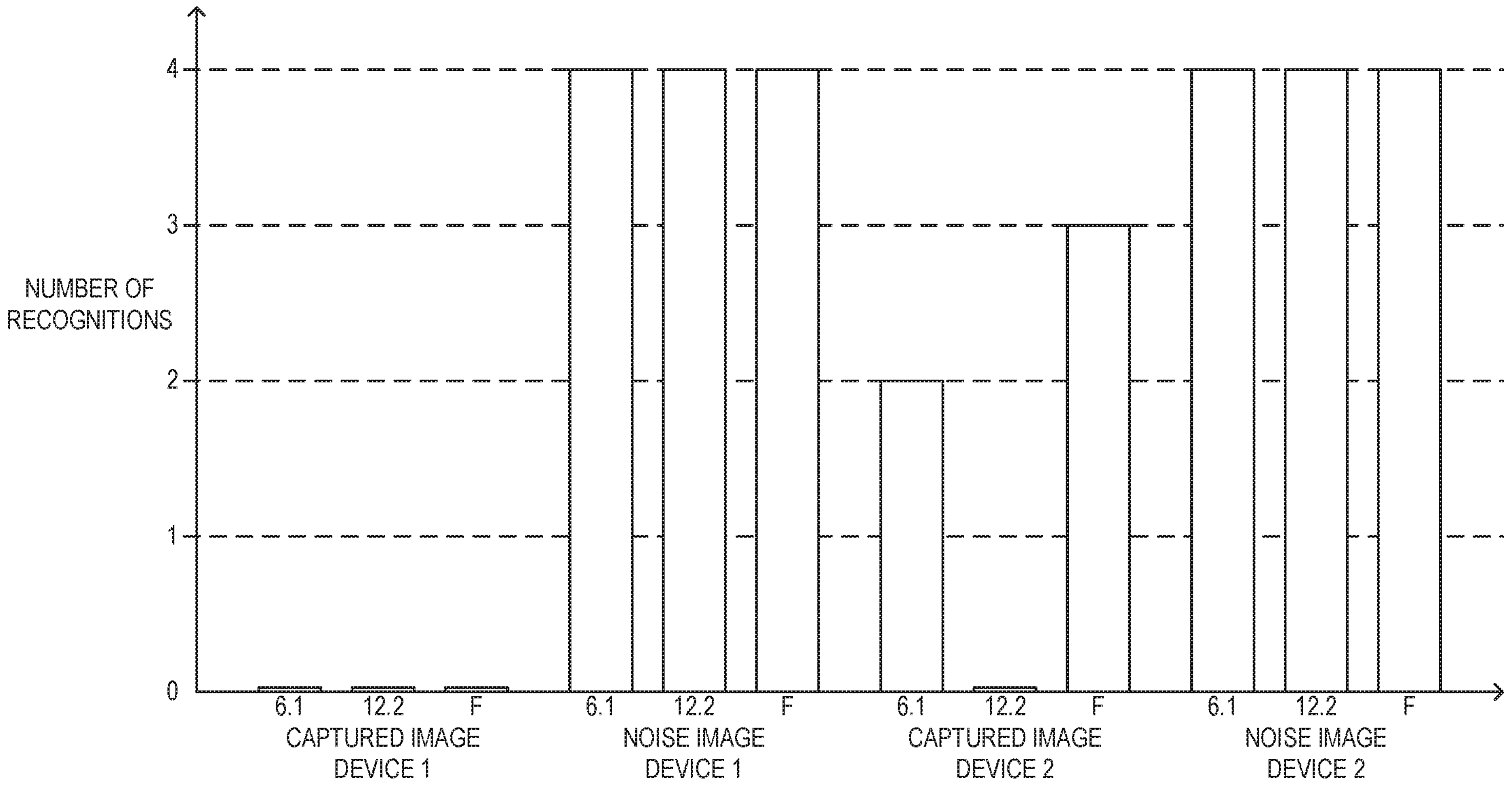
NUMBER OF RECOGNITIONS
4
3
2
1
0
6.1
12.2
F
CAPTURED IMAGE DEVICE 1
6.1
12.2
F
NOISE IMAGE DEVICE 1
6.1
12.2
F
CAPTURED IMAGE DEVICE 2
6.1
12.2
F
NOISE IMAGE DEVICE 2

FIG. 6

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US 2019/068047

A. CLASSIFICATION OF SUBJECT MATTER

G06K 9/62 *(2006.01)*
G06N 3/02 *(2006.01)*

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06K 9/00, 9/62, 9/20, 9/36, 9/46, 19/00, 19/06, G06N 3/00, 3/02, G06T 1/00, 1/20, 1/40, G06F 17/00

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

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

PatSearch (RUPTO Internal), USPTO, PAJ, Espacenet, Information Retrieval System of FIPS

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2018/0005343 A1 (DIGIMARC CORPORATION), 04.01.2018, paragraphs [0073] - [0077], [0080] - [0081], [0244] - [0247], [0250], [0360] - [0361], [0416], [0440], [0443], [0465] - [0466]	1-15
A	US 2019/0287204 A1 (TATA CONSULTANCY SERVICES LIMITED) 19.09.2019	1-15
A	US 8256665 B2 (DIGIMARC CORPORATION) 04.09.2012	1-15

□ Further documents are listed in the continuation of Box C.

□ See patent family annex.

* Special categories of cited documents:

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

"D" document cited by the applicant in the international application

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search 11 August 2020 (11.08.2020)	Date of mailing of the international search report 20 August 2020 (20.08.2020)
Name and mailing address of the ISA/RU: Federal Institute of Industrial Property, Berezhkovskaya nab., 30-1, Moscow, G-59, GSP-3, Russia, 125993 Facsimile No: (8-495) 531-63-18, (8-499) 243-33-37	Authorized officer N. Skokova Telephone No. 8(495) 531-65-15