

IMAGE PROCESSING METHOD AND APPARATUS, AND ELECTRONIC DEVICE AND STORAGE MEDIUM

TECHNICAL FIELD

[0001] The present disclosure relates to the field of picture processing, particularly to picture processing methods and apparatuses, electronic devices, and storage media.

BACKGROUND

[0002] In video encoding and decoding, the purpose of filtering processing is to achieve smooth noise reduction and detail removal within a video picture block, and to preserve picture edges to the greatest extent. Usually, electronic devices perform filtering processing on the boundaries of multiple regions in a picture to preserve the picture edges. However, different users have different viewing permissions for different regions in the picture. In the case where a low permission user does not have the viewing permission for a high permission region in a picture, the electronic device will not perform filtering processing on the boundary between the high permission region and the low permission region during video decoding, resulting in poor picture quality.

SUMMARY

[0003] The present disclosure provides picture processing methods and apparatuses, electronic devices, and storage media, which solve the problem of poor picture quality in different permission regions of a picture after decoding and filtering.

[0004] The present disclosure adopts the following technical solutions.

[0005] In the first aspect, the present disclosure provides a picture processing method, performed by an electronic device, including: dividing a to-be-processed picture into a plurality of picture subregions; obtaining a permission level for each of the plurality of picture subregions; and when permission levels of at least two adjacent picture subregions in the plurality of picture subregions are different, performing a filtering process on an adjoining boundary between the at least two adjacent picture subregions.

[0006] In this embodiment, the electronic device divides the to-be-processed picture into multiple picture subregions, and obtains the permission level of each picture subregion; and then determine whether the permission level of each picture subregion, and if the permission levels of at least two adjacent picture subregions in the multiple picture subregions are different, it indicates that there is a situation where after filtering process in the decoding

process, the two picture subregions may still be unfiltered, and the quality of the to-be-processed picture is not high. Therefore, based on decoding information such as the division of permission regions, the electronic device performs filtering on the adjoining boundary between at least two picture subregions again, which thereby improves the picture quality of regions with different permissions in the to-be-processed picture.

[0007] In some embodiments, the permission level includes a first permission level, a second permission level, or a zero permission level, where a picture subregion of the zero permission level can be viewed by a user of any permission level.

[0008] In some embodiments, before dividing the to-be-processed picture into the plurality of picture subregions and obtaining the permission level for each of the plurality of the picture subregions, the method includes: receiving a decoding bitstream of the to-be-processed picture; and decoding based on the decoding bitstream to obtain the decoded to-be-processed picture and decoding information of the to-be-processed picture.

[0009] In some embodiments, the filtering process includes at least one of: deblocking vertical filtering, deblocking horizontal filtering, sample adaptive offset, adaptive loop filter, or neural network filtering.

[0010] In some embodiments, there is a vertical adjoining boundary between the at least two adjacent picture subregions, and the filtering process includes deblocking vertical filtering, where performing the filtering process on the adjoining boundary between the at least two adjacent picture subregions includes: determining a target to-be-filtered region on a vertical side of a first subregion in the at least two adjacent picture subregions, where the vertical side refers to a left side or a right side of the adjoining boundary of the first subregion, and the target to-be-filtered region is any one of: a right to-be-filtered region on the adjoining boundary, a left to-be-filtered region on the adjoining boundary, or a right to-be-filtered region and a left to-be-filtered region on the adjoining boundary; and performing deblocking vertical filtering on the target to-be-filtered region.

[0011] In some embodiments, the at least two adjacent picture subregions further include a second subregion, where the second subregion is horizontally adjacent to the first subregion. Horizontal adjacency here refers to the second subregion being located on the left or right side of the first subregion in the horizontal direction. The permission levels of the at least two adjacent picture subregions being different includes: the permission level of the first subregion is lower than the permission level of the second subregion; the permission level of the first subregion is higher than the permission level of the second subregion; the permission level of the first subregion is not equal to the permission level of the second subregion; or the

permission level of the first subregion is not equal to the permission level of the second subregion, and a filtering strength of the adjoining boundary is greater than or equal to a preset filtering strength.

[0012] In some embodiments, there is a horizontal adjoining boundary between the at least two adjacent picture subregions, and the filtering process includes deblocking horizontal filtering, where performing the filtering process on the adjoining boundary between the at least two adjacent picture subregions includes: determining a target to-be-filtered region on a horizontal side of a first subregion in the at least two adjacent picture subregions, where the horizontal side refers to an upper side or a lower side of the adjoining boundary of the first subregion, and the target to-be-filtered region is any one of: a lower to-be-filtered region on the adjoining boundary, an upper to-be-filtered region on the adjoining boundary, or a lower to-be-filtered region and an upper to-be-filtered region on the adjoining boundary; and performing deblocking horizontal filtering on the target to-be-filtered region.

[0013] In some embodiments, the at least two adjacent picture subregions further include a third subregion, where the third subregion is vertically adjacent to the first subregion. Vertical adjacency here refers to the third subregion being located on the upper or lower side of the first subregion in the horizontal direction. The permission levels of the at least two adjacent picture subregions being different includes: the permission level of the first subregion is lower than the permission level of the third subregion; the permission level of the first subregion is higher than the permission level of the third subregion; the permission level of the first subregion is not equal to the permission level of the third subregion; or the permission level of the first subregion is not equal to the permission level of the third subregion, and a filtering strength of the adjoining boundary is greater than or equal to a preset filtering strength.

[0014] In some embodiments, the filtering processing includes sample adaptive offset, and performing the filtering process on the adjoining boundary between the at least two adjacent picture subregions includes: determining a target to-be-filtered region on a surrounding side of a first subregion in the at least two adjacent picture subregions according to a set filtering sequence, where the surrounding side sequentially includes: a left side, an upper side, a right side, and a lower side on the adjoining boundary between the first subregion and other subregions, the permission level of the first subregion is different from the permission levels of the other subregions, and the target to-be-filtered region is any one of: a first side region on the adjoining boundary within the first subregion, a second side region on the adjoining boundary outside the first subregion, or the first side region and the second side region; and

performing sample adaptive offset on the target to-be-filtered region according to the set filtering sequence.

[0015] In some embodiments, the filtering processing includes adaptive loop filter, and performing the filtering process on the adjoining boundary between the at least two adjacent picture subregions includes: determining a target to-be-filtered region on a surrounding side of a first subregion in the at least two adjacent picture subregions according to a set filtering sequence, where the surrounding side sequentially includes: a left side, an upper side, a right side, and a lower side on the adjoining boundary between the first subregion and other subregions, the permission level of the first subregion is different from the permission levels of the other subregions, and the target to-be-filtered region is any one of: a first side region on the adjoining boundary within the first subregion, a second side region on the adjoining boundary outside the first subregion, or the first side region and the second side region; and performing adaptive loop filter on the target to-be-filtered region according to the set filtering sequence.

[0016] In some embodiments, the method further includes: according to the decoding information of the to-be-processed picture, determining whether neural network-based filtering is required for a first subregion of the at least two adjacent picture subregions; in a case where neural network-based filtering is required for the first subregion, inputting a reconstructed pixel value and a permission level of the first subregion into a neural network; and obtaining an outputted filtered reconstructed pixel of the first subregion.

[0017] In some embodiments, in a case where the neural network-based filtering is required for the first subregion, inputting the reconstructed pixel value and the permission level of the first subregion into the neural network includes: inputting the permission level of at least one picture subregion adjacent to the first subregion into the neural network.

[0018] In some embodiments, the electronic device obtains, based on a permission level of a user, information of a picture subregion lower than or equal to the permission level of the user, where the information of the picture subregion includes a pixel value of the picture subregion.

[0019] In some embodiments, when the permission levels of the at least two adjacent picture subregions in the plurality of picture subregions are different, performing the filtering process on the adjoining boundary between the at least two adjacent picture subregions includes: performing a filtering process for a luma channel component on the adjoining boundary between the at least two picture subregions; performing a filtering process for a chroma channel component on the adjoining boundary between the at least two picture subregions; or

performing a filtering process for a luma channel component and a chroma channel component on the adjoining boundary between the at least two picture subregions.

[0020] In the second aspect, a picture processing apparatus is provided, including: a picture dividing unit, a picture filtering unit, and a picture decoding unit; where the picture dividing unit is configured to divide a to-be-processed picture into a plurality of picture subregions; the picture decoding unit is configured to obtain a permission level for each of the plurality of picture subregions; and the picture filtering unit is configured to, when permission levels of at least two adjacent picture subregions in the plurality of picture subregions are different, perform a filtering process on an adjoining boundary between the at least two adjacent picture subregions.

[0021] In some embodiments, the permission level of each picture subregion is configured to indicate whether the picture subregion is a set permission region, and if the picture subregion is a permission region, the electronic device determines the permission level of the picture subregion; where in the permission level includes a first permission level and a second permission level.

[0022] In some embodiments, the picture decoding unit is further configured to receive the decoding bitstream of the to-be-processed picture, and decode based on the decoding bitstream to obtain the decoded to-be-processed picture.

[0023] In some embodiments, the picture filtering unit is further configured to determine a target to-be-filtered region on a vertical side of a first subregion in the at least two adjacent picture subregions, where the vertical side refers to a left side or a right side of the adjoining boundary of the first subregion, and the target to-be-filtered region is any one of: a right to-be-filtered region on the adjoining boundary, a left to-be-filtered region on the adjoining boundary, or a right to-be-filtered region and a left to-be-filtered region on the adjoining boundary; and the picture filtering unit is further configured to perform deblocking vertical filtering on the target to-be-filtered region.

[0024] In some embodiments, the at least two adjacent picture subregions further include a second subregion, where the second subregion is horizontally adjacent to the first subregion. Horizontal adjacency here refers to the second subregion being located on the left or right side of the first subregion in the horizontal direction. The permission levels of the at least two adjacent picture subregions being different includes: the permission level of the first subregion is lower than the permission level of the second subregion; the permission level of the first subregion is higher than the permission level of the second subregion; the permission level of the first subregion is not equal to the permission level of the second subregion; or the

permission level of the first subregion is not equal to the permission level of the second subregion, and a filtering strength of the adjoining boundary is greater than or equal to a preset filtering strength.

[0025] In some embodiments, the picture filtering unit is further configured to determine a target to-be-filtered region on a horizontal side of a first subregion in the at least two adjacent picture subregions, where the horizontal side refers to an upper side or a lower side of the adjoining boundary of the first subregion, and the target to-be-filtered region is any one of: a lower to-be-filtered region on the adjoining boundary, an upper to-be-filtered region on the adjoining boundary, or a lower to-be-filtered region and an upper to-be-filtered region on the adjoining boundary; and the picture filtering unit is further configured to perform deblocking horizontal filtering on the target to-be-filtered region.

[0026] In some embodiments, the at least two adjacent picture subregions further include a third subregion, where the third subregion is vertically adjacent to the first subregion. Vertical adjacency here refers to the third subregion being located on the upper or lower side of the first subregion in the horizontal direction. The permission levels of the at least two adjacent picture subregions being different includes: the permission level of the first subregion is lower than the permission level of the third subregion; the permission level of the first subregion is higher than the permission level of the third subregion; the permission level of the first subregion is not equal to the permission level of the third subregion; or the permission level of the first subregion is not equal to the permission level of the third subregion, and a filtering strength of the adjoining boundary is greater than or equal to a preset filtering strength.

[0027] In some embodiments, the picture filtering unit is further configured to determine a target to-be-filtered region on a surrounding side of a first subregion in the at least two adjacent picture subregions according to a set filtering sequence, where the surrounding side sequentially includes: a left side, an upper side, a right side, and a lower side on the adjoining boundary between the first subregion and other subregions, the permission level of the first subregion is different from the permission levels of the other subregions, and the target to-be-filtered region is any one of: a first side region on the adjoining boundary within the first subregion, a second side region on the adjoining boundary outside the first subregion, or the first side region and the second side region; and the picture filtering unit is further configured to perform sample adaptive offset on the target to-be-filtered region according to the set filtering sequence.

[0028] In some embodiments, the picture filtering unit is further configured to determine a target to-be-filtered region on a surrounding side of a first subregion in the at least two adjacent

picture subregions according to a set filtering sequence, where the surrounding side sequentially includes: a left side, an upper side, a right side, and a lower side on the adjoining boundary between the first subregion and other subregions, the permission level of the first subregion is different from the permission levels of the other subregions, and the target to-be-filtered region is any one of: a first side region on the adjoining boundary within the first subregion, a second side region on the adjoining boundary outside the first subregion, or the first side region and the second side region; and the picture filtering unit is further configured to perform adaptive loop filter on the target to-be-filtered region according to the set filtering sequence.

[0029] In some embodiments, the picture filtering unit is further configured to determine whether neural network-based filtering is required for the first subregion of at least two picture subregions based on the decoding information of the to-be-processed picture; when neural network-based filtering is required for the first subregion, input the reconstructed pixel values and the permission level of the first subregion into the neural network; and obtain the outputted filtered reconstructed pixels of the first subregion.

[0030] In some embodiments, the picture filtering unit is further configured to input the permission level of at least one picture subregion adjacent to the first subregion into the neural network.

[0031] In some embodiments, the picture filtering unit is further configured to perform a filtering process for a luma channel component on the adjoining boundary between the at least two picture subregions; or perform a filtering process for a chroma channel component on the adjoining boundary between the at least two picture subregions; or perform a filtering process for a luma channel component and a chroma channel component on the adjoining boundary between the at least two picture subregions.

[0032] In the third aspect, the present disclosure provides an electronic device including: one or more processors and one or more memories, where the one or more memories are configured to store computer instructions, and the one or more processors are configured to call the computer instructions from the memory and run the computer instructions to implement the picture processing method according to the first aspect and any possible implementation thereof.

[0033] In the fourth aspect, the present disclosure provides a computer-readable storage medium including: computer software instructions; where when computer software instructions are executed in a picture processing apparatus, the picture processing device

implements the picture processing method according to the first aspect and any possible implementation thereof.

[0034] In the fifth aspect, the present disclosure provides a computer program product, where when the computer program runs on a picture processing apparatus, the picture processing device implements the picture processing method according to the first aspect and any possible implementation thereof. The beneficial effects of the second to fifth aspects can refer to the corresponding description of the first aspect and will not be further elaborated.

[0035] On the basis of the implementation methods provided in the above aspects, further combinations can be made to provide more implementation methods.

BRIEF DESCRIPTION OF DRAWINGS

[0036] FIG. 1 is a schematic diagram of a framework of a picture processing flow provided in the present disclosure.

[0037] FIG. 2 is a schematic diagram of regions for deblocking filter provided in the present disclosure.

[0038] FIG. 3 is a schematic diagram of a boundary of a filtered block provided in the present disclosure.

[0039] FIG. 4 is a schematic diagram of a filter provided in the present disclosure.

[0040] FIG. 5 is a schematic structural diagram of a picture processing apparatus provided in the present disclosure.

[0041] FIG. 6 is a flowchart of a picture processing method provided in the present disclosure.

[0042] FIG. 7 is a schematic diagram of a detection region provided in the present disclosure.

[0043] FIG. 8 is a flowchart of another picture processing method provided in the present disclosure.

[0044] FIG. 9 is a schematic diagram of a filtered region provided in the present disclosure.

[0045] FIG. 10 is a schematic diagram of another filtered region provided by the present disclosure.

[0046] FIG. 11 is a flowchart of another picture processing method provided in the present disclosure.

[0047] FIG. 12 is a schematic diagram of another filtered region provided by the present disclosure.

[0048] FIG. 13 is a flowchart of another picture processing method provided in the present disclosure.

[0049] FIG. 14 is a schematic diagram of another filtered region provided by the present disclosure.

[0050] FIG. 15 is a flowchart of another picture processing method provided in the present disclosure.

[0051] FIG. 16 is a schematic diagram of another filtered region provided by the present disclosure.

[0052] FIG. 17 is a flowchart of another picture processing method provided in the present disclosure.

[0053] FIG. 18 is a flowchart of another picture processing method provided in the present disclosure.

[0054] FIG. 19 is a schematic diagram of a hardware structure of an electronic device according to the present disclosure.

DETAILED DESCRIPTION

[0055] The technical solutions in the embodiments of the present disclosure are clearly described below with reference to the accompanying drawings in the embodiments of the present disclosure. The embodiments described are merely some embodiments of the present disclosure, and not all embodiments. Other embodiments achieved by those of ordinary skill in the art based on the embodiments in the present disclosure without paying creative work shall all fall within the scope of protection of the present disclosure.

[0056] It should be noted that in the embodiments of the present disclosure, words such as "exemplary" or "for example" are used to indicate examples, embodiments, or illustrations. Any embodiment or design scheme described as "exemplary" or "for example" in the present disclosure should not be interpreted as being more preferred or advantageous than other embodiments or design schemes. Specifically, the use of words such as "exemplary" or "for example" is intended to explain relevant concepts in a concrete way.

[0057] In order to facilitate a clear description of the technical solution of the embodiments of the present disclosure, the words "first", "second", etc. are used to distinguish the same or similar items with basically the same function in the embodiments of the present disclosure. Those skilled in the art can understand that the words "first", "second", etc. are not intended to limit the quantity and execution order.

[0058] In order to enable those skilled in the art to better understand the technical solution provided by the embodiments of the present disclosure, a brief explanation of some technical

terms and the main process of video encoding and decoding involved in the embodiments of the present disclosure will be given below.

[0059] First, technical terms are described.

[0060] 1. Deblocking filter (DBF): is used to remove block boundary effects generated by block encoding, including deblocking vertical filtering and deblocking horizontal filtering.

[0061] 2. Sample adaptive offset (SAO): by classifying based on the pixel values of the samples and the gradient values of surrounding blocks, different compensation values are added to every categories of pixel values to make the reconstructed picture closer to the original picture. The basic principle of SAO is to compensate for the peak pixels in the reconstructed curve by adding negative values, and to supplement the valley pixels by adding positive values. SAO uses a Collect Transfer Unit (CTU) as a basic unit and includes two types of compensation: Edge Offset (EO) and Band Offset (BO). In addition, a parameter fusion technology is introduced. Sample adaptive offset includes basic sample adaptive offset (SAO mentioned above), enhanced sample adaptive offset (ESAO), and cross component sample adaptive offset (CCSAO, only for chroma).

[0062] 3. Adaptive loop filter (ALF): by using a Wiener filter, enhancement filtering is performed on the reconstructed picture to make the reconstructed picture closer to the original picture. Where adaptive loop filter can enable enhanced adaptive loop filter (EALF).

[0063] 4. Neural Network (NN): a neural network is an operational model composed of a large number of interconnected nodes (or neurons). In the neural network, neuron processing units can represent different objects, such as features, letters, concepts, or some meaningful abstract patterns. The types of processing units in the neural network are divided into three categories: input units, output units, and hidden units. The input units receive signals and data from the outside. The output units realize the output of processing results. The hidden units are located between the input units and the output units and cannot be observed from outside the system. The connection weight between neurons reflects the strength of the connection between neurons, and the representation and processing of information are reflected in the connection relationships between network processing units. The artificial neural network is a non programmed, brain like information processing method, the essence of which is to obtain a parallel distributed information processing function through the transform and dynamic behavior of the neural network, and imitate the information processing function of the nervous system of the human brain to different degrees and levels. Currently, in the field of video processing, a commonly used neural network includes: a Convolutional Neural Network (CNN), a Recurrent Neural Network (RNN), or a Fully Connected Network, etc.

[0064] Second, the main process of video encoding and decoding is described.

[0065] As shown in FIG. 1, taking video encoding as an example, video encoding generally includes processes such as prediction, transform, quantization, entropy encoding, etc. Furthermore, the encoding process can also be implemented according to the framework in FIG. 1.

[0066] Where prediction can be divided into intra prediction and inter prediction. Intra prediction uses the surrounding encoded blocks as references to predict the current uncoded block, which effectively removes redundancy in the spatial domain. Inter prediction uses adjacent encoded pictures to predict the current picture, which effectively removes redundancy in the temporal domain.

[0067] Transform refers to the process of transforming a picture from a spatial domain to a transform domain, which uses transform coefficients to represent the picture. The vast majority of pictures contain many flat regions and slowly changing regions. Appropriate transform can transform the picture from a dispersed distribution in the spatial domain to a relatively concentrated distribution in the transform domain, which removes the frequency domain correlation between signals, and effectively compress the bitstream with the quantization process.

[0068] Entropy encoding is a lossless encoding method that can transform a series of element symbols into a binary bitstream for transmission or storage. The input symbols may include quantized transform coefficients, motion vector information, prediction mode information, transform quantization related syntax, etc. Entropy encoding can effectively remove redundancy in video element symbols.

[0069] The above is an example of encoding. The process of video decoding and the process of video encoding are opposite. That is, video decoding usually includes entropy decoding, prediction, inverse quantization, inverse transform, filtering, etc. The implementation principles of each process are the same or similar to entropy encoding. Where filtering processing includes deblocking filter, sample adaptive offset, adaptive loop filter, and neural network-based filtering.

[0070] Third, DBF is described.

[0071] Below is a brief explanation of the implementation of DBF processing.

[0072] DBF processing includes two processes: filtering decision and filtering operation.

[0073] The filtering decision includes: 1) obtaining the boundary strength (BS) value; 2) filter switch decision; and 3) filter strength selection. For chroma, only step 1) exists, and the BS

value of luma is directly reused. For chroma, filtering operation is performed only when the BS value is 2 (i.e. at least one of the picture regions on both sides adopts the intra mode).

[0074] The filtering operation includes: 1) strong filtering and weak filtering for luma; and 2) filtering for chroma.

[0075] A to-be-filtered boundary that meets one of the following conditions 1-6 does not require DBF.

[0076] Condition 1: if the to-be-filtered boundary is a picture boundary, the boundary does not require filtering.

[0077] Condition 2: if the to-be-filtered boundary is a slice boundary and the enabling flag for the cross slice loop filter is 0, the boundary does not require filtering.

[0078] Condition 3: if the to-be-filtered boundary is a luma filtering boundary and the luma filtering boundary is not the boundary of the luma coding block or luma transform block, the boundary does not require filtering.

[0079] Condition 4: if the to-be-filtered boundary is a chroma filtering boundary, and the chroma filtering boundary is not the boundary of the chroma coding block or chroma transform block, and the luma filtering boundary corresponding to the chroma filtering boundary is not the boundary of the luma coding block, the boundary does not require filtering.

[0080] Condition 5: if the to-be-filtered boundary is a luma filtering boundary, and the sub block transform flag of the encoding unit where the luma filtering boundary is located is 1, and the luma filtering boundary is not the boundary of the luma coding block, the boundary does not require filtering.

[0081] Condition 6: if the flag of the intra string copy prediction mode of the encoding unit where the to-be-filtered boundary is located is 1, the boundary does not require filtering.

[0082] DBF processing is generally performed in units of 8x8 for deblocking vertical filtering and deblocking horizontal filtering. DBF processing can filter up to 3 pixels on both sides of the adjoining boundary of the current block (e.g., picture region to be filtered), and can use up to 4 pixels on both sides of the adjoining boundary for filtering. Therefore, vertical/horizontal filtering for different blocks does not affect each other and can be performed in parallel. As shown in FIG. 2, for an 8x8 current block, vertical filtering is first performed on the left 3 columns of pixels in the current block and the right 3 columns of pixels in the left block, and then horizontal filtering is performed on the upper 3 rows of pixels in the current block and the lower 3 rows of pixels in the upper block.

[0083] (1) The derivation process of the boundary strength (BS) value is as follows.

[0084] As shown in FIG. 3, a schematic diagram of the vertical adjoining boundary (vertical boundary) and horizontal adjoining boundary (horizontal boundary) of the current block is presented. The 8 pixel samples on both sides of the vertical/horizontal boundary are denoted as p_0, p_1, p_2, p_3 , and q_0, q_1, q_2, q_3 . It should be noted that when the boundary strength (BS) value is calculated, the boundary strength (BS) value is calculated for the pixels of the current block, that is, the boundary is a boundary of the pixels of the current block, neither the boundary of the current luma block nor the boundary of the current chroma block.

[0085] Method 1 for determining the boundary strength is described.

[0086] If all of the following conditions a), b) and c) are met, the boundary strength (BS) value is equal to 0.

[0087] a) The quantization coefficients of the transform blocks of the encoding units where p_0 and q_0 are located are both 0. Where if p_0 (or q_0) is a luma sample and the encoding unit where p_0 (or q_0) is located only contains luma samples, the encoding unit where p_0 (or q_0) is located refers to the luma encoding unit that contains p_0 (or q_0); and if p_0 (or q_0) is a chroma sample and the encoding unit where p_0 (or q_0) is located only contains chroma samples, the encoding unit where p_0 (or q_0) is located refers to the encoding unit that contains the corresponding luma sample of p_0 (or q_0); otherwise (i.e., if the encoding unit where p_0 or q_0 is located contains both luma samples and chrominance samples), the encoding unit where p_0 (or q_0) is located refers to the encoding unit that contains p_0 (or q_0).

[0088] b) The prediction modes for the encoding units where p_0 and q_0 are located are not intra prediction mode.

[0089] c) B_P and B_Q are respectively the 4x4 luma coding blocks where p_0 and q_0 are located. The motion information of B_P and B_Q simultaneously satisfies both conditions 1 and 2 or simultaneously satisfies conditions 3, 4, and 5.

[0090] 1) The L0 reference indices of the spatial motion information storage units respectively corresponding to B_P and B_Q are both equal to -1, or the reference frames corresponding to the L0 reference indices of the spatial motion information storage units respectively corresponding to B_P and B_Q are the same frame and the difference between all components of the L0 motion vector of the spatial motion information storage unit is less than one integer pixel.

[0091] 2) The L1 reference indices of the spatial motion information storage units respectively corresponding to B_P and B_Q are both equal to -1, or the reference frames corresponding to the L1 reference indices of the spatial motion information storage units respectively corresponding to B_P and B_Q are the same frame and the difference between all components of the L1 motion vector of the spatial motion information storage unit is less than one integer pixel.

[0092] 3) One of the following conditions is met.

[0093] The L0 reference index of the spatial motion information storage unit corresponding to B_P is equal to -1, and the L1 reference index of the spatial motion information storage unit corresponding to B_Q is equal to -1.

[0094] The reference frame corresponding to the L0 reference index of the spatial motion information storage unit corresponding to B_P and the reference frame corresponding to the L1 reference index of the spatial motion information storage unit corresponding to B_Q are the same frame, and the difference between all components of the L0 motion vector of the spatial motion information storage unit corresponding to B_P and the L1 motion vector of the spatial motion information storage unit corresponding to B_Q is less than one integer pixel.

[0095] 4) One of the following conditions is met.

[0096] Condition 1: the L0 reference index of the spatial motion information storage unit corresponding to B_Q is equal to -1, and the L1 reference index of the spatial motion information storage unit corresponding to B_P is equal to -1.

[0097] Condition 2: the reference frame corresponding to the L0 reference index of the spatial motion information storage unit corresponding to B_Q and the reference frame corresponding to the L1 reference index of the spatial motion information storage unit corresponding to B_P are the same frame, and the difference between all components of the L0 motion vector of the spatial motion information storage unit corresponding to B_Q and the L1 motion vector of the spatial motion information storage unit corresponding to B_P is less than one integer pixel.

[0098] 5) The reference frame corresponding to the L0 reference index of the spatial motion information storage unit corresponding to B_P and the reference frame corresponding to the L0 reference index of the spatial motion information storage unit corresponding to B_Q are not the same frame; and the reference frame corresponding to the L1 reference index of the spatial motion information storage unit corresponding to B_P and the reference frame corresponding to the L1 reference index of the spatial motion information storage unit corresponding to B_Q are not the same frame.

[0099] Otherwise, according to Method 2 for determining the boundary strength, the boundary strength (BS) value is calculated.

[0100] Method 2 for determining the boundary strength is described.

[0101] Step 1: the average quantization parameter QP_{av} of the encoding unit where p_0 and q_0 are located is calculated. If it is a luma sample, the quantization parameters of the luma coding block should be used; and if it is a chroma sample, the quantization parameters of the chroma coding block should be used. Let the quantization parameter of the encoding unit where p_0 is

located be QP_p , and the quantization parameter of the encoding unit where q_0 is located be QP_q , then the average quantization parameter is

$$QP_{av} = (QP_p + QP_q + 1) >> 1$$

[0102] Step 2: the indices indexA and indexB are calculated.

$$\begin{aligned} \text{IndexA} &= \text{Clip3}(0, 63, QP_{av} - 8 * (\text{BitDepth} - 8) + \text{AlphaOffset}) \\ \text{IndexB} &= \text{Clip3}(0, 63, QP_{av} - 8 * (\text{BitDepth} - 8) + \text{BetaOffset}) \end{aligned}$$

[0103] Step 3: the table is searched based on indexA and indexB to obtain the values of α' and β' , and then the values of α and β are obtained based on BitDepth.

$$\begin{aligned} \alpha &= \alpha' << (\text{BitDepth} - 8) \\ \beta &= \beta' << (\text{BitDepth} - 8) \end{aligned}$$

[0104] For example, ">>" is a right shift operation used to replace division, where ">>5" is equivalent to dividing by 2^5 (i.e., 32). In addition, in the embodiments of the present disclosure, multiplication (i.e., "*") can be replaced by left shift in practical implementation. For example, multiplying a by 4 can be replaced by a left shift of 2 bits, i.e., by $a << 2$. Multiplying a by 10 can be replaced by $(a << 3) + (a << 1)$.

[0105] For example, "<<" is a left shift operation used to replace multiplication, where " $a << 2$ " is equivalent to multiplying by 2^2 (i.e., 4).

[0106] For example, considering that when division operations are implemented through shifting, the result of the operation is usually rounded directly.

[0107] When the result of the operation is a non integer between N and $N+1$, the result is taken as N . However, considering that the accuracy of taking the result as $N+1$ will be higher when the decimal part is greater than 0.5, in order to improve the accuracy of the determined pixel value, when calculating, the numerator of the weighted sum can be added with half of the denominator (i.e. the divisor) to achieve rounding effect.

[0108] Step 4: if DeblockingFilterType is 1 and $\text{Abs}(p_0 - q_0)$ is greater than or equal to $4x\alpha$, Bs is equal to 0; and otherwise, the Bs value is calculated using the following method.

[0109] 1) the values of fL and fR are set to 0 and fS is calculated.

$$\begin{aligned} &\text{if } (\text{Abs}(p_0 - p_1) < \beta) \{ \\ &\quad \text{fL} += 2 \\ &\} \end{aligned}$$

```

if (Abs(p0-p2) < β) {
  fL++
}
if (Abs(q0-q1) < β) {
  fR += 2
}
if (Abs(q0-q2) < β) {
  fR ++
}
fS = fL + fR

```

[0110] 2) According to fS, the Bs value is determined, and there are the following situations 1-7.

[0111] Situation 1: when fS is equal to 6, if Abs (p₀-p₁) is less than or equal to β/4 and Abs (q₀-q₁) is less than or equal to β/4 and Abs (p₀-q₀) is less than α, then the Bs value is equal to 4; and otherwise, the Bs value is equal to 3.

[0112] Situation 2: when fS is equal to 6 and DeblockingFilterType is equal to 1, if Abs (p₀-p₁) is less than or equal to β/4 and Abs (q₀-q₁) is less than or equal to β/4 and Abs (p₀-p₃) is less than or equal to β/2 and Abs (q₀-q₃) is less than or equal to β/2 and Abs (p₀-q₀) is less than α, then the Bs value is equal to 4; and otherwise, the Bs value is equal to 3.

[0113] Situation 3: when fS is equal to 5, if p₀ is equal to p₁ and q₀ is equal to q₁, then the Bs value is equal to 3; and otherwise, the Bs value is equal to 2.

[0114] Situation 4: when fS is equal to 5 and DeblockingFilterType is equal to 1, if p₀ is equal to p₁ and q₀ is equal to q₁ and Abs (p₂-q₂) is less than α, then the Bs value is equal to 3; and otherwise, the Bs value is equal to 2.

[0115] Situation 5: when fS is equal to 4, if fL is equal to 2, then the Bs value is equal to 2; and otherwise, the Bs value is equal to 1.

[0116] Situation 6: when fS is equal to 3, if Abs (p₁ - q₁) is less than β, then the Bs value is equal to 1; and otherwise, the Bs value is equal to 0.

[0117] situation 7: when fS is other values, the Bs value is equal to 0.

[0118] 3) if the Bs value obtained in step 2) is not equal to 0 and the filtered boundary is the boundary of the chroma encoding block, then subtract 1 from the Bs value.

[0119] (2) Determine the deblocking filter adjustment parameters.

[0120] Determine the deblocking filter adjustment parameters DbrThreshold, DbrOffset0, DbrOffset1, DbrAltOffset0, and DbrAltOffset1.

[0121] If the current boundary is a vertical boundary and the PictureDbrVEnableFlag is 1, or if the current boundary is a horizontal boundary and the PictureDbrHEnableFlag is 1, then the value of the PictureDbrEnableFlag is 1; and otherwise, the value of PictureDbrEnableFlag is 0.

[0122] If the current boundary is a vertical boundary and the PictureAltDbrVEnableFlag is 1, or if the current boundary is a horizontal boundary and the PictureAltDbrHEnableFlag is 1, then the value of the PictureAltDbrEnableFlag is 1; and otherwise, the value of PictureAltDbrEnableFlag is 0.

[0123] For vertical boundaries:

DbrThreshold = DbrVThreshold DbrOffset0= DbrVOffset0 DbrOffset1= DbrVOffset1 DbrAltOffset0= DbrVAltOffset0 DbrAltOffset1= DbrVAltOffset1
--

[0124] For horizontal boundaries:

DbrThreshold = DbrHThreshold DbrOffset0= DbrHOffset0 DbrOffset1= DbrHOffset1 DbrAltOffset0= DbrHAltOffset0 DbrAltOffset1= DbrHAltOffset1
--

[0125] (3) Boundary filtering process when luma component Bs is equal to 4.

[0126] When the value of the boundary strength Bs is 4, the calculation process of filtering for p_0 , p_1 , p_2 and q_0 , q_1 , q_2 is as follows (P_0 , P_1 , P_2 and Q_0 , Q_1 , Q_2 are the filtered values):

$P_0 = (p_2 * 3 + p_1 * 8 + p_0 * 10 + q_0 * 8 + q_1 * 3 + 16) >> 5$ $P_1 = (p_2 * 4 + p_1 * 5 + p_0 * 4 + q_0 * 3 + 8) >> 4$ $P_2 = (p_3 * 2 + p_2 * 2 + p_1 * 2 + p_0 * 1 + q_0 * 1 + 4) >> 3$ $Q_0 = (p_1 * 3 + p_0 * 8 + q_0 * 10 + q_1 * 8 + q_2 * 3 + 16) >> 5$ $Q_1 = (p_0 * 3 + q_0 * 4 + q_1 * 5 + q_2 * 4 + 8) >> 4$ $Q_2 = (p_0 * 1 + q_0 * 1 + q_1 * 2 + q_2 * 2 + q_3 * 2 + 4) >> 3$

[0127] If the value of PictureDbrEnableFlag is 1, the values of P_0 , P_1 , P_2 , Q_0 , Q_1 , and Q_2 are adjusted according to the "(9) Deblocking Filter Adjustment Process" below.

[0128] (4) Boundary filtering process when luma component Bs is equal to 3.

[0129] When the value of the boundary strength Bs is 3, the calculation process of filtering for p_0 , p_1 and q_0 , q_1 is as follows (P_0 , P_1 and Q_0 , Q_1 are the filtered values):

$$\begin{aligned} P_0 &= (p_2 + (p_1 \ll 2) + (p_0 \ll 2) + (p_0 \ll 1) + (q_0 \ll 2) + q_1 + 8) \gg 4 \\ P_1 &= ((p_2 \ll 1) + p_2 + (p_1 \ll 3) + (p_0 \ll 2) + q_0 + 8) \gg 4 \\ Q_0 &= (p_1 + (p_0 \ll 2) + (q_0 \ll 2) + (q_0 \ll 1) + (q_1 \ll 2) + q_2 + 8) \gg 4 \\ Q_1 &= ((q_2 \ll 1) + q_2 + (q_1 \ll 3) + (q_0 \ll 2) + p_0 + 8) \gg 4 \end{aligned}$$

[0130] If the value of PictureDbrEnableFlag is 1, the values of P_0 , P_1 , Q_0 and Q_1 are adjusted according to the "(9) Deblocking Filter Adjustment Process" below.

[0131] (5) Boundary filtering process when luma component Bs is equal to 2.

[0132] When the value of the boundary strength Bs is 2, the calculation process of filtering for p_0 and q_0 is as follows (P_0 and Q_0 are the filtered values):

$$\begin{aligned} P_0 &= ((p_1 \ll 1) + p_1 + (p_0 \ll 3) + (p_0 \ll 1) + (q_0 \ll 1) + q_0 + 8) \gg 4 \\ Q_0 &= ((p_0 \ll 1) + p_0 + (q_0 \ll 3) + (q_0 \ll 1) + (q_1 \ll 1) + q_1 + 8) \gg 4 \end{aligned}$$

[0133] If the value of PictureDbrEnableFlag is 1, the values of P_0 and Q_0 are adjusted according to the "(9) Deblocking Filter Adjustment Process" below.

[0134] (6) Boundary filtering process when luma component Bs is equal to 1.

[0135] When the value of the boundary strength Bs is 1, the calculation process of filtering for p_0 and q_0 is as follows (P_0 and Q_0 are the filtered values):

$$\begin{aligned} P_0 &= ((p_0 \ll 1) + p_0 + q_0 + 2) \gg 2 \\ Q_0 &= ((q_0 \ll 1) + q_0 + p_0 + 2) \gg 2 \end{aligned}$$

[0136] If the value of PictureDbrEnableFlag is 1, the values of P_0 and Q_0 are adjusted according to the "(9) Deblocking Filter Adjustment Process" below.

[0137] (7) Boundary filtering process when luma component Bs is equal to 0.

[0138] When the value of the boundary strength Bs is 0 and the value of the PictureAltDbrEnableFlag is 1, the calculation process of filtering adjusting for p_0 and q_0 is as follows (P_0 and Q_0 are the values after filtering adjusting):

$$\begin{aligned} DP_0 &= (q_0 - p_0 + 2) \gg 2 \\ \text{if } (DP_0 < -DbrThreshold) \{ \end{aligned}$$

```

    P0 = Clip1(p0 + DbrAltOffset0)
}
else if (DP0 > DbrThreshold) {
    P0 = Clip1(p0 + DbrAltOffset1)
}
DQ0 = (p0 - q0 + 2) >> 2
if (DQ0 < -DbrThreshold) {
    Q0 = Clip1(q0 + DbrAltOffset0)
}
else if (DQ0 > DbrThreshold) {
    Q0 = Clip1(q0 + DbrAltOffset1)
}

```

[0139] (8) Boundary filtering process when chroma component Bs is greater than 0

[0140] When the value of the boundary strength Bs is greater than 0, the calculation process of filtering for p₀ and q₀ is as follows (P₀ and Q₀ are the filtered values):

```

P0 = ((p1 << 1) + p1 + (p0 << 3) + (p0 << 1) + (q0 << 1) + q0 + 8) >> 4
Q0 = ((p0 << 1) + p0 + (q0 << 3) + (q0 << 1) + (q1 << 1) + q1 + 8) >> 4

```

[0141] When the value of the boundary strength Bs is equal to 3, the calculation process of filtering for p₁ and q₁ is as follows (P₁ and Q₁ are the filtered values):

```

P1 = ((p2 << 1) + p2 + (p1 << 3) + (p0 << 1) + p0 + (q0 << 1) + 8) >> 4
Q1 = ((q2 << 1) + q2 + (q1 << 3) + (q0 << 1) + q0 + (p0 << 1) + 8) >> 4

```

[0142] (9) Adjustment process of deblocking filter

[0143] Adjust the values of P_i and Q_i (i can be 0, 1, or 2):

```

if (pi > Pi + DbrThreshold) {
    Pi = Clip1((Pi + pi + 1) >> 1 + DbrOffset0)
}
else if (pi < Pi - DbrThreshold) {
    Pi = Clip1((Pi + pi + 1) >> 1 + DbrOffset1)
}
if (qi > Qi + DbrThreshold) {
    Qi = Clip1((Qi + qi + 1) >> 1 + DbrOffset0)
}

```

```

    }
    else if ( $q_i < Q_i - \text{DbrThreshold}$ ) {
         $Q_i = \text{Clip1}((Q_i + q_i + 1) \gg 1 + \text{DbrOffset1})$ 
    }

```

[0144] Clip1 (x) represents limiting x between $[0, 2^{(\text{bit_depth}) - 1}]$ (including 0 and $2^{(\text{bit_depth}) - 1}$). The bit_depth represents the bit depth of a picture, usually 8, 10, or 12, etc.

[0145] Fourth, SAO

[0146] SAO is configured to eliminate ringing effects. The ringing effect is caused by the quantization distortion of high-frequency AC coefficients, which generates ripples around the edges after decoding. The larger the size of the transform block, the more obvious the ringing effect. The basic principle of SAO is to compensate for the peak pixels in the reconstructed curve by adding negative values, and to supplement the valley pixels by adding positive values. SAO uses a Collect Transfer Unit (CTU) as a basic unit and includes two types of compensation: Edge Offset (EO) and Band Offset (BO). In addition, the parameter fusion technology is introduced.

[0147] Sample adaptive offset includes basic sample adaptive offset (SAO mentioned above), enhanced sample adaptive offset (the above ESAO), and cross component sample adaptive offset (the above CCSAO, only for chroma).

[0148] The basic process of SAO is described.

[0149] If SAO is enabled for the current component of the current block, the basic sample offset compensation unit is first determined, then the basic sample offset compensation information corresponding to the current basic sample offset compensation unit is determined, and finally the operation is performed on each component of each sample in the current basic sample offset compensation unit to obtain the offset sample value; and otherwise, the filtered value of a corresponding component of the sample is directly used as the offset value of the component of the sample.

[0150] The basic process of ESAO (ESAO and SAO are mutually exclusive, i.e., if SAO is enabled, ESAO is not enabled) is described.

[0151] If ESAO is enabled for the current component of the current block, the enhanced sample offset compensation unit is first determined, and then the operation is performed on each component of each sample in the enhanced sample offset compensation unit to obtain the offset

sample value; and otherwise, the filtered value of a corresponding component of the sample is directly used as the offset value of the component of the sample.

[0152] The basic process of CCSAO is described.

[0153] If CCSAO is enabled for the current component of the current block, the cross component sample offset compensation unit is first determined, and then the operation is performed on each component sample in the current cross component sample offset compensation unit to obtain the cross component offset sample value; and otherwise, the offset value of the corresponding component of the sample is directly used as the value of the sample component after cross component offset.

[0154] For chroma pixels that require CCSAO, the classification of the current chroma pixel is determined based on the luma value (Y5) after the deblocking filter (adjustment) of the corresponding position and the current chroma pixel (the pixel after ESAO). Based on classification, the compensation value for the current chroma pixel is determined, and the compensation value is added to the current chroma pixel, to obtain the CCSAO filtered chroma pixel value.

[0155] Fifth, ALF is described.

[0156] ALF is an optimal filter (i.e., Wiener filter) in the mean square sense that is obtained by calculating based on the original signal and distortion signal. The Adaptive Loop Filter (ALF) can either enable the Enhanced Adaptive Loop Filter (EALF) or not enable the Enhanced Adaptive Loop Filter.

[0157] The basic process of ALF is as follows:

[0158] If ALF is not enabled for the current component of the picture, the offset value of the sample component will be directly used as the corresponding reconstructed value of the sample component; and otherwise, Adaptive Loop Filter is performed on the corresponding offset sample component.

[0159] The unit of Adaptive Loop Filter is an adaptive loop filter unit derived from the maximum coding unit and processed sequentially according to the raster scanning order. Firstly, the Adaptive Loop Filter coefficient of each component is decoded, then the Adaptive Loop Filter unit is obtained, and the index of the Adaptive Loop Filter coefficient for the luma component of the current Adaptive Loop Filter unit is determined. Finally, Adaptive Loop Filter is performed on the luma and chroma components of the Adaptive Loop Filter unit to obtain the reconstructed sample.

[0160] If EALF is enabled for the current picture, there are 15 filter coefficients in each group of ALF for the current picture, as shown in (a) of FIG. 4, the maximum number of adaptive

correction filters for the luma component of the current picture is 64; and otherwise, there are 9 filter coefficients in each group of ALF for the current picture, as shown in (b) of FIG. 4, the maximum number of adaptive correction filters for the luma component of the current picture is 16.

[0161] Whether it is DBF, SAO, or ALF, classification is performed based on the current pixel value of the to-be-filtered adjoining boundary, or the relationship (such as the permission level) between the pixel value of the current block and the pixel values of adjacent blocks surrounding the current block, and then different filtering operations are performed based on different categories.

[0162] As mentioned earlier, in video encoding and decoding, the purpose of filtering processing is to achieve smooth noise reduction and detail removal within a video picture block, and to preserve picture edges to the greatest extent. Usually, electronic devices perform filtering processing on the boundaries of multiple regions in a picture to preserve the picture edges.

[0163] In order to protect user data security, a general picture can be set with permission regions (such as a high permission level region, and/or a low permission level region) and a non permission region (such as a zero permission level region). The permission regions can include multiple permission regions of different levels. The permission region refers to the region where only users with relevant permissions can correctly decode the picture content of the region. For example, only users with high permissions can correctly view picture regions of any permission level, and users with low permissions cannot view picture regions of higher permission levels, and can only view picture regions of low and zero permission levels.

[0164] Therefore, due to the different viewing permissions of different users for different regions in the picture, in the case where low permission users do not have viewing permissions for high permission regions in the picture, electronic devices will not perform filtering, during video decoding, on the adjoining boundary of picture regions with different permission levels, which results in poor picture quality.

[0165] For this problem, the embodiments of the present disclosure provide picture processing methods and apparatuses, electronic devices, and storage media. In the methods, by an electronic device, the to-be-processed picture is divided into multiple picture subregions, and the permission level of each picture subregion is obtained. By determining whether the permission level of each picture subregion is the same as the permission levels of at least two adjacent picture subregions in the multiple picture subregions are different, it indicates that there is a situation where after filtering process in the decoding process, the two picture subregions may

still be unfiltered, and the quality of the to-be-processed picture is not high. Therefore, based on decoding information such as the division of permission regions, the electronic device can perform filtering on the adjoining boundary between at least two picture subregions again, which can improve the picture quality of regions with different permissions in the to-be-processed picture.

[0166] The picture processing methods provided in the embodiments of the present disclosure can be applied to the picture processing apparatus 11 shown in FIG. 5. The picture processing apparatus includes a picture dividing unit 101, a picture filtering unit 102, and a picture decoding unit 103.

[0167] The picture dividing unit 101 is configured to divide the to-be-processed picture into multiple picture subregions.

[0168] The picture decoding unit 103 is configured to obtain the permission level of each picture subregion.

[0169] The picture filtering unit 102 is configured to, if the permission levels of at least two adjacent picture subregions in the multiple picture subregions are different, perform filtering on the adjoining boundary between the at least two adjacent picture subregions.

[0170] The picture decoding unit 103 is further configured to receive the decoding bitstream of the to-be-processed picture, and decode based on the decoding bitstream to obtain the decoded to-be-processed picture.

[0171] In some embodiments, the picture filtering unit 102 is further configured to determine a target to-be-filtered region on a vertical side of a first subregion in the at least two adjacent picture subregions, where the vertical side refers to a left side or a right side of the adjoining boundary of the first subregion, and the target to-be-filtered region is any one of: a right to-be-filtered region on the adjoining boundary, a left to-be-filtered region on the adjoining boundary, or a right to-be-filtered region and a left to-be-filtered region on the adjoining boundary; and the picture filtering unit 102 is further configured to perform deblocking vertical filtering on the target to-be-filtered region.

[0172] In some embodiments, the picture filtering unit 102 is further configured to determine a target to-be-filtered region on a horizontal side of a first subregion in the at least two adjacent picture subregions, where the horizontal side refers to an upper side or a lower side of the adjoining boundary of the first subregion, and the target to-be-filtered region is any one of: a lower to-be-filtered region on the adjoining boundary, an upper to-be-filtered region on the adjoining boundary, or a lower to-be-filtered region and an upper to-be-filtered region on the

adjoining boundary; and the picture filtering unit 102 is further configured to perform deblocking horizontal filtering on the target to-be-filtered region.

[0173] In some embodiments, the picture filtering unit 102 is further configured to determine a target to-be-filtered region on a surrounding side of a first subregion in the at least two adjacent picture subregions according to a set filtering sequence, where the surrounding side sequentially includes: a left side, an upper side, a right side, and a lower side on the adjoining boundary between the first subregion and other subregions, the permission level of the first subregion is different from the permission levels of the other subregions, and the target to-be-filtered region is any one of: a first side region on the adjoining boundary within the first subregion, a second side region on the adjoining boundary outside the first subregion, or the first side region and the second side region; and the picture filtering unit 102 is further configured to perform sample adaptive offset on the target to-be-filtered region according to the set filtering sequence.

[0174] In some embodiments, the picture filtering unit 102 is further configured to determine a target to-be-filtered region on a surrounding side of a first subregion in the at least two adjacent picture subregions according to a set filtering sequence, where the surrounding side sequentially includes: a left side, an upper side, a right side, and a lower side on the adjoining boundary between the first subregion and other subregions, the permission level of the first subregion is different from the permission levels of the other subregions, and the target to-be-filtered region is any one of: a first side region on the adjoining boundary within the first subregion, a second side region on the adjoining boundary outside the first subregion, or the first side region and the second side region; and the picture filtering unit 102 is further configured to perform adaptive loop filter on the target to-be-filtered region according to the set filtering sequence.

[0175] In some embodiments, the picture filtering unit 102 is further configured to determine whether neural network-based filtering is required for the first subregion of at least two picture subregions based on the decoding information of the to-be-processed picture; when neural network-based filtering is required for the first subregion, input the reconstructed pixel values and the permission level of the first subregion into the neural network; and obtain the outputted filtered reconstructed pixels of the first subregion.

[0176] In some embodiments, the picture filtering unit 102 is further configured to input the permission level of at least one picture subregion adjacent to the first subregion into the neural network.

[0177] In some embodiments, the picture filtering unit 102 is further configured to perform a filtering process for a luma channel component on the adjoining boundary between the at least two picture subregions; or perform a filtering process for a chroma channel component on the adjoining boundary between the at least two picture subregions; or perform a filtering process for a luma channel component and a chroma channel component on the adjoining boundary between the at least two picture subregions.

[0178] FIG. 6 is a flowchart of a picture processing method provided in embodiments the present disclosure. For example, the picture processing method provided in the present disclosure can be applied to the picture processing apparatus shown in FIG. 5. As shown in FIG. 6, the picture processing method provided in the present disclosure may specifically include the following steps S101 to S102.

[0179] In S101, the to-be-processed picture is divided into multiple picture subregions and the permission level of each picture subregion is obtained.

[0180] Where permission levels are divided into at least two different permission levels.

[0181] In some embodiments, pictures can be configured with permission regions and non permission regions (zero permission level regions). The picture subregion of the non permission region (zero permission level region) can be viewed by any user. The permission regions can include multiple permission regions of different levels. For example, the permission regions can include a first permission level region (high permission level region), a second permission level region (medium permission level region), and a third permission level region (low permission level region). The permission level of the first permission level region is higher than the permission level of the second permission level region, and the permission level of the second permission level region is higher than the permission level of the third permission level region. Each permission level can further be divided into multiple sub permission levels, which is not specifically limited in this embodiment.

[0182] In some embodiments, the permission level of the target region in the picture is configured based on the category of the target region. For example, the face of a person in the picture is a high permission level region, the upper body is a low permission level region, and the lower body is a zero permission level region. The license plate part of a vehicle in a picture is a high permission level region, the window part of the vehicle is a low permission level region, and the body part of the vehicle is a zero permission level region.

[0183] In some embodiments, the permission levels of picture subregions in the to-be-processed picture can be divided into high permission level regions, low permission level regions, and zero permission level regions. For example, as shown in FIG. 7, multiple picture

subregions can include low permission level regions, high permission level regions, and zero permission level regions.

[0184] It can be understood that different users have different viewing permissions for different regions in the picture. High permission users can view picture subregions of any permission level (such as high permission level, low permission level region, or zero permission level region), low permission users can view picture subregions of low permission level and zero permission level, and non permission users can only view picture subregions of zero permission level.

[0185] In some embodiments, before the to-be-processed picture is divided into multiple picture subregions, the picture decoding unit 103 receives the decoding bitstream of the to-be-processed picture, and decode based on the decoding bitstream to obtain the decoded to-be-processed picture.

[0186] In other embodiments, when the to-be-processed picture includes the target picture region, and the target picture region is used as the filtering processing object, the picture decoding unit 103 receives the decoding bitstream of the to-be-processed picture and decodes the target picture region based on the decoding bitstream. After the target picture region is decoded, there is no need to wait for the entire picture to be completely decoded, and the target picture region can be directly divided into multiple picture subregions, and the permission level of each picture subregion can be obtained.

[0187] In other embodiments, when the to-be-processed picture includes a target picture region, and the target picture region is used as the filtering processing object, the target picture region can be taken as a center and expanded to the surrounding side (the left side, the right side, the upper side, or the lower side) to obtain a detection region with a width of CW and a height of CH. The detection region can be divided to obtain multiple picture sub block regions with $N \times N$ (N is preferably 2, 4, or 8) pixel units, where each picture sub block region has the same permission level. Where the extended pixel units of the detection region in the four directions can be different or completely the same.

[0188] For example, as shown in (a) of FIG. 8, the detection region can be expanded around the target picture region to the left and up by T (preferably 4, 8, or 16) unit pixels.

[0189] For example, as shown in (b) of FIG. 8, an additional region can also be added on the upper right of the detection region in (a) of FIG. 8 to obtain the entire detection region.

[0190] In S102, when the permission levels of at least two adjacent picture subregions in multiple picture subregions are different, filtering process is performed on the adjoining boundary (to-be-filtered boundary) between at least two picture subregions.

[0191] In some embodiments, when the to-be-processed picture includes the target picture region of the filtering processing object, after the target picture region is decoded by the picture decoding unit 103, the target picture region can be directly divided into multiple picture subregions. When the permission levels of at least two adjacent picture subregions in the multiple picture subregions are different, filtering process can be directly performed on the adjoining boundary between the two picture subregions without waiting for the entire picture to be decoded.

[0192] In some embodiments, the filtering process includes at least one of: deblocking vertical filtering, deblocking horizontal filtering, sample adaptive offset, adaptive loop filter, or neural network filtering.

[0193] In some embodiments, at least two adjacent picture subregions may be two picture subregions that are contiguous or may be two picture subregions that are not contiguous but are in close proximity to each other.

[0194] In some embodiments, a filtering process for a luma channel component can be performed on the adjoining boundary between the at least two picture subregions; or a filtering process for a chroma channel component can be performed on the adjoining boundary between the at least two picture subregions; or a filtering process for a luma channel component and a chroma channel component can be performed on the adjoining boundary between the at least two picture subregions.

[0195] In some embodiments, the picture processing apparatus can obtain picture region information of any permission level in the to-be-processed picture, and picture region information of non permission regions; Alternatively, the picture processing apparatus can obtain picture region information of the second permission level in the to-be-processed picture, and picture region information of non permission regions.

[0196] It can be understood that the to-be-processed picture is divided into multiple picture subregions, and the permission level of each picture subregion is obtained. Then, by determining whether the permission level of each picture subregion, if it is determined that the permission levels of at least two adjacent picture subregions in the multiple picture subregions are different, it indicates that there is a situation where after filtering process in the decoding process, the two picture subregions may still be unfiltered, and the quality of the to-be-processed picture is not high. Therefore, based on decoding information such as the division of permission regions, the filtering process can be performed on the adjoining boundary between at least two picture subregions again, which can improve the picture quality of regions with different permissions in the to-be-processed picture.

[0197] In some embodiments, if there is a vertical adjoining boundary between at least two picture subregions (i.e., the positional relationship between the two picture subregions is horizontally adjacent), the filtering process includes deblocking vertical filtering, and the DBF condition is met, the adjoining boundary between at least two picture subregions can be subjected to deblocking vertical filtering. Referring to FIG. 9, the decoding method provided in this embodiment includes the following steps Sa1 to Sa2.

[0198] In Sa1, the target to-be-filtered region on the vertical side of the first subregion in at least two picture subregions is determined.

[0199] Where the vertical side refers to a left side or a right side of the adjoining boundary of the first subregion, and the target to-be-filtered region is any one of: a right to-be-filtered region on the adjoining boundary, a left to-be-filtered region on the adjoining boundary, or a right to-be-filtered region and a left to-be-filtered region on the adjoining boundary.

[0200] In Sa2, deblocking vertical filtering is performed on the target to-be-filtered region.

[0201] In some embodiments, at least two adjacent picture subregions further include a second subregion, where the second subregion is horizontally adjacent to the first subregion. Horizontal adjacency here refers to the second subregion being located on the left or right side of the first subregion in the horizontal direction. The permission levels of at least two adjacent picture subregions are different, including any of the following situations 1 to 4.

[0202] 1. The permission level of the first subregion is lower than the permission level of the second subregion.

[0203] 2. The permission level of the first subregion is higher than the permission level of the second subregion.

[0204] 3. The permission level of the first subregion is not equal to the permission level of the second subregion.

[0205] 4. The permission level of the first subregion is not equal to the permission level of the second subregion, and a filtering strength (BS value) of the adjoining boundary is greater than or equal to a preset filtering strength.

[0206] As shown in (a) of FIG. 10, in the case where the second subregion is located on the left of the first subregion, when the permission levels of at least two adjacent picture subregions are different, the target to-be-filtered region is subjected to deblocking vertical filtering, and a detailed description is given.

[0207] Embodiment 1: in the case where the permission level of the first subregion is lower than the permission level of the second subregion, the target to-be-filtered region is the to-be-filtered region on the right side of the adjoining boundary (within the first subregion), and the

target to-be-filtered region is subjected to deblocking vertical filtering. Where the to-be-filtered region on the right is a region of pixel points with a width of n columns to the right of the adjoining boundary, where n preferably is 1.

[0208] Embodiment 2: in the case where the permission level of the first subregion is higher than that of the second subregion, the target to-be-filtered region is the to-be-filtered region on the right side of the adjoining boundary (within the first subregion), and the target to-be-filtered region is subjected to deblocking vertical filtering. Where the to-be-filtered region on the right is a region of pixel points with a width of n columns to the right of the adjoining boundary, where n preferably is 1.

[0209] Embodiment 3: in the case where the permission level of the first subregion is not lower than that of the second subregion, the target to-be-filtered region is the to-be-filtered region on the right side of the adjoining boundary (within the first subregion), and the target to-be-filtered region is subjected to deblocking vertical filtering. Where the to-be-filtered region on the right is a region of pixel points with a width of n columns to the right of the adjoining boundary, where n preferably is 1.

[0210] Embodiment 4: in the case where the permission level of the first subregion is not equal to the permission level of the second subregion, and the BS value of the adjoining boundary is greater than or equal to 1, the target to-be-filtered region is the to-be-filtered region on the right side of the adjoining boundary (within the first subregion), and the target to-be-filtered region is subjected to deblocking vertical filtering. Where the to-be-filtered region on the right is a region of pixel points with a width of n columns to the right of the adjoining boundary, where n preferably is 1.

[0211] As shown in (b) of FIG. 10, in Embodiments 1 to 4, the target to-be-filtered region is the right to-be-filtered region (within the first subregion) of the adjoining boundary.

[0212] Embodiment 5: in the case where the permission level of the first subregion is lower than the permission level of the second subregion, the target to-be-filtered region (within the second subregion) is on the left side of the adjoining boundary, and the target to-be-filtered region is subjected to deblocking vertical filtering. Where the to-be-filtered region on the left is a region of pixel points with a width of n columns to the left of the adjoining boundary, where n preferably is 1.

[0213] Embodiment 6: in the case where the permission level of the first subregion is higher than the permission level of the second subregion, the target to-be-filtered region (within the second subregion) is on the left side of the adjoining boundary, and the target to-be-filtered region is subjected to deblocking vertical filtering. Where the to-be-filtered region on the left

is a region of pixel points with a width of n columns to the left of the adjoining boundary, where n preferably is 1.

[0214] Embodiment 7: in the case where the permission level of the first subregion is not equal to the permission level of the second subregion, the target to-be-filtered region (within the second subregion) is on the left side of the adjoining boundary, and the target to-be-filtered region is subjected to deblocking vertical filtering. Where the to-be-filtered region on the left is a region of pixel points with a width of n columns to the left of the adjoining boundary, where n preferably is 1.

[0215] Embodiment 8: in the case where the permission level of the first subregion is not equal to the permission level of the second subregion, and the BS value of the adjoining boundary is greater than or equal to 1, the target to-be-filtered region (within the second subregion) is on the left side of the adjoining boundary, and the target to-be-filtered region is subjected to deblocking vertical filtering. Where the to-be-filtered region on the left is a region of pixel points with a width of n columns to the left of the adjoining boundary, where n preferably is 1.

[0216] As shown in (c) of FIG. 10, in Embodiments 5 to 8, the target to be filtered region is the left to be filtered region (within the second subregion) of the adjoining boundary.

[0217] Embodiment 9: in the case where the permission level of the first subregion is lower than the permission level of the second subregion, the target to-be-filtered region is on the right and left regions (within the first and second subregions) of the adjoining boundary, and the target to-be-filtered region is subjected to deblocking vertical filtering. Where the target to-be-filtered region is a region of pixel points with a width of $2n$ pixels, located in the right n columns and left n sides of the adjoining boundary, and n is preferably 1.

[0218] Embodiment 10: in the case where the permission level of the first subregion is higher than the permission level of the second subregion, and the target to-be-filtered region is the right and left regions (within the first and second subregions) of the adjoining boundary, the target to-be-filtered region is subjected to deblocking vertical filtering. Where the target to-be-filtered region is a region of pixel points with a width of $2n$ pixels, located in the right n columns and left n sides of the adjoining boundary, and n is preferably 1.

[0219] Embodiment 11: in the case where the permission level of the first subregion is not equal to the permission level of the second subregion, and the target to-be-filtered region is the right and left regions (within the first and second subregions) of the adjoining boundary, the target to-be-filtered region is subjected to deblocking vertical filtering. Where the target to-be-filtered region is a region of pixel points with a width of $2n$ pixels, located in the right n columns and left n sides of the adjoining boundary, and n is preferably 1.

[0220] Embodiment 12: in the case where the permission level of the first subregion is not equal to the permission level of the second subregion, and the BS value of the adjoining boundary is greater than or equal to 1, and the target to-be-filtered region is the right and left regions (within the first and second subregions) of the adjoining boundary, the target to-be-filtered region is subjected to deblocking vertical filtering. Where the target to-be-filtered region is a region of pixel points with a width of $2n$ pixels, located in the right n columns and left n sides of the adjoining boundary, and n is preferably 1.

[0221] As shown in (d) of FIG. 10, in Embodiments 9 to 12, the target regions to be filtered are the right and left regions (within the first and second subregions) of the adjoining boundary.

[0222] It can be understood that if there is a vertical adjoining boundary between at least two picture subregions mentioned above, and the filtering process includes deblocking vertical filtering, the target to-be-filtered region on the vertical side of the first subregion in at least two picture subregions can be determined, and the target to-be-filtered region can be subjected to deblocking vertical filtering. Where based on the permission levels of the two picture subregions, the corresponding target to-be-filtered region can be selected for deblocking vertical filtering. In the case where the picture processing apparatus obtains information about the corresponding picture region based on the permission level of users, this method can meet the needs of users with any permission level to perform deblocking vertical filtering on the decoded picture, improve the picture quality of different permission regions in the to-be-processed picture, and thus enhance the user experience at different permission levels.

[0223] In some embodiments, if there is a horizontal adjoining boundary between at least two picture subregions (i.e., the positional relationship between the two picture subregions is horizontally adjacent), the filtering process includes deblocking horizontal filtering, and the DBF condition is met, the adjoining boundary between at least two picture subregions can be subjected to deblocking horizontal filtering. Referring to FIG. 11, the decoding method provided in this embodiment includes the following steps Sb1 to Sb2.

[0224] In Sb1, the target to-be-filtered region on the horizontal side of the first subregion in at least two picture subregions is determined.

[0225] Where the horizontal side refers to an upper side or a lower side of the adjoining boundary of the first subregion, and the target to-be-filtered region is any one of: an upper to-be-filtered region on the adjoining boundary, a lower to-be-filtered region on the adjoining boundary, or an upper to-be-filtered region and a lower to-be-filtered region on the adjoining boundary.

[0226] In Sb2, deblocking horizontal filtering is performed on the target to-be-filtered region.

[0227] In some embodiments, at least two adjacent picture subregions further include a third subregion, where the third subregion is vertically adjacent to the first subregion. Vertical adjacency here refers to the third subregion being located on the upper or lower side of the first subregion in the horizontal direction. The permission levels of at least two adjacent picture subregions are different, including any of the following situations 1 to 4.

[0228] 1. The permission level of the first subregion is lower than the permission level of the third subregion.

[0229] 2. The permission level of the first subregion is higher than the permission level of the third subregion.

[0230] 3. The permission level of the first subregion is not equal to the permission level of the third subregion.

[0231] 4. The permission level of the first subregion is not equal to the permission level of the third subregion, and a filtering strength (BS value) of the adjoining boundary is greater than or equal to a preset filtering strength.

[0232] As shown in (a) of FIG. 12, the third subregion is located on the upper side of the first subregion. Below, for the different permission levels of at least two adjacent picture subregions mentioned above, the target to-be-filtered region is subjected to deblocking horizontal filtering, and a detailed description is given.

[0233] Embodiment 13: in the case where the permission level of the first subregion is lower than the permission level of the third subregion, the target to-be-filtered region is on the lower side (within the first subregion) of the adjoining boundary, and the target to-be-filtered region is subjected to deblocking horizontal filtering. Where the to-be-filtered region on the lower side is a region of pixel points with a width of n rows from the adjoining boundary upwards, and n is preferably 1.

[0234] Embodiment 14: in the case where the permission level of the first subregion is higher than the permission level of the third subregion, the target to-be-filtered region (within the first subregion) is on the lower side of the adjoining boundary, and the target to-be-filtered region is subjected to deblocking horizontal filtering. Where the to-be-filtered region on the lower side is a region of pixel points with a width of n rows from the adjoining boundary upwards, and n is preferably 1.

[0235] Embodiment 15: in the case where the permission level of the first subregion is not lower than the permission level of the third subregion, the target to-be-filtered region is on the lower side (within the first subregion) of the adjoining boundary, and the target to-be-filtered region is subjected to deblocking horizontal filtering. Where the to-be-filtered region on the

lower side is a region of pixel points with a width of n rows from the adjoining boundary upwards, and n is preferably 1.

[0236] Embodiment 16: in the case where the permission level of the first subregion is not equal to the permission level of the third subregion, and the BS value of the adjoining boundary is greater than or equal to 1, the target to-be-filtered region is the to-be-filtered region on the lower side of the adjoining boundary (within the first subregion), and the target to-be-filtered region is subjected to deblocking horizontal filtering. Where the to-be-filtered region on the lower side is a region of pixel points with a width of n rows from the adjoining boundary upwards, and n is preferably 1.

[0237] As shown in (b) of FIG. 12, in Embodiments 13 to 16, the target to-be-filtered region is the lower to-be-filtered region (within the first subregion) of the adjoining boundary.

[0238] Embodiment 17: in the case where the permission level of the first subregion is lower than the permission level of the third subregion, the target to-be-filtered region is on the upper side (within the third subregion) of the adjoining boundary, and the target to-be-filtered region is subjected to deblocking horizontal filtering. Where the to-be-filtered region on the upper side is a region of pixel points with a width of n rows from the adjoining boundary downward, and n is preferably 1.

[0239] Embodiment 18: in the case where the permission level of the first subregion is higher than the permission level of the third subregion, the target to-be-filtered region (within the third subregion) is on the upper side of the adjoining boundary, and the target to-be-filtered region is subjected to deblocking horizontal filtering. Where the to-be-filtered region on the upper side is a region of pixel points with a width of n rows from the adjoining boundary downward, and n is preferably 1.

[0240] Embodiment 19: in the case where the permission level of the first subregion is not equal to the permission level of the third subregion, the target to-be-filtered region is on the upper side (within the third subregion) of the adjoining boundary, and the target to-be-filtered region is subjected to deblocking horizontal filtering. Where the to-be-filtered region on the upper side is a region of pixel points with a width of n rows from the adjoining boundary downward, and n is preferably 1.

[0241] Embodiment 20: in the case where the permission level of the first subregion is not equal to the permission level of the third subregion, and the BS value of the adjoining boundary is greater than or equal to 1, the target to-be-filtered region is the to-be-filtered region on the upper side (within the third subregion) of the adjoining boundary, and the target to-be-filtered region is subjected to deblocking horizontal filtering. Where the to-be-filtered region on the

upper side is a region of pixel points with a width of n rows from the adjoining boundary downward, and n is preferably 1.

[0242] As shown in (c) of FIG. 12, in Embodiments 17 to 20, the target to be filtered region is the upper to be filtered region (within the third subregion) of the adjoining boundary.

[0243] Embodiment 21: in the case where the permission level of the first subregion is lower than the permission level of the third subregion, and the target to-be-filtered region is in the lower and upper regions (within the first and third subregions) of the adjoining boundary, the target to-be-filtered region is subjected to deblocking horizontal filtering. Where the target to-be-filtered region is a region of pixel points with a width of $2n$ pixels on the upper n rows and lower n rows of the adjoining boundary, and n is preferably 1.

[0244] Embodiment 22: in the case where the permission level of the first subregion is higher than the permission level of the third subregion, and the target to-be-filtered region is in the lower and upper regions (within the first and third subregions) of the adjoining boundary, the target to-be-filtered region is subjected to deblocking horizontal filtering. Where the target to-be-filtered region is a region of pixel points with a width of $2n$ pixels on the upper n rows and lower n rows of the adjoining boundary, and n is preferably 1.

[0245] Embodiment 23: in the case where the permission level of the first subregion is not equal to the permission level of the third subregion, and the target to-be-filtered region is in the lower and upper regions (within the first and third subregions) of the adjoining boundary, the target to-be-filtered region is subjected to deblocking horizontal filtering. Where the target to-be-filtered region is a region of pixel points with a width of $2n$ pixels on the upper n rows and lower n rows of the adjoining boundary, and n is preferably 1.

[0246] Embodiment 24: in the case where the permission level of the first subregion is not equal to the permission level of the third subregion, the BS value of the adjoining boundary is greater than or equal to 1, and the target to-be-filtered region is the lower and upper to be filtered regions (within the first and third subregions) of the adjoining boundary, the target to-be-filtered region is subjected to deblocking horizontal filtering. Where the target to-be-filtered region is a region of pixel points with a width of $2n$ pixels on the upper n rows and lower n rows of the adjoining boundary, and n is preferably 1.

[0247] As shown in (d) of FIG. 12, in Embodiments 21 to 24, the target regions to be filtered are the upper and lower regions (within the first and third subregions) of the adjoining boundary.

[0248] It can be understood that if there is a horizontal adjoining boundary between at least two picture subregions mentioned above, and the filtering process includes deblocking

horizontal filtering, the target to-be-filtered region on the horizontal side of the first subregion in at least two picture subregions can be determined, and the target to-be-filtered region can be subjected to deblocking horizontal filtering. Where based on the permission levels of the two picture subregions, the corresponding target to-be-filtered region can be selected for deblocking horizontal filtering. In the case where the picture processing apparatus obtains information about the corresponding picture region based on the permission level of users, this method can meet the needs of users with any permission level to perform deblocking horizontal filtering on the decoded picture, improve the picture quality of different permission regions in the to-be-processed picture, and thus enhance the user experience at different permission levels.

[0249] In some embodiments, the filtering process includes sample adaptive offset; and if the permission levels of at least two adjacent picture subregions in multiple picture subregions are different, sample adaptive offset is performed on the boundary between at least two picture subregions. Referring to FIG. 13, the decoding method provided in this embodiment includes the following steps Sc1 to Sc2.

[0250] In Sc1, the target to-be-filtered region is determined from the surrounding side of the first subregion in at least two picture subregions according to the set filtering sequence.

[0251] Where the surrounding side sequentially includes: a left side, an upper side, a right side, and a lower side on the adjoining boundary between the first subregion and other subregions; and in the case where the permission level of the first subregion is different from the permission levels of the other subregions, the target to-be-filtered region is any one of: a first side region on the adjoining boundary within the first subregion, a second side region on the adjoining boundary outside the first subregion, or the first side region and the second side region.

[0252] In some embodiments, the permission levels of other subregions around the first subregion can be determined in sequence according to the set filtering sequence. In the case where the permission levels of other subregions around the first subregion are different from the permission level of the first subregion, the target to-be-filtered region can be determined. The target to-be-filtered region can include the following two situations 1 and 2.

[0253] Situation 1: if other subregions with different permission levels from the first subregion are located on the vertical side of the first subregion, the width of the first side region is n columns of pixels from the adjoining boundary towards the first subregion, and the width of the second side region is n columns of pixels from the adjoining boundary towards another subregion.

[0254] Situation 2: if other subregions with different permission levels from the first subregion are located on the horizontal side of the first subregion, the width of the first side region is n

rows of pixels from the adjoining boundary towards the first subregion, and the width of the second side region is n rows of pixels from the adjoining boundary towards another subregion.

[0255] The above two situations can exist simultaneously.

[0256] For example, as shown in FIG. 14, there are adjacent subregion A on the left, adjacent subregion B on the upper left, adjacent subregion C on the upper, and adjacent subregion D on the upper right. According to the filtering sequence from right to left and then from top to bottom, the permission levels of other subregions around the first subregion are sequentially determined. In the case where the permission levels of adjacent subregions A and C are different from the permission level of the first subregion, and the permission levels of adjacent subregions B and D are the same as the permission level of the first subregion, the target to-be-filtered region is determined as the column pixel region with a width of 1 pixel on the left side of the adjoining boundary with the adjacent subregion A and the column pixel region with a width of 1 pixel on the right side of the adjoining boundary with the adjacent subregion A, and the row pixel region with a width of 1 pixel on the upper side of the adjoining boundary with the adjacent subregion C and the row pixel region with a width of 1 pixel on the lower side of the adjoining boundary with the adjacent subregion C, as the shaded region shown in FIG. 14.

[0257] In other embodiments, the target to-be-filtered region is determined when the permission levels of other subregions around the first subregion are higher than the permission level of the first subregion. The target to-be-filtered region can include the following two situations 1 and 2.

[0258] Situation 1: if other subregions with permission levels higher than the first subregion are located on the vertical side of the first subregion, the width of the first side region is n columns of pixels from the adjoining boundary towards the first subregion, and the width of the second side region is n columns of pixels from the adjoining boundary towards another subregion.

[0259] Situation 2: if other subregions with permission levels higher than the first subregion are located on the horizontal side of the first subregion, the width of the first side region is n rows of pixels from the adjoining boundary towards the first subregion, and the width of the second side region is n rows of pixels from the adjoining boundary towards another subregion.

[0260] The above two situations can exist simultaneously.

[0261] In Sc2, sample adaptive offset is performed on the target to-be-filtered region according to the set filtering sequence.

[0262] In some embodiments, as shown in FIG. 14 above, sample adaptive offset is performed on the target to-be-filtered region (i.e., the shaded region in FIG. 14) in the filtering sequence from right to left and then from top to bottom.

[0263] In some embodiments, adjacent subregions around the first subregion that have not undergone filtering processing maintain their original pixel values. For example, as shown in FIG. 14 above, the pixel values of the adjoining boundary between the first subregion and the adjacent subregion B and the adjoining boundary between the first subregion and the adjacent subregion D remain unchanged.

[0264] It can be understood that if the permission levels of at least two adjacent picture subregions in multiple picture subregions are different, the target to-be-filtered region can be determined from the surrounding side of the first subregion in at least two picture subregions according to the set filtering sequence, and sample adaptive offset can be performed on the target to-be-filtered region according to the set filtering sequence. Where based on the permission levels of adjacent subregions around the first subregion, the corresponding target to-be-filtered region can be selected for sample adaptive offset. In the case where the picture processing apparatus obtains information about the corresponding picture region based on the permission level of users, this method can meet the needs of users with any permission level to perform sample adaptive offset on the decoded picture, improve the picture quality of different permission regions in the to-be-processed picture, and thus enhance the user experience at different permission levels.

[0265] In some embodiments, the filtering process includes adaptive loop filter; and if the permission levels of at least two adjacent picture subregions in multiple picture subregions are different, adaptive loop filter is performed on the adjoining boundary between at least two picture subregions. Referring to FIG. 15, the decoding method provided in this embodiment includes the following steps Sd1 to Sd2.

[0266] In Sd1, the target to-be-filtered region is determined from the surrounding side of the first subregion in at least two picture subregions according to the set filtering sequence.

[0267] Where the surrounding side sequentially includes: a left side, an upper side, a right side, and a lower side on the adjoining boundary between the first subregion and other subregions; and in the case where the permission level of the first subregion is different from the permission levels of the other subregions, the target to-be-filtered region is any one of: a first side region on the adjoining boundary within the first subregion, a second side region on the adjoining boundary outside the first subregion, or the first side region and the second side region.

[0268] In some embodiments, the permission levels of other subregions around the first subregion can be determined in sequence according to the set filtering sequence. In the case where the permission levels of other subregions around the first subregion are different from the permission level of the first subregion, the target to-be-filtered region can be determined. The target to-be-filtered region can include the following two situations 1 and 2.

[0269] Situation 1: if other subregions with different permission levels from the first subregion are located on the vertical side of the first subregion, the width of the first side region is n columns of pixels from the adjoining boundary towards the first subregion, and the width of the second side region is n columns of pixels from the adjoining boundary towards another subregion.

[0270] Situation 2: if other subregions with different permission levels from the first subregion are located on the horizontal side of the first subregion, the width of the first side region is n rows of pixels from the adjoining boundary towards the first subregion, and the width of the second side region is n rows of pixels from the adjoining boundary towards another subregion.

[0271] The above two situations can exist simultaneously.

[0272] For example, as shown in FIG. 16, there are adjacent subregion A on the left, adjacent subregion B on the upper left, adjacent subregion C on the upper, and adjacent subregion D on the upper right. According to the filtering sequence from right to left and then from top to bottom, the permission levels of other subregions around the first subregion are sequentially determined. In the case where the permission levels of adjacent subregions B, C, and D are different from the permission level of the first subregion, and the permission level of adjacent subregion A is the same as the permission level of the first subregion, the target to-be-filtered region is determined as a row pixel region with a width of 1 pixel on the lower side of the adjoining boundary with adjacent subregions B, C, and D and a row pixel region with a width of 1 pixel on the upper side of the adjoining boundary with adjacent subregions B, C, and D, i.e., the shaded region shown in FIG. 16.

[0273] In other embodiments, the target to-be-filtered region is determined when the permission levels of other subregions around the first subregion are higher than the permission level of the first subregion. The target to-be-filtered region can include the following two situations 1 and 2.

[0274] Situation 1: if other subregions with permission levels higher than the first subregion are located on the vertical side of the first subregion, the width of the first side region is n columns of pixels from the adjoining boundary towards the first subregion, and the width of

the second side region is n columns of pixels from the adjoining boundary towards another subregion.

[0275] Situation 2: if other subregions with permission levels higher than the first subregion are located on the horizontal side of the first subregion, the width of the first side region is n rows of pixels from the adjoining boundary towards the first subregion, and the width of the second side region is n rows of pixels from the adjoining boundary towards another subregion.

[0276] The above two situations can exist simultaneously.

[0277] In Sd2, adaptive loop filter is performed on the target to-be-filtered region according to the set filtering sequence.

[0278] In some embodiments, as shown in FIG. 16 above, adaptive loop filter is performed on the target to-be-filtered region (i.e., the shaded region in FIG. 16) in the filtering sequence from right to left and then from top to bottom.

[0279] In some embodiments, adjacent subregions around the first subregion that have not undergone filtering processing maintain their original pixel values. For example, as shown in FIG. 16 above, the pixel values at the boundary between the first subregion and the adjacent subregion A remain unchanged.

[0280] It can be understood that if the permission levels of at least two adjacent picture subregions in multiple picture subregions are different, the target to-be-filtered region can be determined from the surrounding side of the first subregion in at least two picture subregions according to the set filtering sequence, and sample adaptive offset can be performed on the target to-be-filtered region according to the set filtering sequence. Where based on the permission levels of adjacent subregions around the first subregion, the corresponding target to-be-filtered region can be selected for adaptive loop filter. In the case where the picture processing apparatus obtains information about the corresponding picture region based on the permission level of users, this method can meet the needs of users with any permission level to perform adaptive loop filter on the decoded picture, improve the picture quality of different permission regions in the to-be-processed picture, and thus enhance the user experience at different permission levels.

[0281] In some embodiments, filtering process includes neural network-based filtering processing; and if the permission levels of at least two adjacent picture subregions in multiple picture subregions are different, neural network-based filtering processing can be performed on the adjoining boundary between at least two picture subregions. Referring to FIG. 17, the decoding method provided in this embodiment includes the following steps Se1 to Se2.

[0282] In Se1, in a case where neural network-based filtering is required for the first subregion, inputting a pixel value and a permission level of the first subregion into a neural network.

[0283] In some embodiments, the pixel values of the first subregion may be reconstructed pixel values, where the reconstructed pixel values are the pixel values of the first subregion after decoding and filtering processing, which may be the pixel values after decoding and other filtering processing (such as deblocking vertical filtering), which is not limited in this embodiment.

[0284] In some embodiments, when neural network-based filtering is required in the first subregion, the reconstructed pixel values and the permission level of the first subregion, and other decoding information, are inputted into the neural network. Where the other decoding information includes the boundary strength (BS) value mentioned above, the quantization coefficients of the encoding unit transform block, etc, which is not limit in this embodiment.

[0285] In some embodiments, the neural network may be a convolutional neural network (CNN) or other neural networks, which is not limit in this embodiment.

[0286] In some embodiments, step Se1 may include step Se12.

[0287] In Se12, the permission level of at least one picture subregion adjacent to the first subregion is inputted into the neural network.

[0288] For example, the permission levels of adjacent subregions A, B, C, and D around the first subregion in FIG. 16 are inputted into the neural network.

[0289] In Se2, the outputted filtered pixels of the first subregion are obtained.

[0290] As shown in FIG. 18, the reconstructed pixel values of the first subregion, the permission level of the first subregion, the permission level of at least one adjacent picture subregion, and other decoding information are input into the neural network to obtain the outputted filtered reconstructed pixels of the first subregion.

[0291] It can be understood that based on the permission levels of the picture subregions, the corresponding target to-be-filtered region can be selected for neural network-based filtering. In the case where the picture processing apparatus obtains information about the corresponding picture region based on the permission level of users, this method can meet the needs of users with any permission level to perform neural network-based filtering on the decoded picture, improve the picture quality of different permission regions in the to-be-processed picture, and thus enhance the user experience at different permission levels.

[0292] In some embodiments, the process of executing Sa1 to Sa2, Sb1 to Sb2, Sc1 to Sc2, Sd1 to Sd2 can be carried out sequentially, that is, the output of Sa1 to Sa2 is the input of Sb1

to Sb2, the output of Sb1 to Sb2 is the input of Sc1 to Sc2, and the output of Sc1 to Sc2 is the input of Sd1 to Sd2.

[0293] In other embodiments, the process of executing Sa1 to Sa2, Sb1 to Sb2, Sc1 to Sc2, Sd1 to Sd2, and Se1 to Se2 can be carried out sequentially, that is, the output of Sa1 to Sa2 is the input of Sb1 to Sb2, the output of Sb1 to Sb2 is the input of Sc1 to Sc2, the output of Sc1 to Sc2 is the input of Sd1 to Sd2, and the output of Sd1 to Sd2 is the input of Se1 to Se2.

[0294] In other embodiments, the above Sa1 to Sa2, Sb1 to Sb2, Sc1 to Sc2, Sd1 to Sd2, Se1 to Se2 can be executed in any combination.

[0295] It can be understood that based on the permission levels of the picture subregions, the corresponding target to-be-filtered region can be selected for the above filtering process. And according to the filtering processing requirements of the to-be-processed picture, the pixel points on one side or both sides of the adjoining boundary of the picture subregion can be selected for filtering processing. At the same time, different filtering processing methods can be combined to filter the to-be-processed picture in the order of combination, which improves the picture quality of different permission regions in the to-be-processed picture and enhances the user experience.

[0296] The present disclosure further provides an electronic device, as shown in FIG. 19. FIG. 19 is a structural schematic diagram of the electronic device provided in the present disclosure. The electronic device 12 includes a processor(s) 201 and a communication interface(s) 202. The processor(s) 201 and the communication interface(s) 202 are mutually coupled. It can be understood that communication interface 202 can be a transceiver or an input-output interface. In some embodiments, the electronic device 12 may also include a memory(s) 203 for storing instructions executed by the processor(s) 201 or input data required for the processor(s) 201 to execute instructions, or data generated by the processor(s) 201 after executing instructions.

[0297] In some embodiments, the processor(s) 201 and communication interface(s) 202 are used to perform the functions of the picture dividing unit 101, the picture filtering unit 102, and the picture decoding unit 103 as described above.

[0298] The specific connection medium between the communication interface(s) 202, processor(s) 201, and memory(s) 203 is not limited in this embodiment of the present disclosure. In the embodiment of the present disclosure, the communication interface(s) 202, processor(s) 201, and memory(s) 203 are connected through a bus 204 as in FIG. 17. Bus 204 is indicated by a thick line in FIG. 19. The connection method between other components is only for illustrative purposes and is not limited. The bus can be divided into address bus, data

bus, control bus, etc. For ease of representation, only one thick line is used in FIG. 19, but it does not indicate that there is only one bus or one type of bus.

[0299] The memory(s) 203 can be used to store software programs and modules, such as the program instructions/modules corresponding to the decoding or encoding methods provided in the embodiments of the present disclosure. The processor(s) 201 executes various functional applications and data processing by executing the software programs and modules stored in the memory(s) 203. This communication interface(s) 202 can be used for signaling or data communication with other devices. In The present disclosure, the electronic device 12 may have multiple communication interfaces 202.

[0300] It can be understood that the processor in the embodiments of the present disclosure can be a central processing unit (CPU), a neural processing unit (NPU), or a graphics processing unit (GPU), or other general-purpose processors, a digital signal processor (DSP), an application specific integrated circuit (ASIC), a field programmable gate array (FPGA), or other programmable logic devices, or a transistor logic device, a hardware component, or any combination thereof. A general-purpose processor can be a microprocessor or any conventional processor.

[0301] The method steps in the embodiments of the present disclosure can be implemented by hardware or by a processor(s) executing software instructions. Software instructions can be composed of corresponding software modules, which can be stored in random access memory (RAM), flash memory, read-only memory (ROM), programmable read-only memory (PROM), erasable programmable read-only memory (EPROM), electrically erasable programmable read-only memory (EEPROM), registers, hard drives, portable hard drives, CD ROMs, or any other form of storage medium well known in the art. An exemplary storage medium is coupled to a processor, enabling the processor to read information from and write information to the storage medium. Of course, the storage medium can also be a component of the processor. The processor and storage medium can be located in an ASIC. In addition, the ASIC can be located in a network device or a terminal device. Of course, processor(s) and storage medium(s) can also exist as separate components in a network device or a terminal device.

[0302] The above embodiments can be fully or partially implemented through software, hardware, firmware, or any combination thereof. When implemented using software, all or part of the steps can be implemented in the form of a computer program product. The computer program product includes one or more computer programs or instructions. When loading and executing the computer program or instructions on a computer, all or part of the processes or functions described in the embodiments of the present disclosure are executed. The computer

may be a general-purpose computer, a specialized computer, a computer network, a network device, a user equipment, or other programmable device. The computer program or instructions may be stored in a computer-readable storage medium or transmitted from one computer-readable storage medium to another. For example, the computer program or instructions may be transmitted from one website site, computer, server, or data center to another website site, computer, server, or data center via wired or wireless means. The computer-readable storage medium can be any available medium that the computer can access, or a data storage device such as a server, data center, etc. that integrates one or more available media. The available medium may be a magnetic medium, such as a floppy disk, a hard disk, a magnetic tape; the available medium can also be an optical medium, such as a digital video disc (DVD); the available medium can also be a semiconductor medium, such as a solid state drive (SSD).

[0303] In various embodiments of the present disclosure, unless otherwise specified or logically conflicting, the terms and/or descriptions between different embodiments are consistent and can be referenced to each other. The technical features in different embodiments can be combined to form new embodiments based on their inherent logical relationships. In the present disclosure, "at least one" refers to one or more, and "multiple" refers to two or more. 'And/or' describes the association relationship between related objects, indicating that there can be three types of relationships, for example, A and/or B can represent the situation where A exists alone, A and B exist simultaneously, and B exists alone, where A and B can be singular or plural. In the textual description of the present disclosure, the character "/" generally indicates a "or" relationship between the related objects before and after. In the formula of the present disclosure, the character "/" represents a "division" relationship between the related objects before and after.

[0304] It can be understood that the various numerical numbers mentioned in the embodiments of the present disclosure are only for the convenience of description and are not intended to limit the scope of the embodiments of the present disclosure. The sequence numbers of the above processes do not imply the order of execution. The execution order of each process should be determined by its function and internal logic.

CLAIMS

1. A picture processing method, performed by an electronic device, comprising:
 - dividing a to-be-processed picture into a plurality of picture subregions;
 - obtaining a permission level for each of the plurality of picture subregions; and
 - when permission levels of at least two picture subregions which are adjacent in the plurality of picture subregions are different, performing a filtering process on an adjoining boundary between the at least two picture subregions.
2. The method according to claim 1, wherein the permission level comprises a first permission level, a second permission level, or a zero permission level, wherein a picture subregion of the zero permission level is viewable by a user of any permission level.
3. The method according to claim 1 or 2, wherein before dividing the to-be-processed picture into the plurality of picture subregions and obtaining the permission level for each of the plurality of the picture subregions, the method comprises:
 - receiving a decoding bitstream of the to-be-processed picture; and
 - decoding based on the decoding bitstream to obtain the decoded to-be-processed picture and decoding information of the to-be-processed picture.
4. The method according to claim 1 or 2, wherein the filtering process comprises at least one of: deblocking vertical filtering, deblocking horizontal filtering, sample adaptive offset, adaptive loop filter, or neural network filtering.
5. The method according to claim 4, wherein there is a vertical adjoining boundary between the at least two picture subregions, and the filtering process comprises deblocking vertical filtering, wherein performing the filtering process on the adjoining boundary between the at least two picture subregions comprises:
 - determining a target to-be-filtered region on a vertical side of a first subregion in the at least two picture subregions, wherein the vertical side refers to a left side or a right side of the adjoining boundary of the first subregion, and the target to-be-filtered region is any one of: a right to-be-filtered region on the adjoining boundary, a left to-be-filtered region on the adjoining boundary, or a right to-be-filtered region and a left to-be-filtered region on the adjoining boundary; and
 - performing deblocking vertical filtering on the target to-be-filtered region.

6. The method according to claim 5, wherein the at least two adjacent picture subregions further comprise a second subregion, where the second subregion is horizontally adjacent to the first subregion, and the permission levels of the at least two adjacent picture subregions being different comprises:

the permission level of the first subregion is lower than the permission level of the second subregion;

the permission level of the first subregion is higher than the permission level of the second subregion;

the permission level of the first subregion is not equal to the permission level of the second subregion; or

the permission level of the first subregion is not equal to the permission level of the second subregion, and a filtering strength of the adjoining boundary is greater than or equal to a preset filtering strength.

7. The method according to claim 4, wherein there is a horizontal adjoining boundary between the at least two picture subregions, and the filtering process comprises deblocking horizontal filtering, wherein performing the filtering process on the adjoining boundary between the at least two adjacent picture subregions comprises:

determining a target to-be-filtered region on a horizontal side of a first subregion in the at least two picture subregions, wherein the horizontal side refers to an upper side or a lower side of the adjoining boundary of the first subregion, and the target to-be-filtered region is any one of: a lower to-be-filtered region on the adjoining boundary, an upper to-be-filtered region on the adjoining boundary, or a lower to-be-filtered region and an upper to-be-filtered region on the adjoining boundary; and

performing deblocking horizontal filtering on the target to-be-filtered region.

8. The method according to claim 7, wherein the at least two adjacent picture subregions further comprise a third subregion, where the third subregion is vertically adjacent to the first subregion, and the permission levels of the at least two adjacent picture subregions being different comprises:

the permission level of the first subregion is lower than the permission level of the third subregion;

the permission level of the first subregion is higher than the permission level of the third

subregion;

the permission level of the first subregion is not equal to the permission level of the third subregion; or

the permission level of the first subregion is not equal to the permission level of the third subregion, and a filtering strength of the adjoining boundary is greater than or equal to a preset filtering strength.

9. The method according to claim 4, wherein the filtering processing comprises sample adaptive offset, and performing the filtering process on the adjoining boundary between the at least two picture subregions comprises:

determining a target to-be-filtered region on a surrounding side of a first subregion in the at least two picture subregions according to a set filtering sequence, wherein the surrounding side sequentially comprises: a left side, an upper side, a right side, and a lower side on the adjoining boundary between the first subregion and other subregions, the permission level of the first subregion is different from the permission levels of the other subregions, and the target to-be-filtered region is any one of: a first side region on the adjoining boundary within the first subregion, a second side region on the adjoining boundary outside the first subregion, or the first side region and the second side region; and

performing sample adaptive offset on the target to-be-filtered region according to the set filtering sequence.

10. The method according to claim 4, wherein the filtering processing comprises adaptive loop filter, and performing the filtering process on the adjoining boundary between the at least two picture subregions comprises:

determining a target to-be-filtered region on a surrounding side of a first subregion in the at least two picture subregions according to a set filtering sequence, wherein the surrounding side sequentially comprises: a left side, an upper side, a right side, and a lower side on the adjoining boundary between the first subregion and other subregions, the permission level of the first subregion is different from the permission levels of the other subregions, and the target to-be-filtered region is any one of: a first side region on the adjoining boundary within the first subregion, a second side region on the adjoining boundary outside the first subregion, or the first side region and the second side region; and

performing adaptive loop filter on the target to-be-filtered region according to the set filtering sequence.

11. The method according to claim 4, further comprising:

according to the decoding information of the to-be-processed picture, determining whether neural network-based filtering is required for a first subregion of the at least two picture subregions;

in a case where neural network-based filtering is required for the first subregion, inputting a reconstructed pixel value and a permission level of the first subregion into a neural network; and

obtaining an outputted filtered reconstructed pixel of the first subregion.

12. The method according to claim 11, wherein in a case where the neural network-based filtering is required for the first subregion, inputting the reconstructed pixel value and the permission level of the first subregion into the neural network comprises:

inputting the permission level of at least one picture subregion adjacent to the first subregion into the neural network.

13. The method according to claim 1, wherein the electronic device obtains, based on a permission level of a user, information of a picture subregion lower than or equal to the permission level of the user, wherein the information of the picture subregion comprises a pixel value of the picture subregion.

14. The method according to claim 1, wherein when the permission levels of the at least two picture subregions which are adjacent in the plurality of picture subregions are different, performing the filtering process on the adjoining boundary between the at least two picture subregions comprises:

performing a filtering process for a luma channel component on the adjoining boundary between the at least two picture subregions;

performing a filtering process for a chroma channel component on the adjoining boundary between the at least two picture subregions; or

performing a filtering process for a luma channel component and a chroma channel component on the adjoining boundary between the at least two picture subregions.

15. A picture processing apparatus, comprising: a picture dividing unit, a picture filtering unit, and a picture decoding unit; wherein the picture dividing unit, the picture filtering unit, and the

picture decoding unit are configured to implement the method according to any one of claims 1 to 14.

16. An electronic device, comprising: one or more processors and one or more memories, wherein the one or more memories are configured to store computer instructions, and the one or more processors are configured to call the computer instructions from the memory and run the computer instructions to implement the method according to any one of claims 1 to 14.

17. A non-transitory computer-readable storage medium storing a computer program or instructions, wherein an electronic device executes the computer program or instructions to implement the method according to any one of claims 1 to 14.