

FILTERING METHOD, DEVICE, AND STORAGE MEDIUM

TECHNICAL FIELD

[0001] The present disclosure relates to the field of image processing technologies, and in particular, to a filtering method, a device and a storage medium.

BACKGROUND

[0002] Different regions in a coded image (picture) may have different permissions, which can be classified as confidentiality levels or other access permission levels. Users with the highest permission can view correct content of all regions. Users with the lowest permission can only view correct content of regions corresponding to their permission level and cannot view the correct content of regions with higher permission levels. For example, these higher-permission regions are presented to lower-permission users with mosaic patterns or obscured content. One application scenario of permission regions is to present different ranges of correct images to users with different permissions. In principle, the higher the permission level of users, the larger the range of correct images they can see, and the smaller the range of obscured images.

[0003] Existing solutions do not perform filtering on permission regions, which can affect both the subjective and objective quality of the image.

[0004] The above content is only used to assist in understanding the technical solution of the present disclosure and does not represent an acknowledgment that the above content is prior art.

SUMMARY

[0005] The main purpose of the present disclosure is to provide a filtering method, a device and a storage medium, aiming at solving the technical problem that the subjective and objective quality of an image will be affected if the permission region is not filtered in the prior art.

[0006] In order to achieve the above purpose, the present disclosure provides a filtering method, which includes the following steps:

[0007] determining a filtering strategy corresponding to a current filtering unit and a permission region where the current filtering unit is located, where the permission region is an image region with a permission level for an image therein;

[0008] determining a permission level of a sample within the current filtering unit according to the permission level of the permission region;

[0009] classifying the sample within the current filtering unit according to the permission

level of the sample; and

[0010] performing a corresponding filtering operation on the current filtering unit according to the filtering strategy and a classification result.

[0011] Optionally, the filtering strategy is determined with indication information parsed from a bit stream, and the indication information is transmitted at a sequence level, an image level, a Slice level, a Tile level, or a filtering unit level.

[0012] Optionally, the permission region includes at least one filtering unit, the current filtering unit is allowed to span a plurality of permission regions, and the filtering unit includes at least one sample;

[0013] where the determining a permission level of a sample within the current filtering unit according to the permission level of the permission region includes:

[0014] setting the permission level of the sample within the current filtering unit to be consistent with the permission level of the permission region where the sample is located.

[0015] Optionally, the current filtering unit is a basic unit in a filtering process and different filtering units are allowed to perform different filtering operations;

[0016] wherein the performing a corresponding filtering operation on the current filtering unit according to the filtering strategy and a classification result includes:

[0017] not performing the filtering operation on one or more first samples within the current filtering unit and performing the filtering operation on one or more second samples within the current filtering unit;

[0018] or,

[0019] performing the filtering operation on all samples within the current filtering unit;

[0020] or,

[0021] not performing the filtering operation on all the samples within the current filtering unit.

[0022] Optionally, each of the one or more first samples is a non-filterable sample and each of the one or more second samples is a filterable sample.

[0023] Optionally, the performing the filtering operation on all samples within the current filtering unit includes:

[0024] converting a non-filterable sample within the current filtering unit into a filterable sample by deriving information of reference blocks or reference samples, and performing the filtering operation on all filterable samples, where the information of the reference blocks or the reference samples is derived from available blocks or available samples within and around the permission region or is derived from preset information;

[0025] where the available blocks or the available samples are blocks or samples which have a permission lower than or equal to a permission of the non-filterable samples and have been decoded or coded.

[0026] Optionally, further including at least one of:

[0027] taking a sample within the current filtering unit, for which reference block information is required in the filtering process while at least one reference block is unavailable, as the non-filterable sample;

[0028] taking a sample within the current filtering unit, for which reference samples are required in the filtering process while at least one reference sample is unavailable, as the non-filterable sample;

[0029] taking a sample within the current filtering unit, for which reference samples are required in the filtering process while all reference samples are unavailable, as the non-filterable sample; or

[0030] taking a sample within the current filtering unit, for which reference samples are required in the filtering process while a proportion of unavailable reference samples among all reference samples is greater than or equal to a preset value, as the non-filterable sample.

[0031] Optionally, the filtering method further includes:

[0032] determining, according to a preset filtering manner, whether the reference block information is required in the filtering process for the sample within the current filtering unit; and

[0033] determining, based on a permission of the sample and a permission of the reference block, whether the reference block is available.

[0034] Optionally, the reference block is a block required for making a filtering decision and/or performing the filtering operation, the reference block includes at least one sample, and the permission of the reference block is a highest permission of all samples within the reference block.

[0035] Optionally, the determining, based on a permission of the sample and a permission of the reference block, whether the reference block is available includes:

[0036] determining that the reference block is available in response to determining that the permission of the reference block is lower than or equal to the permission of the sample and at least one of following preset conditions is met;

[0037] where the preset conditions include:

[0038] the reference block has been decoded or coded;

[0039] the reference block and the sample are located in a same Slice or Tile; or

[0040] the reference block and the sample are located in different Slices or Tiles, and performing filtering operation on boundaries of Slices or Tiles is allowed.

[0041] Optionally, after determining that the reference block is available in response to determining that the permission of the reference block is lower than or equal to the permission of the sample and at least one of the following preset conditions is met, the method further includes:

[0042] deriving the reference block information from decoded blocks or coded blocks with a permission level lower than or equal to the permission of the sample around an unavailable reference block, and/or deriving the reference block information from preset information; and

[0043] replacing information of the unavailable reference block with the derived reference block information to convert the unavailable reference block into an available reference block.

[0044] Optionally, the filtering method further includes:

[0045] acquiring reference samples corresponding to any one sample within the current filtering unit according to a preset filtering manner; and

[0046] determining, based on a permission of the sample and a permission of the reference samples, whether the reference samples are available.

[0047] Optionally, the reference samples are samples required for making a filtering decision and/or performing the filtering operation.

[0048] Optionally, the determining, based on a permission of the sample and a permission of the reference samples, whether the reference samples are available includes:

[0049] determining that the reference samples are available in response to determining that the permission of the reference samples is lower than or equal to the permission of the sample and at least one of following preset conditions is met;

[0050] where the preset conditions include:

[0051] the reference samples have been decoded or coded;

[0052] the reference samples and the sample are located in a same slice or tile; or

[0053] the reference samples and the sample are located in different Slices or Tiles, and performing filtering operation on Slice or Tile boundaries is allowed.

[0054] Optionally, after determining that the reference samples are available in response to determining that the permission of the reference samples is lower than or equal to the permission of the sample and at least one of the following preset conditions is met, the method further includes:

[0055] deriving reference sample information from decoded samples or coded samples with a permission level lower than or equal to the permission of the sample around unavailable

reference samples, and/or deriving the reference sample information from preset information; and

[0056] replacing information of the unavailable reference samples with the derived reference sample information to convert the unavailable reference samples into available reference samples.

[0057] Optionally, the preset filtering manner includes at least one of:

[0058] deblocking filtering;

[0059] sample adaptive offset;

[0060] enhanced sample adaptive offset;

[0061] cross-component sample adaptive offset; or

[0062] adaptive loop filtering.

[0063] In addition, in order to achieve the above objectives, the present disclosure further proposes a filtering device, including: a memory, a processor, and a filtering program stored on the memory and executable on the processor, and the filtering program is configured to implement the filtering method as described above.

[0064] In addition, in order to achieve the above purpose, the present disclosure further provides a storage medium, on which a filtering program is stored, and when the filtering program is executed by a processor, the filtering method described above is realized.

[0065] In addition, in order to achieve the above purpose, the present disclosure further provides a filtering apparatus, which includes:

[0066] an obtaining module, configured to determine a filtering strategy corresponding to a current filtering unit and a permission region where the current filtering unit is located, where the permission region is an image region with a permission level for an image therein;

[0067] an identification module, configured to determine a permission level of a sample within the current filtering unit according to the permission level of the permission region;

[0068] a classification module, configured to classify the sample within the current filtering unit according to the permission level of the sample; and

[0069] a filtering module, configured to perform a corresponding filtering operation on the current filtering unit according to the filtering strategy and a classification result.

[0070] In the present disclosure, by determining a filtering strategy corresponding to a current filtering unit and a permission region where the current filtering unit is located, determining a permission level of a sample within the current filtering unit according to the permission level of the permission region; classifying the sample within the current filtering unit according to the permission level of the sample; and performing a corresponding filtering operation on the

current filtering unit according to the filtering strategy and a classification result, so that constrained filtering operations on permission regions is realized and the subjective and objective quality of images is improved.

BRIEF DESCRIPTION OF DRAWINGS

[0071] FIG. 1 is a schematic diagram of a structure of a filtering device in a hardware operating environment according to an embodiment of the present disclosure.

[0072] FIG. 2 is a flow diagram of a first embodiment of a filtering method of the present disclosure.

[0073] FIG. 3 is a schematic diagram of a video coding framework in an embodiment of a filtering method of the present disclosure.

[0074] FIG. 4 is a flow diagram of a second embodiment of a filtering method of the present disclosure.

[0075] FIG. 5 is a schematic diagram of reference samples of a deblocking filtering manner in an embodiment of a filtering method of the present disclosure.

[0076] FIG. 6 is a schematic diagram of a vertical boundary and a horizontal boundary corresponding to a deblocking filtering manner in an embodiment of a filtering method of the present disclosure.

[0077] FIG. 7 is a schematic diagram of reference samples of a sample adaptive compensation manner in an embodiment of a filtering method of the present disclosure.

[0078] FIG. 8 is a schematic diagram of reference blocks of a sample adaptive offset manner in an embodiment of a filtering method of the present disclosure.

[0079] FIG. 9 is a schematic diagram of reference samples of an enhanced sample adaptive offset manner in an embodiment of a filtering method of the present disclosure.

[0080] FIG. 10 is a schematic diagram of reference samples of a cross-component sample adaptive offset manner in an embodiment of a filtering method of the present disclosure.

[0081] FIG. 11 is a schematic diagram of reference samples of an adaptive loop filtering manner in an embodiment of a filtering method of the present disclosure.

[0082] FIG. 12 is another schematic diagram of reference samples of an adaptive loop filtering manner in an embodiment of a filtering method of the present disclosure.

[0083] FIG. 13 is a flow diagram of a third embodiment of a filtering method of the present disclosure.

[0084] FIG. 14 is a structural block diagram of a first embodiment of a filtering apparatus of the present disclosure.

[0085] The realization, functional characteristics and advantages of the present disclosure will be further described with reference to the accompanying drawings in conjunction with the embodiments.

DETAILED DESCRIPTION

[0086] It should be understood that the specific embodiments described herein are only for explaining the present disclosure and are not intended to limit the present disclosure.

[0087] Referring to FIG. 1, FIG. 1 is a schematic diagram of a structure of a filtering device in a hardware operating environment according to an embodiment of the present disclosure.

[0088] As shown in FIG. 1, the filtering device may include: a processor 1001, such as a Central Processing Unit (CPU), a communication bus 1002, a user interface 1003, a network interface 1004, and a memory 1005. The communication bus 1002 is configured to realize connection communication between these components. The user interface 1003 may include a Display and an input unit such as a Keyboard, and optionally, the user interface 1003 may further include a standard wired interface and a wireless interface. The network interface 1004 may optionally include a standard wired interface and a wireless interface (such as a Wireless-Fidelity (Wi-Fi) interface). The memory 1005 may be a high-speed Random Access Memory (RAM) memory or a stable Non-Volatile Memory (NVM), such as a disk memory. The memory 1005 may optionally further be a storage apparatus independent of the aforementioned processor 1001.

[0089] Those skilled in the art can understand that the structure shown in FIG. 1 does not constitute a limitation on the filtering device, and it may include more or fewer components than those shown, or combine certain components, or arrange different components.

[0090] As shown in FIG. 1, the memory 1005, as a storage medium, may include an operating system, a network communication module, a user interface module, and a filtering program.

[0091] In the filtering device shown in FIG. 1, the network interface 1004 is mainly used for data communication with a network server; the user interface 1003 is mainly used for data interaction with users; the processor 1001 and memory 1005 in the filtering device of the present disclosure may be disposed in the filtering device, and the filtering device calls the filtering program stored in the memory 1005 through the processor 1001 to execute the filtering method provided in the embodiments of the present disclosure.

[0092] Reference will now be made in detail to exemplary embodiments, examples of which are illustrated in the accompanying drawings. When the following description refers to the accompanying drawings, unless otherwise indicated, the same numbers in different

accompanying drawings indicate the same or similar elements. Embodiments described in the following exemplary embodiments do not represent all embodiments consistent with the present disclosure. Rather, they are merely examples of apparatuses and methods consistent with some aspects of the present disclosure as detailed in the appended claims.

[0093] It should be noted that, as used herein, the terms “including”, “containing” or any other variations thereof are intended to cover non-exclusive inclusion, so that a process, method, article, or apparatus including a series of elements not only includes those elements, but also includes other elements not explicitly listed or elements inherent to such process, method, article, or apparatus. Without further limitations, elements defined by the phrase “including a...” do not exclude the existence of additional identical elements in the process, method, article, or apparatus that includes the element. Furthermore, components, features, and elements with the same names in different embodiments of the present disclosure may have the same or different meanings, and their specific meanings need to be determined based on their explanations in the specific embodiments or further combined with the context in the specific embodiments.

[0094] It should be understood that although terms first, second, third, etc. may be used herein to describe various information, these information should not be limited to these terms. These terms are used only to distinguish the same type of information from one another. For example, without departing from the scope herein, first information may also be named as second information, and similarly, the second information may also be named as the first information. Depending on the context, the word “if” as used herein can be interpreted as “at” or “when” or “in response to determining”. Furthermore, as used herein, the singular forms “a”, “an” and “the” are intended to also include the plural forms, unless the context indicates to the contrary. It should be further understood that the terms “include” and “contain” indicate the presence of the stated features, steps, operations, elements, components, items, categories, and/or groups, but do not exclude the presence, occurrence, or addition of one or more other features, steps, operations, elements, components, items, categories, and/or groups. The terms “or”, “and/or”, and “including at least one of” used in the present disclosure may be interpreted as inclusive or mean any one or any combination. For example, “including at least one of A, B, and C” means “any one of A, B, and C; A and B; A and C; B and C; or A, B, and C.” Similarly, “A, B, or C” or “A, B, and/or C” means “any one of A, B, and C; A and B; A and C; B and C; or A, B, and C”. Exceptions to this definition occur only when the combination of elements, functions, steps, or operations is inherently mutually exclusive in some way.

[0095] It should be understood that although the various steps in the flowcharts of the embodiments of the present disclosure are shown sequentially as indicated by the arrows, the

steps are not necessarily performed sequentially in the order indicated by the arrows. Unless explicitly stated herein, the execution of these steps is not strictly limited in order, and they can be executed in other orders. Moreover, at least a part of the steps in the accompanying drawings may include a plurality of sub-steps or stages, which may not necessarily be completed at the same time, but may be executed at different times, and the execution order may not necessarily be sequential, but may be executed alternately or alternatively with other steps or at least a part of sub-steps or stages of other steps.

[0096] Depending on the context, the word “if” and “in a case that” as used herein can be interpreted as “at” or “when” or “in response to determining” or “in response to detecting”. Similarly, depending on the context, the phrases “if determining” or “if detecting (stated condition or event)” may be interpreted as “when determining” or “in response to determining” or “when detecting (stated condition or event)” or “in response to detecting (stated condition or event)”.

[0097] It should be noted that in the present disclosure, step codes such as S10, S20, etc., are used to more clearly and concisely express the corresponding content and do not constitute a substantive limitation on the order. Those skilled in the art may execute S20 before S10 in specific implementations, but these should all be within the protection scope of the present disclosure.

[0098] It should be understood that the specific embodiments described herein are only for explaining the present disclosure and are not intended to limit the present disclosure.

[0099] An embodiment of the present disclosure provides a filtering method. Referring to FIG. 2, which is a flowchart diagram of a first embodiment of a filtering method according to the present disclosure, the filtering method can be applied to a decoding method flow for processing video data, and can also be applied to a coding method flow for processing video data.

[0100] In this embodiment, the filtering method includes the following steps.

[0101] Step S10: determining a filtering strategy corresponding to a current filtering unit and a permission region where the current filtering unit is located, where the permission region is an image region with a permission level for an image therein.

[0102] The execution subject of this embodiment may be a filtering device, and the filtering device may be implemented in various forms. For example, the filtering device described in the present disclosure may include a mobile terminal or smart terminal such as a tablet computer, a laptop computer, a palmtop computer, a video camera, and a fixed terminal device such as a desktop computer, all of which are equipped with the function of image processing,

and will be illustrated as an example of the filtering device in the subsequent description.

[0103] Referring to FIG. 3, which is a video coding framework, the video coding framework includes modules such as prediction, transformation, quantization, entropy coding, and filtering. The prediction module includes intra prediction and inter prediction, the intra prediction uses reconstructed pixels around a current block for prediction to remove spatial redundancy, and the inter prediction uses reconstructed pixels on temporal reference frames for prediction to remove temporal redundancy. The transformation module linearly maps spatial residual information to a transform domain (such as a frequency domain), with the goal of concentrating energy and removing frequency domain correlation of the signal. In theory, a transformation matrix is reversible and does not cause signal loss. The quantization module is a “many-to-one” mapping process that is irreversible and causes signal loss; the advantage is that it can significantly reduce the range of signal values, allowing a coder to provide a good approximation of an original signal with a small number of symbols, thereby improving compression rate. The entropy coding module is a lossless coding manner based on the principle of information entropy, which converts a series of element symbols (such as transform coefficients and mode information) used to represent a video sequence into a binary bit stream, removing statistical redundancy of these video element symbols. The filtering module enhances a reconstructed image, aiming to make the reconstructed image closer to an original image, while reducing the impact of blocking artifacts and ringing effects, and improving the quality of the reconstructed image.

[0104] In an optional embodiment, the filtering operation in this embodiment is aimed at filtering of a coded image (picture), and different regions in the coded image may have different permissions, which can be classified as confidentiality levels or other access permission levels. Users with the highest permission can view correct content of all regions. Users with the lowest permission can only view correct content of regions corresponding to their permission level and cannot view the correct content of regions with higher permission levels. For example, these higher-permission regions are presented to lower-permission users with mosaic patterns or obscured content. One application scenario of permission regions is to present different ranges of correct images to users with different permissions. The higher the permission level of users, the larger the range of correct images they can see, and the smaller the range of obscured images.

[0105] However, the current filtering manner does not perform filtering on samples within permission regions, which can greatly affect the subjective and objective quality of the image. In this embodiment, in order to solve the above technical problem, constrained filtering

operations are performed on permission regions with different permission levels according to a filtering operation and a permission level of a permission region. For example, there is no restriction on filtering of high-permission regions because surrounding reference sample information is available; for filtering of low-permission regions, if reference sample(s) is not available (for example, the reference sample(s) is located in a high-permission region), a constrained filtering operation is performed.

[0106] In this embodiment, when filtering an image (picture), a filtering unit is used as a basic unit in a filtering process for filtering. During filtering, each filtering unit can be filtered sequentially, or a plurality of filtering units can be filtered simultaneously. Each filtering unit corresponds to an image subregion in the image. In contrast, a permission region is an image region with a permission level, and for a single image, the image has a plurality of filter units and also a plurality of permission regions, and there is no correspondence between the number of filter units and the number of permission regions. A plurality of filter units may be contained within a permission region. For example, permission region X may contain one filtering unit A, or can simultaneously contain two filtering units A and B. Furthermore, one filtering unit may also be located in multiple different permission regions. For example, filtering unit C may be located in permission region X, or can also span permission regions X and Y simultaneously.

[0107] Filtering units divide the image into multiple image regions in horizontal and vertical directions. In this embodiment, the image can further be divided into chrominance and luminance components according to chrominance and luminance, and the filtering units contained in these two components are corresponding. One filtering unit may include the above two components, and one component may also include multiple filtering units. In this embodiment, the filtering process of the entire image is explained using filtering units as an example. The number and size of filtering units divided by a single image in this embodiment, as well as the number and size of permission regions, can be set according to actual image processing requirements, and are not limited in this embodiment.

[0108] In an optional embodiment, a filtering operation is performed according to a corresponding filtering strategy, and each filtering unit may be allowed to execute different filtering strategies. Taking a current filtering unit as an example, the filtering strategy executed by the current filtering unit may be the same as or different from filtering strategies of other filtering units. In this embodiment, the filtering strategy is determined with indication information, which can be transmitted at a sequence level (or a SPS (Sequence Parameter Set) level), or an image level (or a PPS (Picture Parameter Set) level or a frame level), or a Slice level, or a Tile level, or a filtering unit level (or CTU (Coding Tree Unit)/ CTB/ LCU (Largest

Coding Unit)/LCB (Largest Coding Block) level). The indication information is parsed from a bit stream, and based on the parsed indication information, the filtering strategy corresponding to each filtering unit can be determined. The level of the filtering strategy can also be a patch level. Syntax used to manage or control several images may be named as the sequence level; syntax used to manage or control a single frame image may be named as the image level; syntax used to manage or control parallel units (i.e., image regions) including several coding units may be named as the Slice level, or the Tile level, or the Patch level; syntax used to manage or control a largest coding unit may be named as the LCU level; syntax used to manage or control filtering units may be named as the filtering unit level. One filtering unit may include multiple largest coding units, and one largest coding unit may also include multiple filtering units.

[0109] Step S20: determining a permission level of a sample within the current filtering unit according to the permission level of the permission region.

[0110] In an optional embodiment, each filtering unit includes at least one sample, which can be a chrominance sample or a luminance sample of a pixel. A permission level of each sample within the filtering unit is determined by a permission level of a permission region where the filtering unit is located. The permission level can be a confidentiality level or another type of access permission level.

[0111] Since the filtering unit in this embodiment can be located in one permission region or multiple permission regions simultaneously, the permission levels of the samples within the filtering unit are not necessarily the same. This embodiment provides explanations for the two scenarios mentioned above. For the first scenario, when the filtering unit is located in one permission region, all samples within the filtering unit are also located in that permission region. In this case, the permission level of each sample within the filtering unit is consistent with the permission level of the permission region. For example, when the current filtering unit is located in permission region X, the permission level of all samples within the current filtering unit is the permission level of permission region X. For the second scenario, when the filtering unit is located in multiple permission regions, the samples within the filtering unit are also located in different permission regions accordingly. The permission level of each of the samples within the filtering unit is consistent with the permission level of the permission region where the sample located. For example, when the current filtering unit is simultaneously located in permission regions X and Y, some samples within the current filtering unit are located in permission region X, while others are located in permission region Y. The permission level of the samples located in permission region X is the permission level of permission region X, and the permission level of the samples located in permission region Y

is the permission level of permission region Y.

[0112] Step S30: classifying the sample within the current filtering unit according to the permission level of the sample.

[0113] In an optional embodiment, after determining the permission level of each sample within the current filtering unit, in this embodiment, all samples within the current filtering unit can be classified based on their permission levels. In this embodiment, all samples within the current filtering unit can be divided into first samples and second samples based on their permission levels. The specific process of classifying a sample within the current filtering unit in this embodiment can be implemented as follows.

[0114] An implementation is provided in this embodiment, specifically, according to the permission level of each sample, samples of a high permission level and samples of a low permission level within the current filtering unit can be determined, and all samples within the current filtering unit can be classified according to the high or low permission level of the samples. Optionally, a sample of the high permission level can be taken as a first sample and a sample of the low permission level can be taken as a second sample.

[0115] Another implementation is provided in this embodiment, specifically, in the filtering process within the current filtering unit, reference block information is used, and a permission level of a sample is compared with a permission level of a reference block. Based on the comparison result, all samples within the current filtering unit are classified. Optionally, a sample with a permission level lower than or equal to the permission level of the reference block can be taken as a first sample, and a sample with a permission level higher than the permission level of the reference block can be taken as a second sample.

[0116] Another implementation is provided in this embodiment, specifically, in the filtering process within the current filtering unit, reference sample(s) is used, and a permission level of a sample is compared with a permission level of the reference sample(s). Based on the comparison result, all samples within the current filtering unit are classified. Optionally, a sample with a permission level lower than or equal to the permission level of the reference sample(s) can be taken as a first sample, and a sample with a permission level higher than the permission level of the reference sample(s) can be taken as a second sample.

[0117] The above manners are only examples, and other manners can be selected in the actual filtering process, which is not limited in this embodiment.

[0118] In an optional embodiment, each of the one or more first samples may be a non-filterable sample and each of the one or more second samples may be a filterable sample.

[0119] Step S40: performing a corresponding filtering operation on the current filtering unit

according to the filtering strategy and a classification result.

[0120] In an optional embodiment, all samples within the current filtering unit may be divided into filterable samples and non-filterable samples based on their permission levels in this embodiment. A filterable sample is a sample that can undergo filtering operations directly, while a non-filterable sample is a sample that cannot undergo filtering operations directly.

[0121] In an optional embodiment, indication information parsed from the bit stream is used to determine the filtering strategy corresponding to the current filtering unit, and the permission level is used to classify the samples within the current filtering unit. Due to different classification situations of samples within the filtering unit, even if two filtering units have the same filtering strategy, the results after performing filtering operations may also be different. For example, assuming that the filtering strategy is partial filtering, where partial filtering strategy means not performing the filtering operation on the one or more first samples (i.e., non-filterable samples) within the filtering unit and performing a filtering operation on the one or more second samples (i.e., filterable samples) within the filtering unit, due to the different classification situations of samples within the filtering unit, the number of first samples and second samples within each filtering unit is also different, resulting in different final filtering results.

[0122] In an optional embodiment, the filtering strategy includes at least three manners: full filtering, partial filtering, and no filtering. Based on the classification result of the samples, it can be determined that the samples within the current filtering unit include at least one non-filterable sample and/or at least one filterable sample. Full filtering means performing filtering operations on all samples within the current filtering unit when there is at least one filterable sample; partial filtering means not performing the filtering operation on one or more non-filterable samples within the current filtering unit and performing the filtering operation on one or more filterable samples; no filtering means not performing any filtering operations on all samples within the current filtering unit when there is at least one non-filterable sample.

[0123] In an optional embodiment, a preset filtering manner is adopted when performing filtering operations on samples. The preset filtering manner in this embodiment includes at least one of deblocking filtering, sample adaptive offset, enhanced sample adaptive offset, cross-component sample adaptive offset, or adaptive loop filtering, and then the filtering operations can be performed on the samples according to the corresponding manner. Unlike the existing method of not performing any filtering operations on samples within permission regions, this embodiment performs filtering operations based on the preset filtering manner combined with the permission levels of the samples within the permission region, achieving

constrained filtering on permission regions with different permission levels, thereby improving the subjective and objective quality of images.

[0124] In this embodiment, by determining a filtering strategy corresponding to a current filtering unit and a permission region where the current filtering unit is located, determining a permission level of a sample within the current filtering unit according to the permission level of the permission region; classifying the sample within the current filtering unit according to the permission level of the sample; and performing a corresponding filtering operation on the current filtering unit according to the filtering strategy and a classification result, so that constrained filtering operations on permission regions is realized and the subjective and objective quality of images is improved.

[0125] Referring to FIG. 4, FIG. 4 is a flow diagram of a second embodiment of a filtering method of the present disclosure.

[0126] Based on the first embodiment mentioned above, the filtering method in this embodiment further includes the following steps.

[0127] Step S301: taking a sample within the current filtering unit, for which reference block information is required in the filtering process while at least one reference block is unavailable, as the non-filterable sample.

[0128] In an optional embodiment, in this embodiment, filterable samples and non-filterable samples within the current filtering unit can be determined based on information related to reference blocks. Specifically, for a certain sample within the current filtering unit, it is first necessary to determine whether reference block information is required during the filtering process for the sample. If the reference block information is required, it is further determined whether at least one reference block is unavailable. If both conditions are met, the sample can be determined as a non-filterable sample. Here, a reference block is a block required for making a filtering decision and/or performing a filtering operation. Conversely, if the reference block information is not required; or if the reference block information is required and all reference block information is available, it cannot be directly determined whether the sample is filterable, and it is necessary to further determine whether the sample is filterable or not based on a situation of the reference sample.

[0129] In an optional embodiment, whether reference block information is required during the filtering process for the sample within the current filtering unit in this embodiment can be determined according to a preset filtering manner, where the preset filtering manner includes at least one of deblocking filtering, sample adaptive offset, enhanced sample adaptive offset, cross-component sample adaptive offset, or adaptive loop filtering.

[0130] In this embodiment, the preset filtering manner is first explained using deblocking filtering as an example. Referring to FIG. 5, according to the deblocking filtering manner, a P block and a Q block on both sides of the boundary are used when calculating a boundary strength. During the deblocking filtering process, coding information of the reference blocks, including a prediction manner, a non-zero transform coefficient, a reference image, and a motion vector, is used. When filtering a sample using this filtering manner, the sample uses reference block information, and the two reference blocks, P block and Q block, will be needed.

[0131] In this embodiment, further explanation is provided using sample adaptive offset filtering as another preset filtering manner. The sample adaptive offset filtering manner includes various different modes, such as edge offset, band offset, and parameter fusion (merge). Taking the parameter fusion mode as an example, referring to FIG. 8, the parameter fusion mode means that for a Sample Adaptive Offset (SAO) filtering unit, its SAO parameters directly use SAO parameters of a neighboring filtering unit (left or above), and its reference block is a SAO filtering unit on the left or above. A, B, and C all represent SAO filtering units. When the C block makes SAO parameter decisions, there can be three situations: (a) directly using parameters of block A; (b) directly using parameters of block B; (c) selecting parameters different from those of blocks A and B by analyzing the characteristics of its own pixel block. When filtering a sample using this filtering manner, the sample uses reference block information, and taking the case where the C block makes SAO parameter decisions as an example, two reference blocks, block A and block B, are needed. The above are examples illustrating whether different filtering manners require reference block information. In specific implementations, it can also be determined whether reference block information is required during the filtering process for the sample according to other manners, and this embodiment does not impose restrictions on this.

[0132] In an optional embodiment, in this embodiment, whether the reference block required by the sample is available can be further determined based on a permission of the sample and a permission of the reference block. Specifically, for a certain sample within the current filtering unit, it is necessary to obtain a permission of the sample and a permission of the reference block required by the sample. The corresponding permission can be determined through a permission region where the sample is located. The reference block also includes multiple samples, and the permission of the reference block is the highest permission of all samples within the reference block. For example, if there are samples Q, W, and E within the reference block, and assuming that sample Q has the highest permission, the permission of the reference block is consistent with the permission of sample Q. Then, the permission of the

reference block is first compared with the permission of the sample. If the permission of the reference block is lower than or equal to the permission of the sample, it is further determined whether the following preset conditions are met. The preset condition can be that the reference block has been decoded or coded, i.e., for the coding device, it is also necessary to ensure that the reference block has been coded, or for the decoding device, it is necessary to ensure that the reference block has been decoded. The preset condition can further be that the reference block and the sample are located in a same Slice or Tile. The preset condition can further be that the reference block and the sample are located in different Slices or Tiles, and performing filtering operation on boundaries of Slices or Tiles is allowed. If both the criteria for determining the permission level and any of the preset conditions are met simultaneously, the reference block is considered available.

[0133] In an optional embodiment, in this embodiment, after determining an unavailable reference block, information of the unavailable reference block can also be replaced by derived reference block information, thereby converting the unavailable reference block into an available reference block. In this embodiment, an implementation for the derived reference block information is provided, specifically for the coding device, the reference block information is derived through coded blocks around the unavailable reference block, and for the decoding device, the reference block information is derived through decoded blocks around the unavailable reference block. It should be emphasized that since the permission of the unavailable reference block is usually higher than the permission of the sample, in this embodiment, the permission of the decoded blocks or coded blocks need to be lower than or equal to the permission of the sample.

[0134] In this embodiment, another implementation is further provided, specifically deriving the reference block information through preset information, where the preset information includes a default filtering parameter, a prediction manner, a non-zero transform coefficient, a reference image, and a motion vector.

[0135] Step S302: taking a sample within the current filtering unit, for which reference samples are required in the filtering process while at least one reference sample is unavailable, as the non-filterable sample.

[0136] Step S303: taking a sample within the current filtering unit, for which reference samples are required in the filtering process while all reference samples are unavailable, as the non-filterable sample.

[0137] Step S304: taking a sample within the current filtering unit, for which reference samples are required in the filtering process while a proportion of unavailable reference

samples among all reference samples is greater than or equal to a preset value, as the non-filterable sample.

[0138] In an optional embodiment, in this embodiment, filterable samples and non-filterable samples within the current filtering unit can further be determined according to reference sample information, where a reference sample is a sample required for making a filtering decision and/or performing a filtering operation.

[0139] In an optional embodiment, for a specific sample within the current filtering unit, it is necessary to first obtain its reference samples. After determining the reference samples, it is further determined whether the reference samples are available. Finally, based on the determination result of whether the reference samples are available, it is determined whether the sample is a non-filterable sample. The specific determination process can be implemented in the following manner.

[0140] An implementation is provided in this embodiment. When it is determined that the reference samples are required and at least one reference sample is unavailable, the sample is identified as a non-filterable sample. For example, if the reference samples corresponding to sample S within the current filtering unit are S1, S2, and S3, and it is assumed that at least one of S1, S2, and S3 is an unavailable reference sample, then sample S can be identified as a non-filterable sample.

[0141] Another implementation is provided in this embodiment. When it is determined that the reference samples are required and all reference samples are unavailable, the sample is identified as a non-filterable sample. For example, if the reference samples corresponding to sample S within the current filtering unit are S1, S2, and S3, and it is assumed that S1, S2, and S3 are all unavailable reference samples, then sample S can be identified as a non-filterable sample.

[0142] Another implementation is provided in this embodiment. When it is determined that reference samples are required and a proportion of unavailable reference samples is greater than or equal to a preset value, the sample is identified as a non-filterable sample. For example, if the reference samples corresponding to sample S within the current filtering unit are S1, S2, and S3, and it is assumed that S1 and S2 are unavailable reference samples and S3 is an available reference sample, then the proportion of unavailable reference samples can be calculated as $2/3$. If a preset value for the proportion is 50%, it can be determined that the proportion of unavailable reference samples is greater than the preset value, and thus sample S can be identified as a non-filterable sample. The preset value can be set according to actual filtering requirements and is not limited in this embodiment.

[0143] Furthermore, in this embodiment, the reference samples corresponding to the sample can be determined based on different preset filtering manners. This embodiment further provides examples for various different filtering manners.

[0144] In an optional embodiment, the filtering manner is explained as deblocking filtering. The deblocking filtering module processes 8x8 block boundary among all PU (Prediction Unit) and TU (Transform Unit) boundaries, involving two stages: filtering decision and filtering operation. Firstly, the filtering decision is made to determine a boundary filtering strength (no filtering, weak filtering, or strong filtering) and filtering parameters. Subsequently, the filtering operation is performed, i.e., pixels are corrected accordingly to the selected filter strength and filter parameters. The deblocking filter algorithm has the ability to adapt to different video contents and different coding parameters, i.e., it can adaptively choose whether to filter and the filtering strength for different block boundaries. For example, strong filtering is applied to discontinuous boundaries in smooth regions, while weak filtering or no filtering is applied to boundaries in textured regions. In addition, the filter parameters (β and tc) are allowed to be adjusted according to individual characteristics of different video sequences at a slice level, i.e., the filter parameters are fine-tuned by increasing an offset value, so that the filtering intensity can be increased or decreased, the decoded video quality can be optimized, and a better effect than a default value can be obtained. It is important to note that while the smallest unit for deblocking filtering is 8x8 block boundary, in practice, the 8x8 block boundary is divided into two parts for independent deblocking filtering. As shown in FIGS. 5 and 6, a vertical boundary takes 8x4 as the basic unit and a horizontal boundary takes 4x8 as the basic unit. Taking deblocking filtering manner of vertical boundary as an example, as shown in FIG. 5, reference samples are the samples needed during the filtering process, which are the samples required for calculating boundary strength and implementing filtering operations. Specifically, depending on different filtering operations (e.g., strong luminance filtering, weak luminance filtering, and chrominance filtering), reference samples can be all or some of the samples shown in FIG. 5. Furthermore, deblocking filtering can also be applied to the horizontal boundary, as shown in FIG. 6. Similarly, depending on different filtering operations, reference samples can be all or some of the samples in blocks P and Q shown in FIG. 6.

[0145] In an optional embodiment, the filtering manner is explained as sample adaptive offset. Sample adaptive offset takes CTB (Coding Tree Block) as the basic unit, classifying reconstructed pixels using a suitable classifier, and then using different compensation values for different types of pixels, the subjective and objective quality of video can be effectively improved. SAO includes two types of compensation, namely edge offset (EO) and band offset

(BO). In addition, in order to save the bit rate of coding parameters, parameter fusion (merge) technology is also introduced. Edge offset technology is to classify a current pixel by comparing a current pixel value with neighboring pixel values, and then compensate the same value for the same type of pixels. The compensation values for different types of pixels can be different. According to position differences of selected neighboring pixels, the edge offset is divided into four modes: horizontal (EO_0), vertical (EO_1), 135-degree (EO_2), and 45-degree (EO_3), as shown in FIG. 7, where c represents the current pixel, and a and b represent neighboring pixels. Referring to FIG. 7, for this filtering manner, a preset number of samples surrounding any sample in the current filtering unit can be used as reference samples. As shown in FIG. 7, the reference samples are the two samples surrounding the current sample, with c being the current sample and a and b being the reference samples. Furthermore, band offset is classified according to a pixel intensity value, a pixel range is equally divided into 32 bands, each band is compensated according to its own characteristics, and the same band uses the same compensation value. For example, for an 8-bit image, a range of pixel values is [0, 255], which is divided into 32 bands, each of which contains 8 pixel values, and an index range of sidebands is [0, 31]. The parameter fusion mode means that for a CTB, its SAO parameters directly use SAO parameters of neighboring blocks (left or top), and this mode does not involve reference samples, which can be seen in the process of obtaining reference blocks described above.

[0146] In an optional embodiment, the filtering manner is explained as enhanced sample adaptive offset. When the enhanced sample adaptive offset manner is adopted, for luminance samples, all pixels are first divided into C1 and C2 categories according to two dimensions, and a final classification result $C_t = C_1 \times C_2$ is obtained by taking a Cartesian product of the categories of the two dimensions. For chrominance samples, only the second dimension (i.e., C2) is used for classification, with the final classification result C_t obtained through a lookup table. Each category of samples has an offset value, which is transmitted in a bit stream. ESAO supports both image level control switch and LCU (Largest Coding Unit) level control switch. For the C1 category of the first dimension, a classification template is as shown in FIG. 9. For the current pixel, an initial category is set to $c_x = 0$, and then it is compared with the surrounding 8 pixels from left to right and top to bottom. If a surrounding pixel is larger than the current pixel, c_x is incremented by 1; otherwise, c_x is decremented by 1 (or remains unchanged in the second method). After traversal, the final category of the current pixel is obtained. Therefore, a value range of c_x is [-8, 8] (the second method is [0, 8]), so $C_1 = 17$ (the second method is 9). The choice between the two methods is decided by the coding device, with identification

information transmitted in an image header. For the C2 category of the second dimension, pixel values within a range of $[0, (1 \ll \text{bitdepth}) - 1]$ are uniformly divided into C2 categories according to a size interval. The value of C2 is determined by the coding device and transmitted in the image header. Assuming the current pixel value is y , the classification formula is $cy = (y * C2) \gg \text{bitdepth}$, where cy is a specific category within $[0, C2]$. Referring to FIG. 9, for this filtering manner, a preset number of samples surrounding any sample in the current filtering unit can also be used as reference samples. As shown in FIG. 9, the reference samples are eight samples surrounding the current sample. For example, the shaded portion in FIG. 9 is the current sample, and the eight non-shaded portions surrounding the shaded portion are the reference samples of the current sample.

[0147] In an optional embodiment, the filtering manner is explained as cross-component sample adaptive offset. Cross-component sample adaptive offset is to classify a current chrominance sample according to a corresponding luminance sample value, and then superimpose an offset value according to the category. This technology only supports the classification method of BO (Band Offset). Specifically, a value range of the luminance sample is uniformly divided into BandNum bands, with each band representing a category. A band category into which the luminance sample value corresponding to the current chrominance sample falls is used as the category for the current chrominance sample, and then the offset value corresponding to the category is superimposed on the chrominance sample. Referring to FIG. 10, for this filtering manner, a preset number of samples surrounding any sample in the current filtering unit can also be used as reference samples. In this filtering manner, the sample and reference sample are in different components. For example, the reference sample for the current chrominance sample is one of the 9 surrounding luminance samples, as shown in FIG. 10, where c is the current chrominance sample, and $0 \sim 8$ are the 9 surrounding luminance samples, and one luminance sample is selected as the reference sample of the current chromaticity sample.

[0148] In an optional embodiment, the filtering manner is explained as adaptive loop filtering. For a luminance component in adaptive loop filtering, a frame of image is divided into 4×4 coefficient regions (as shown in FIG. 14), and pixels within the same coefficient region use the same set of filter coefficients. According to a scanning order of the coefficient regions, two neighboring coefficient regions can be merged into a new region, and the merged region can continue to be merged with its neighboring region until the whole frame of image is one coefficient region. Thus, the number of filter coefficient groups for a frame of an image is equal to the number of coefficient regions after merging, and the value range is $[1, 16]$. The number

of filter coefficient groups (i.e. the number of merged coefficient regions), the size of each region (i.e. how many original coefficient regions each merged region contains), and each group of filter coefficients that need to be transmitted in the bit stream for the current frame image. For the chrominance component, one frame of image uses one filter (i.e., the entire frame of the image corresponds to a coefficient region and a group of filter coefficients), and the Cb and Cr components have their own filters. Referring to FIGS. 11 and 12, in which the shape of the adaptive loop filter is shown in FIG. 11 and the shape of the enhanced adaptive loop filter is shown in FIG. 12, for this filtering manner, surrounding samples centered at the current sample and within the filter coverage range can be used as reference samples, and the surrounding samples within the filter coverage range, i.e., the reference samples, are shown in FIGS. 11 and 12. The filter coverage range can be set based on different standards and is not limited in this embodiment.

[0149] After determining the reference samples, whether the reference samples required by the sample are available can be further determined based on a permission of the sample and a permission of the reference samples. Specifically, for a certain sample within the current filtering unit, the permission of the sample and the permission of the reference samples required by the sample need to be obtained. The corresponding permission can be determined based on the permission region where the sample is located and the permission region where the reference samples is located. Then, the permission of the reference samples is compared with the permission of the sample. If the permission of the reference samples is lower than or equal to the permission of the sample, it is further determined whether or not the following predetermined conditions are met. The preset condition can be that the reference samples have been decoded or coded, i.e., for the coding device, it is also necessary to ensure that the reference samples have been coded, or for the decoding device, it is necessary to ensure that the reference samples have been decoded. The preset condition can further be that the reference samples and the sample are located in a same Slice or Tile. The preset condition can further be that the reference samples and the sample are located in different Slices or Tiles, and performing filtering operation on boundaries of Slices or Tiles is allowed. If both the criteria for determining the permission level and any of the preset conditions are met simultaneously, the reference sample(s) is considered available.

[0150] Further, in this embodiment, after determining unavailable reference samples, information of the unavailable reference samples can also be replaced by derived reference sample information, thereby converting the unavailable reference samples into available reference samples. In this embodiment, an implementation for the derived reference sample

information is provided, specifically for the coding device, the reference sample information is derived through coded samples around the unavailable reference samples, and for the decoding device, the reference sample information is derived through decoded samples around the unavailable reference samples. It should be emphasized that since the permission of the unavailable reference samples is usually higher than the permission of the sample, in this embodiment, the permission of the decoded samples or coded samples need to be lower than or equal to the permission of the sample.

[0151] In this embodiment, another implementation is further provided, specifically deriving the reference sample information through preset information, where the preset information includes a default sample value.

[0152] In this embodiment, non-filterable samples and filterable samples within the current filtering unit are classified through reference block information or reference sample information. At the same time, an unavailable reference block or an unavailable reference sample can also be converted into an available reference block or an available reference sample. Through the above manners, non-filterable samples and filterable samples can be accurately identified, and then corresponding filtering strategies can be executed, effectively improving the subjective and objective quality of the image.

[0153] Referring to FIG. 13, FIG. 13 is a flow diagram of a second embodiment of a filtering method of the present disclosure.

[0154] Based on the first embodiment mentioned above, the filtering method in this embodiment further includes the following steps.

[0155] Step S401: not performing the filtering operation on one or more first samples within the current filtering unit and performing the filtering operation on one or more second samples within the current filtering unit.

[0156] The preset filtering manner in this embodiment includes at least one of deblocking filtering, sample adaptive offset, enhanced sample adaptive offset, cross-component sample adaptive offset, or adaptive loop filtering. The specific process of the filtering operation can be implemented as follows.

[0157] In an optional embodiment, when the filtering strategy is partial filtering, regardless of the number of non-filterable samples and filterable samples within the current filtering unit, the filtering operation performed is to not perform the filtering operation on the one or more first samples (non-filterable samples) within the filtering unit, and to perform the filtering operation on the one or more second samples (filterable samples) within the filtering unit. Assuming that there are N non-filterable samples and M filterable samples within the current

filtering unit, the filtering operation is not performed on these N non-filterable samples, and the filtering operation is performed on these M filterable samples.

[0158] In an optional embodiment, when the filtering strategy is no filtering, in this case, it is necessary to first determine whether there is one or more non-filterable samples within the current filtering unit. If there is one or more non-filterable samples, the filtering operation is not performed on all samples within the current filtering unit. Assuming that there are N non-filterable samples and M filterable samples within the current filtering unit, the filtering operation is not performed on all samples within the current filtering unit, i.e., $M+N$ samples.

[0159] In an optional embodiment, when the filtering strategy is full filtering, in this case, it is necessary to first determine whether there is one or more filterable samples within the current filtering unit. If there is one or more filterable samples, the filtering operation is performed on all samples within the current filtering unit. Furthermore, whether there is one or more filterable samples within the current filtering unit includes two situations. The first situation is that all the samples within the current filtering unit are filterable samples. For example, if the current filtering unit contains N samples and all of these N samples are filterable, the filtering operation can be performed on these N filterable samples. The second situation is that there is one or more non-filterable samples within the current filtering unit. For this situation, when executing the full filtering strategy, it is necessary to first convert all non-filterable samples within the current filtering unit into filterable samples, and then perform the filtering operation on all filterable samples. For example, if the current filtering unit contains N filterable samples and M non-filterable samples, these M non-filterable samples are first converted into M filterable samples, and then the filtering operation is performed on these $N+M$ filterable samples.

[0160] In an optional embodiment, a non-filterable sample within the current filtering unit can be converted into a filterable sample by deriving information of the reference blocks or reference samples, where the reference blocks are neighboring blocks that are required to be used in the sample filtering process and the reference samples are surrounding samples that are required to be used in the sample filtering process. In this embodiment, an implementation for obtaining the information of the derived reference blocks or reference samples is provided, specifically, the information of the reference blocks or reference samples is derived from available blocks or available samples inside and around (surrounding) the permission region. The available blocks or available samples can be in the same permission region or in different permission regions as the non-filterable samples, which is not limited in this embodiment. A permission of the available blocks or available samples need to be lower than or equal to a permission of the non-filterable samples. For the decoding device, the available blocks or

available samples are decoded blocks or samples, and for the coding device, the available blocks or available samples are coded blocks or samples.

[0161] In this embodiment, another implementation for obtaining the information of the derived reference blocks or reference samples is provided, specifically, the information of the reference blocks or reference samples is derived from preset information, where the preset information includes a default sample value, a filtering parameter, a prediction manner, a non-zero transform coefficient, a reference image, and a motion vector, and other information.

[0162] This embodiment realizes constrained filtering operations on permission regions by performing different filtering operations on samples with different permission levels in various permission regions according to the filtering strategy, thereby improving the subjective and objective quality of the image.

[0163] Referring to FIG. 14, FIG. 14 is a structural block diagram of a first embodiment of a filtering apparatus of the present disclosure.

[0164] As shown in FIG. 14, the filtering apparatus proposed in the embodiment of the present disclosure includes:

[0165] an obtaining module 10, configured to determine a filtering strategy corresponding to a current filtering unit and a permission region where the current filtering unit is located, where the permission region is an image region with a permission level for an image therein;

[0166] an identification module 20, configured to determine a permission level of a sample within the current filtering unit according to the permission level of the permission region;

[0167] a classification module 30, configured to classify the sample within the current filtering unit according to the permission level of the sample; and

[0168] a filtering module 40, configured to perform a corresponding filtering operation on the current filtering unit according to the filtering strategy and a classification result.

[0169] In this embodiment, by determining a filtering strategy corresponding to a current filtering unit and a permission region where the current filtering unit is located, determining a permission level of a sample within the current filtering unit according to the permission level of the permission region; classifying the sample within the current filtering unit according to the permission level of the sample; and performing a corresponding filtering operation on the current filtering unit according to the filtering strategy and a classification result, so that constrained filtering operations on permission regions is realized and the subjective and objective quality of images is improved.

[0170] In an embodiment, the filtering strategy is determined with indication information parsed from a bit stream, and the indication information is transmitted at a sequence level, an

image level, a Slice level, a Tile level, or a filtering unit level.

[0171] In an embodiment, the permission region includes at least one filtering unit, the current filtering unit is allowed to span a plurality of permission regions, and the filtering unit includes at least one sample;

[0172] the identification module 20 is further configured to set the permission level of the sample within the current filtering unit to be consistent with the permission level of the permission region where the sample is located.

[0173] In an embodiment, the current filtering unit is a basic unit in a filtering process and different filtering units are allowed to perform different filtering operations;

[0174] the filtering module 40 is further configured to not perform the filtering operation on one or more first samples within the current filtering unit and perform the filtering operation on one or more second samples within the current filtering unit; or, perform the filtering operation on all samples within the current filtering unit; or, not perform the filtering operation on all the samples within the current filtering unit.

[0175] In an embodiment, each of the one or more first samples is a non-filterable sample and each of the one or more second samples is a filterable sample.

[0176] In an embodiment, the filtering module 40 is further configured to convert a non-filterable sample within the current filtering unit into a filterable sample by deriving information of reference blocks or reference samples, and perform the filtering operation on all filterable samples, where the information of the reference blocks or the reference samples is derived from available blocks or available samples within and around the permission region or is derived from preset information; where the available blocks or the available samples are blocks or samples which have a permission lower than or equal to a permission of the non-filterable samples and have been decoded or coded.

[0177] In an embodiment, the classification module 30 is further configured to take a sample within the current filtering unit, for which reference block information is required in the filtering process while at least one reference block is unavailable, as the non-filterable sample; take a sample within the current filtering unit, for which reference samples are required in the filtering process while at least one reference sample is unavailable, as the non-filterable sample; take a sample within the current filtering unit, for which reference samples are required in the filtering process while all reference samples are unavailable, as the non-filterable sample; and take a sample within the current filtering unit, for which reference samples are required in the filtering process while a proportion of unavailable reference samples among all reference samples is greater than or equal to a preset value, as the non-filterable sample.

[0178] In an embodiment, the classification module 30 is further configured to determine, according to a preset filtering manner, whether the reference block information is required in the filtering process for the sample within the current filtering unit; and determine, based on a permission of the sample and a permission of the reference block, whether the reference block is available.

[0179] In an embodiment, the reference block is a block required for making a filtering decision and/or performing the filtering operation, the reference block includes at least one sample, and the permission of the reference block is a highest permission of all samples within the reference block.

[0180] In an embodiment, the classification module 30 is further configured to determine that the reference block is available in response to determining that the permission of the reference block is lower than or equal to the permission of the sample and at least one of following preset conditions is met; where the preset conditions include: the reference block has been decoded or coded; the reference block and the sample are located in a same Slice or Tile; or the reference block and the sample are located in different Slices or Tiles, and performing filtering operation on boundaries of Slices or Tiles is allowed.

[0181] In an embodiment, the filtering apparatus further includes a conversion module;

[0182] the conversion module is configured to derive the reference block information from decoded blocks or coded blocks with a permission level lower than or equal to the permission of the sample around an unavailable reference block, and/or derive the reference block information from preset information; and replace information of the unavailable reference block with the derived reference block information to convert the unavailable reference block into an available reference block.

[0183] In an embodiment, the classification module 30 is further configured to acquire reference samples corresponding to any one sample within the current filtering unit according to a preset filtering manner; and determine, based on a permission of the sample and a permission of the reference samples, whether the reference samples are available.

[0184] In an embodiment, the reference samples are samples required for making a filtering decision and/or performing the filtering operation.

[0185] In an embodiment, the classification module 30 is further configured to determine that the reference samples are available in response to determining that the permission of the reference samples is lower than or equal to the permission of the sample and at least one of following preset conditions is met; where the preset conditions include: the reference samples have been decoded or coded; the reference samples and the sample are located in a same slice

or tile; or the reference samples and the sample are located in different Slices or Tiles, and performing filtering operation on Slice or Tile boundaries is allowed.

[0186] In an embodiment, the filtering apparatus further includes a conversion module;

[0187] the conversion module is configured to derive reference sample information from decoded samples or coded samples with a permission level lower than or equal to the permission of the sample around unavailable reference samples, and/or derive the reference sample information from preset information; and replace information of the unavailable reference samples with the derived reference sample information to convert the unavailable reference samples into available reference samples.

[0188] In an embodiment, the preset filtering manner includes at least one of: deblocking filtering; sample adaptive offset; enhanced sample adaptive offset; cross-component sample adaptive offset; or adaptive loop filtering.

[0189] In addition, in order to achieve the above objectives, the present disclosure further proposes a filtering device, including: a memory, a processor, and a filtering program stored on the memory and executable on the processor, and the filtering program is configured to implement the filtering method as described above.

[0190] In addition, an embodiment of the present disclosure further provides a storage medium, on which a filtering program is stored, and when the filtering program is executed by a processor, the steps of the filtering method described above are implemented.

[0191] Since the storage medium adopts all the technical solutions of all the above embodiments, it has at least all the beneficial effects brought by the technical solutions of the above embodiments, which will not be described one by one here.

[0192] It should be understood that the above is only an example and does not constitute any limitation on the technical solution of the present disclosure. In specific applications, those skilled in the art can make settings according to their needs, and the present disclosure imposes no restrictions in this regard.

[0193] It should be noted that the workflow described above is only schematic, and does not limit the scope of protection of the present disclosure. In practical application, those skilled in the art can choose some or all of them to achieve the purpose of this embodiment, and there is no limitation here.

[0194] In addition, for technical details that are not described in detail in this embodiment, reference can be made to the filtering method provided in any embodiment of the present disclosure, and will not be repeated here.

[0195] In addition, it should be noted that, as used herein, the terms “including”, “containing”

or any other variations thereof are intended to cover non-exclusive inclusion, so that a process, method, article, or system including a series of elements not only includes those elements, but also includes other elements not explicitly listed or elements inherent to such process, method, article, or system. Without further limitations, an element defined by the phrase “including one” does not exclude the existence of other identical elements in the process, method, object or system including the element.

[0196] The serial numbers of the embodiments of the present disclosure mentioned above are solely for descriptive purposes and do not represent the superiority or inferiority of the embodiments.

[0197] Through the description of the above embodiments, those skilled in the art can clearly understand that the method according to the above embodiments can be realized by means of software and necessary general hardware platform, and of course it can also be realized by hardware, but in many cases the former is a better embodiment. Based on this understanding, the technical solution of the present disclosure can be embodied in the form of a software product, which is stored in a storage medium (such as a Read Only Memory (ROM)/RAM, magnetic disk and optical disk) and includes several instructions to make a terminal device (which can be a mobile phone, a computer, a server or a network device, etc.) execute the methods described in various embodiments of the present disclosure.

[0198] The above are merely preferred embodiments of the present disclosure and do not limit the patent scope of the present disclosure. Any equivalent structural or equivalent process transformations made based on the contents of the specification and accompanying drawings of the present disclosure, or direct or indirect applications in other related technical fields, are similarly included in the patent protection scope of the present disclosure.

CLAIMS

1. A filtering method, comprising:

determining a filtering strategy corresponding to a current filtering unit and a permission region where the current filtering unit is located, wherein the permission region is an image region with a permission level for an image therein;

determining a permission level of a sample within the current filtering unit according to the permission level of the permission region;

classifying the sample within the current filtering unit according to the permission level of the sample; and

performing a corresponding filtering operation on the current filtering unit according to the filtering strategy and a classification result.

2. The filtering method according to claim 1, wherein the permission region comprises at least one filtering unit, the current filtering unit is allowed to span a plurality of permission regions, and the filtering unit comprises at least one sample;

wherein the determining a permission level of a sample within the current filtering unit according to the permission level of the permission region comprises:

setting the permission level of the sample within the current filtering unit to be consistent with the permission level of the permission region where the sample is located.

3. The filtering method according to claim 2, wherein the current filtering unit is a basic unit in a filtering process and different filtering units are allowed to perform different filtering operations;

wherein the performing a corresponding filtering operation on the current filtering unit according to the filtering strategy and a classification result comprises:

not performing the filtering operation on one or more first samples within the current filtering unit and performing the filtering operation on one or more second samples within the current filtering unit.

4. The filtering method according to claim 3, wherein each of the one or more first samples is a non-filterable sample and each of the one or more second samples is a filterable sample.

5. The filtering method according to claim 4, comprising at least one of:

taking a sample within the current filtering unit, for which reference block information is required in the filtering process while at least one reference block is unavailable, as the non-filterable sample;

taking a sample within the current filtering unit, for which reference samples are required in the filtering process while at least one reference sample is unavailable, as the non-filterable sample;

taking a sample within the current filtering unit, for which reference samples are required in the filtering process while all reference samples are unavailable, as the non-filterable sample;
or

taking a sample within the current filtering unit, for which reference samples are required in the filtering process while a proportion of unavailable reference samples among all reference samples is greater than or equal to a preset value, as the non-filterable sample.

6. The filtering method according to claim 5, further comprising:

determining, according to a preset filtering manner, whether the reference block information is required in the filtering process for the sample within the current filtering unit;
and

determining, based on a permission of the sample and a permission of the reference block, whether the reference block is available.

7. The filtering method according to claim 6, wherein the reference block is a block required for making a filtering decision or performing the filtering operation or both, the reference block comprises at least one sample, and the permission of the reference block is a highest permission of all samples within the reference block.

8. The filtering method according to claim 6, wherein the determining, based on a permission of the sample and a permission of the reference block, whether the reference block is available comprises:

determining that the reference block is available in response to determining that the permission of the reference block is lower than or equal to the permission of the sample and at least one of following preset conditions is met;

wherein the preset conditions comprise:

the reference block has been decoded or coded;

the reference block and the sample are located in a same Slice or Tile; or

the reference block and the sample are located in different Slices or Tiles, and performing filtering operation on boundaries of Slices or Tiles is allowed.

9. The filtering method according to claim 5, further comprising:

acquiring reference samples corresponding to any one sample within the current filtering unit according to a preset filtering manner; and

determining, based on a permission of the sample and a permission of the reference samples, whether the reference samples are available.

10. The filtering method according to claim 9, wherein the reference samples are samples required for making a filtering decision or performing the filtering operation or both.

11. The filtering method according to claim 9, wherein the determining, based on a permission of the sample and a permission of the reference samples, whether the reference samples are available comprises:

determining that the reference samples are available in response to determining that the permission of the reference samples is lower than or equal to the permission of the sample and at least one of following preset conditions is met;

wherein the preset conditions comprise:

the reference samples have been decoded or coded;

the reference samples and the sample are located in a same slice or tile; or

the reference samples and the sample are located in different Slices or Tiles, and performing filtering operation on Slice or Tile boundaries is allowed.

12. The filtering method according to claim 6 or 9, wherein the preset filtering manner comprises at least one of:

deblocking filtering;

sample adaptive offset;

enhanced sample adaptive offset;

cross-component sample adaptive offset; or

adaptive loop filtering.

13. A filtering device, comprising a memory, a processor and a filtering program stored in the memory and running on the processor, wherein the filtering program is configured to implement the filtering method according to any one of claims 1 to 12.

14. A storage medium, wherein a filtering program is stored on the storage medium, and when the filtering program is executed by a processor, the filtering method according to any one of claims 1 to 12 is implemented.

15. A filtering apparatus, comprising:

an obtaining module, configured to determine a filtering strategy corresponding to a current filtering unit and a permission region where the current filtering unit is located, wherein the permission region is an image region with a permission level for an image therein;

an identification module, configured to determine a permission level of a sample within the current filtering unit according to the permission level of the permission region;

a classification module, configured to classify the sample within the current filtering unit according to the permission level of the sample; and

a filtering module, configured to perform a corresponding filtering operation on the current filtering unit according to the filtering strategy and a classification result.