United States Patent	12395649
Kind Code	B2
Date of Patent	August 19, 2025
Inventor(s)	Zhao; Liang et al.

Multiple linear models employed with chroma-from-luma prediction

Abstract

A method, computer program, and computer system are provided for encoding video data. Luma and chroma samples are identified from image data corresponding to a video frame. In particular, at least one chroma sample may be predicted from at least one luma sample based on a DC contribution from a chroma component of the image data and an AC contribution from a luma component of the image data. Also, a plurality of linear models may be determined from the identified luma and chroma samples. The luma and chroma samples may be classified into a number of categories corresponding to a number of linear models, where each category has at least one corresponding linear model parameters signaled to the bitstream; the chroma sample prediction may be further based on such parameters.

Inventors: **Zhao; Liang (Sunnyvale, CA), Zhao; Xin (San Jose, CA), Liu; Shan (San Jose, CA)**

Applicant: **TENCENT AMERICA LLC** (Palo Alto, CA)

Family ID: **1000008765522**

Assignee: **TENCENT AMERICA LLC (Palo Alto, CA)**

Appl. No.: **18/622691**

Filed: **March 29, 2024**

Prior Publication Data

Document Identifier	Publication Date
US 20240244235 A1	Jul. 18, 2024

Related U.S. Application Data

continuation parent-doc US 18078285 20221209 US 12003739 child-doc US 18622691
continuation parent-doc US 17459674 20210827 US 11575918 20230207 child-doc US 18078285

continuation parent-doc US 16917146 20200630 US 11146801 20211012 child-doc US 17459674

Publication Classification

Int. Cl.: **H04N19/186** (20140101); **H04N19/136** (20140101); **H04N19/159** (20140101); **H04N19/172** (20140101)

U.S. Cl.:

CPC **H04N19/186** (20141101); **H04N19/136** (20141101); **H04N19/159** (20141101); **H04N19/172** (20141101);

Field of Classification Search

CPC: H04N (19/186)

References Cited

U.S. PATENT DOCUMENTS

Patent No.	Issued Date	Patentee Name	U.S. Cl.	CPC
2016/0198190	12/2015	Budagavi	375/240.12	H04N 19/44
2017/0272749	12/2016	Pettersson	N/A	H04N 19/17
2017/0347102	12/2016	Panusopone et al.	N/A	N/A
2017/0359595	12/2016	Zhang	N/A	H04N 19/186
2018/0077426	12/2017	Zhang	N/A	H04N 19/157

OTHER PUBLICATIONS

Benjamin Bross et al., “Versatile Video Coding (Draft 7)”, Joint Video Experts Team (JVET) of ITU-T Sg 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JVET-P2001-vE, Oct. 1-11, 2019, 16th Meeting: Geneva, CH. cited by applicant
Extended European Search Report and Search Opinion received for European Application No. 21833167.6, mailed on Sep. 4, 2023, 12 pages. cited by applicant
International Search Report and Written Opinion received for PCT Patent Application No. PCT/US2021/026795, mailed on May 12, 2021, 11 pages. cited by applicant
Peter De Rivaz et al., “AV1 Bitstream & Decoding Process Specification”, The Alliance for Open Media, 2018, 681 pgs. cited by applicant
Trudeau, L. et al., “Predicting Chroma from Luma in AV1”, Arxiv.org, Cornell University Library, Nov. 10, 2017, 9 pages. cited by applicant
Yue Chen et al., “An Overview of Core Coding Tools in the AV1 Video Codec”, IEEE, 2018, 5 pgs. cited by applicant

Primary Examiner: Jean Baptiste; Jerry T

Attorney, Agent or Firm: ArentFox Schiff LLP

Background/Summary

CROSS-REFERENCE TO RELATED APPLICATIONS (1) This application is a continuation of U.S. application Ser. No. 18/078,285, “IMPROVED CROSS COMPONENT INTRA PREDICTION MODE” filed on Dec. 9, 2022, which is a continuation of U.S. application Ser. No. 17/459,674, “CROSS COMPONENT INTRA PREDICTION MODE” filed on Aug. 27, 2021, which is a continuation of U.S. application Ser. No. 16/917,146, “CROSS COMPONENT INTRA PREDICTION MODE” filed on Jun. 30, 2020, in the United States patent and Trademark Office. Each of the above-listed applications is herein incorporated by reference in their entirety.

FIELD

(1) This disclosure relates generally to field of data processing, and more particularly to video encoding and decoding.

BACKGROUND

(2) AOMedia Video 1 (AV1) is an open video coding format designed for video transmissions over the Internet. It was developed as a successor to VP9 by the Alliance for Open Media (AOMedia), a consortium founded in 2015 that includes semiconductor firms, video on demand providers, video content producers, software development companies and web browser vendors. AV1 specifies a chroma-from-luma mode that allows for chroma-only intra prediction and may model chroma pixels as a linear function of coincident reconstructed luma pixels.

SUMMARY

(3) Embodiments relate to a method, system, and computer readable medium for encoding video data. According to one aspect, a method for encoding video data is provided. The method may include receiving data corresponding to a video frame. One or more luma and chroma samples from are identified from the received image data. Two or more linear models are determined from the identified luma and chroma samples. The luma and chroma samples are classified into a number of categories corresponding to a number of linear models, and each category has one or more corresponding linear model parameters signaled to the bitstream.

(4) According to another aspect, a computer system for encoding video data is provided. The computer system may include one or more processors, one or more computer-readable memories, one or more computer-readable tangible storage devices, and program instructions stored on at least one of the one or more storage devices for execution by at least one of the one or more processors via at least one of the one or more memories, whereby the computer system is capable of performing a method. The method may include receiving data corresponding to a video frame. One or more luma and chroma samples from are identified from the received image data. Two or more linear models are determined from the identified luma and chroma samples. The luma and chroma samples are classified into a number of categories corresponding to a number of linear models, and each category has one or more corresponding linear model parameters signaled to the bitstream.

(5) According to yet another aspect, a computer readable medium for encoding video data is provided. The computer readable medium may include one or more computer-readable storage devices and program instructions stored on at least one of the one or more tangible storage devices, the program instructions executable by a processor. The program instructions are executable by a processor for performing a method that may accordingly include receiving data corresponding to a video frame. One or more luma and chroma samples from are identified from the received image data. Two or more linear models are determined from the identified luma and chroma samples. The luma and chroma samples are classified into a number of categories corresponding to a number of linear models, and each category has one or more corresponding linear model parameters signaled to the bitstream.

Description

BRIEF DESCRIPTION OF THE DRAWINGS

(1) These and other objects, features and advantages will become apparent from the following detailed description of illustrative embodiments, which is to be read in connection with the accompanying drawings. The various features of the drawings are not to scale as the illustrations are for clarity in facilitating the understanding of one skilled in the art in conjunction with the detailed description. In the drawings:

(2) FIG. **1** illustrates a networked computer environment according to at least one embodiment;

(3) FIG. **2** is a function block diagram of a chroma-from-luma prediction process, according to at least one embodiment;

(4) FIG. **3** is an operational flowchart illustrating the steps carried out by a program that encodes video data based on multiple linear models, according to at least one embodiment;

(5) FIG. **4** is a block diagram of internal and external components of computers and servers depicted in FIG. **1** according to at least one embodiment;

(6) FIG. **5** is a block diagram of an illustrative cloud computing environment including the computer system depicted in FIG. **1**, according to at least one embodiment; and

(7) FIG. **6** is a block diagram of functional layers of the illustrative cloud computing environment of FIG. **5**, according to at least one embodiment.

DETAILED DESCRIPTION

(8) Detailed embodiments of the claimed structures and methods are disclosed herein; however, it can be understood that the disclosed embodiments are merely illustrative of the claimed structures and methods that may be embodied in various forms. Those structures and methods may however, be embodied in many different forms and should not be construed as limited to the exemplary embodiments set forth herein. Rather, these exemplary embodiments are provided so that this disclosure will be thorough and complete and will fully convey the scope to those skilled in the art. In the description, details of well-known features and techniques may be omitted to avoid unnecessarily obscuring the presented embodiments.

(9) Embodiments relate generally to the field of data processing, and more particularly to video encoding and decoding. The following described exemplary embodiments provide a system, method and computer program to, among other things, encode videos using multiple linear models for chroma-from-luma prediction. Therefore, some embodiments have the capacity to improve the field of computing by allowing for improved chroma-from-luma prediction by considering luma samples within an image boundary and calculating average luma values based on multiple categories of parameters.

(10) As previously described, AOMedia Video 1 (AV1) is an open video coding format designed for video transmissions over the Internet. It was developed as a successor to VP9 by the Alliance for Open Media (AOMedia), a consortium founded in 2015 that includes semiconductor firms, video on demand providers, video content producers, software development companies and web browser vendors. AV1 specifies a chroma-from-luma mode that allows for chroma-only intra prediction and may model chroma pixels as a linear function of coincident reconstructed luma pixels. However, in chroma-from-luma mode, only one linear model may be employed between luma and chroma samples within one whole coded block, but the relationship between luma and chroma samples within one whole coded block may not always be well fitted by one single linear model. Additionally, when some samples in co-located luma blocks are out of a picture boundary, these samples may be padded and used to calculate the average of luma samples, which may cause increased complexity. Moreover, all the samples in the corresponding luma blocks may be used to calculate the average luma values when chroma-from-luma mode may be selected, which may be complex when current block may be equal to or greater than 32 pixels by 32 pixels. It may be advantageous, therefore, to determine one or more linear models for the luma and chroma samples in order to improve cross component intra prediction and decrease complexity costs.

(11) Aspects are described herein with reference to flowchart illustrations and/or block diagrams of methods, apparatus (systems), and computer readable media according to the various embodiments. It will be understood that each block of the flowchart illustrations and/or block diagrams, and combinations of blocks in the flowchart illustrations and/or block diagrams, can be implemented by computer readable program instructions.
(12) Referring now to FIG. **1**, a functional block diagram of a networked computer environment illustrating a video frame encoding system **100** (hereinafter “system”) for encoding video data based on multiple linear models. It should be appreciated that FIG. **1** provides only an illustration of one implementation and does not imply any limitations with regard to the environments in which different embodiments may be implemented. Many modifications to the depicted environments may be made based on design and implementation requirements.
(13) The system **100** may include a computer **102** and a server computer **114**. The computer **102** may communicate with the server computer **114** via a communication network **110** (hereinafter “network”). The computer **102** may include a processor **104** and a software program **108** that is stored on a data storage device **106** and is enabled to interface with a user and communicate with the server computer **114**. As will be discussed below with reference to FIG. **4** the computer **102** may include internal components **800**A and external components **900**A, respectively, and the server computer **114** may include internal components **800**B and external components **900**B, respectively. The computer **102** may be, for example, a mobile device, a telephone, a personal digital assistant, a netbook, a laptop computer, a tablet computer, a desktop computer, or any type of computing devices capable of running a program, accessing a network, and accessing a database.
(14) The server computer **114** may also operate in a cloud computing service model, such as Software as a Service (SaaS), Platform as a Service (PaaS), or Infrastructure as a Service (IaaS), as discussed below with respect to FIGS. **6** and **7**. The server computer **114** may also be located in a cloud computing deployment model, such as a private cloud, community cloud, public cloud, or hybrid cloud.
(15) The server computer **114**, which may be used for encoding video data is enabled to run a Video Encoding Program **116** (hereinafter “program”) that may interact with a database **112**. The Video Encoding Program method is explained in more detail below with respect to FIG. **3**. In one embodiment, the computer **102** may operate as an input device including a user interface while the program **116** may run primarily on server computer **114**. In an alternative embodiment, the program **116** may run primarily on one or more computers **102** while the server computer **114** may be used for processing and storage of data used by the program **116**. It should be noted that the program **116** may be a standalone program or may be integrated into a larger video encoding program.
(16) It should be noted, however, that processing for the program **116** may in some instances be shared amongst the computers **102** and the server computers **114** in any ratio. In another embodiment, the program **116** may operate on more than one computer, server computer, or some combination of computers and server computers, for example, a plurality of computers **102** communicating across the network **110** with a single server computer **114**. In another embodiment, for example, the program **116** may operate on a plurality of server computers **114** communicating across the network **110** with a plurality of client computers. Alternatively, the program may operate on a network server communicating across the network with a server and a plurality of client computers.
(17) The network **110** may include wired connections, wireless connections, fiber optic connections, or some combination thereof. In general, the network **110** can be any combination of connections and protocols that will support communications between the computer **102** and the server computer **114**. The network **110** may include various types of networks, such as, for example, a local area network (LAN), a wide area network (WAN) such as the Internet, a telecommunication network such as the Public Switched Telephone Network (PSTN), a wireless network, a public switched network, a satellite network, a cellular network (e.g., a fifth generation

(5G) network, a long-term evolution (LTE) network, a third generation (3G) network, a code division multiple access (CDMA) network, etc.), a public land mobile network (PLMN), a metropolitan area network (MAN), a private network, an ad hoc network, an intranet, a fiber optic-based network, or the like, and/or a combination of these or other types of networks.

(18) The number and arrangement of devices and networks shown in FIG. **1** are provided as an example. In practice, there may be additional devices and/or networks, fewer devices and/or networks, different devices and/or networks, or differently arranged devices and/or networks than those shown in FIG. **1**. Furthermore, two or more devices shown in FIG. **1** may be implemented within a single device, or a single device shown in FIG. **1** may be implemented as multiple, distributed devices. Additionally, or alternatively, a set of devices (e.g., one or more devices) of system **100** may perform one or more functions described as being performed by another set of devices of system **100**.

(19) Referring now to FIG. **2**, a function block diagram **200** for a chroma-from-luma prediction process is depicted. Chroma-from-luma prediction may be expressed as CfL(α)=α×L.sub.AC+DC, where L.sub.AC may denote an AC contribution of luma component, α may denote a parameter of the linear model, and DC denotes a DC contribution of the chroma component. Reconstructed luma pixels may be subsampled by the subsampling module **202** into the chroma resolution, and the average value determined by the averaging module **204** may be subtracted to form the AC contribution by the subtraction module **206**. To approximate the chroma AC component from the AC contribution, the parameter α is determined based on the original chroma pixels and signals them in the bitstream. The parameter α may be multiplied by the AC contribution by the multiplication module **208**. The DC contribution of the chroma component may be computed using intra DC mode and may be added to the output of the multiplication module **208** by the addition module **210**.

(20) Multiple linear models (MLM) can be employed on top of chroma-from-luma between luma and chroma samples within one whole coded blocks, whereby a number of linear models may be denoted as N, where N>1. When MLM mode may be selected, the samples within the luma and chroma block may be classified into N categories, and each category may have its own linear model parameters. For example, the parameter α of N models may be signaled to the bitstream for each of the models. The samples in chroma block may be classified into N models based on their coordinates in the coded block. According to one or more embodiments, the samples in the even rows or columns may be classified into a first category and the samples in odd rows or columns may be classified into category 1. In one or more embodiments, the samples may be classified into two categories by one straight and one cross current block.

(21) In one or more embodiments, the samples in chroma block may be classified into N models based on the values of the corresponding reconstructed luma samples. In one or more embodiments, one threshold T1 may be calculated based on the corresponding luma samples, whereby T1 may be the average or median values of the samples within corresponding luma block. Each corresponding luma samples may be compared to the threshold T1 to determine its category. For example, when the values of current luma samples may be smaller than T1, its corresponding chroma samples may be classified into the first category. Otherwise, this sample may be classified into the second category.

(22) In one or more embodiments, the values of corresponding luma samples may be compared to a DC value to determine its category. The DC value may be median value of the pixel value range, for example, the DC value may be 128 when the internal bit-depth of the codec may be 8, and the DC value may be 512 when the internal bit-depth of the codec may be 10. In one or more embodiments, the parameters of the first linear model may be directly signaled, and the difference of parameters α with the first model may be signaled for the remaining linear models. In one or more embodiments, for the remaining linear models, the difference of the parameters with the first model may be restricted to [−K, K], K may be a positive integer, such as 1 or 2.

(23) In one or more embodiments, the average luma values for each linear model may be computed by using the samples belonging to each individual category. In one or more embodiments, multiple linear models (MLM) and chroma-from-luma may be signaled together in one flag, namely cross component mode (CCM). When the current mode may be CCM, then one additional flag may be signaled to indicate whether current mode may be MLM or chroma-from-luma. In one or more embodiments, MLM and chroma-from-luma may be signaled together with nominal angles and non-directional modes.
(24) When some of the corresponding luma samples may be out of the picture, these out-of-picture luma samples may be not padded or used for calculating the average luma values for chroma-from-luma mode. In one or more embodiments, all and only the samples within the picture may be used for calculating the average luma values for chroma-from-luma mode. In one or more embodiments, only the first M rows (and/or N columns) samples within the picture may be used for calculating the average luma values for chroma-from-luma mode, where M or N may be less than or equal to the height or width of a block, and the value of M and N may be both powers of 2.
(25) Average luma values may be computed by using the samples in one or multiple predefined positions when chroma-from-luma mode may be used. In one or more embodiments, after down-sampling process, the luma samples in the even/odd rows (or columns) may be utilized to calculate the average value. In one or more embodiments, after down-sampling process, the luma samples in corner positions and/or center positions may be utilized to calculate the average value.
(26) One or multiple predicted linear model parameters (e.g., slope values and offset values) may be derived using neighboring reconstructed luma and chroma samples, then one or multiple delta values between the actually used linear model parameters may be signaled. In one or more embodiments, the delta values may be signaled using unary code. In one or more embodiments, the delta values may be restricted to a predefined range, such as [−L,L], such that the computed delta values may be out of this range, it may be clipped to −L or L.
(27) Referring now to FIG. **3**, an operational flowchart **300** illustrating the steps carried out by a program that encodes video data is depicted. FIG. **3** may be described with the aid of
(28) FIGS. **1** and **2**. As previously described, the Video Encoding Program **116** (FIG. **1**) may quickly and effectively encode videos using chroma-from-luma prediction based on multiple linear models.
(29) At **302**, data corresponding to a video frame is received. The data may be a still image or may video data from which one or more frames may be extracted. In operation, the Video Encoding Program **116** (FIG. **1**) on the server computer **114** (FIG. **1**) may receive video frame data from the computer **102** (FIG. **1**) over the communication network **110** (FIG. **1**) or may retrieve the video frame data from the database **112** (FIG. **1**).
(30) At **304**, one or more luma and chroma samples are identified from the received image data. The luma and chroma samples may allow for intra frame prediction to allow for compression of the video frame data. In operation, the Video Encoding Program **116** (FIG. **1**) may extract the luma and chroma samples from the received video frame data. The extracted luma and chroma samples may be stored within the database **112** (FIG. **1**).
(31) At **306**, two or more linear models are determined from the identified luma and chroma samples. The luma and chroma samples are classified into a number of categories corresponding to a number of linear models, and each category has one or more corresponding linear model parameters signaled to the bitstream. In operation, the Video Encoding Program **116** (FIG. **1**) may determine one or more linear models based on the luma and chroma samples stored in the database **112** (FIG. **1**) or the chroma-from-luma prediction output of the multiplication module **208** (FIG. **2**).
(32) It may be appreciated that FIG. **3** provides only an illustration of one implementation and does not imply any limitations with regard to how different embodiments may be implemented. Many modifications to the depicted environments may be made based on design and implementation requirements.

(33) FIG. **4** is a block diagram **400** of internal and external components of computers depicted in FIG. **1** in accordance with an illustrative embodiment. It should be appreciated that FIG. **4** provides only an illustration of one implementation and does not imply any limitations with regard to the environments in which different embodiments may be implemented. Many modifications to the depicted environments may be made based on design and implementation requirements.

(34) Computer **102** (FIG. **1**) and server computer **114** (FIG. **1**) may include respective sets of internal components **800**A,B and external components **900**A,B illustrated in FIG. **4**. Each of the sets of internal components **800** include one or more processors **820**, one or more computer-readable RAMs **822** and one or more computer-readable ROMs **824** on one or more buses **826**, one or more operating systems **828**, and one or more computer-readable tangible storage devices **830**.

(35) Processor **820** is implemented in hardware, firmware, or a combination of hardware and software. Processor **820** is a central processing unit (CPU), a graphics processing unit (GPU), an accelerated processing unit (APU), a microprocessor, a microcontroller, a digital signal processor (DSP), a field-programmable gate array (FPGA), an application-specific integrated circuit (ASIC), or another type of processing component. In some implementations, processor **820** includes one or more processors capable of being programmed to perform a function. Bus **826** includes a component that permits communication among the internal components **800**A,B.

(36) The one or more operating systems **828**, the software program **108** (FIG. **1**) and the Video Encoding Program **116** (FIG. **1**) on server computer **114** (FIG. **1**) are stored on one or more of the respective computer-readable tangible storage devices **830** for execution by one or more of the respective processors **820** via one or more of the respective RAMs **822** (which typically include cache memory). In the embodiment illustrated in FIG. **4**, each of the computer-readable tangible storage devices **830** is a magnetic disk storage device of an internal hard drive. Alternatively, each of the computer-readable tangible storage devices **830** is a semiconductor storage device such as ROM **824**, EPROM, flash memory, an optical disk, a magneto-optic disk, a solid state disk, a compact disc (CD), a digital versatile disc (DVD), a floppy disk, a cartridge, a magnetic tape, and/or another type of non-transitory computer-readable tangible storage device that can store a computer program and digital information.

(37) Each set of internal components **800**A,B also includes a R/W drive or interface **832** to read from and write to one or more portable computer-readable tangible storage devices **936** such as a CD-ROM, DVD, memory stick, magnetic tape, magnetic disk, optical disk or semiconductor storage device. A software program, such as the software program **108** (FIG. **1**) and the Video Encoding Program **116** (FIG. **1**) can be stored on one or more of the respective portable computer-readable tangible storage devices **936**, read via the respective R/W drive or interface **832** and loaded into the respective hard drive **830**.

(38) Each set of internal components **800**A,B also includes network adapters or interfaces **836** such as a TCP/IP adapter cards; wireless Wi-Fi interface cards; or 3G, 4G, or 5G wireless interface cards or other wired or wireless communication links. The software program **108** (FIG. **1**) and the Video Encoding Program **116** (FIG. **1**) on the server computer **114** (FIG. **1**) can be downloaded to the computer **102** (FIG. **1**) and server computer **114** from an external computer via a network (for example, the Internet, a local area network or other, wide area network) and respective network adapters or interfaces **836**. From the network adapters or interfaces **836**, the software program **108** and the Video Encoding Program **116** on the server computer **114** are loaded into the respective hard drive **830**. The network may comprise copper wires, optical fibers, wireless transmission, routers, firewalls, switches, gateway computers and/or edge servers.

(39) Each of the sets of external components **900**A,B can include a computer display monitor **920**, a keyboard **930**, and a computer mouse **934**. External components **900**A,B can also include touch screens, virtual keyboards, touch pads, pointing devices, and other human interface devices. Each of the sets of internal components **800**A,B also includes device drivers **840** to interface to computer display monitor **920**, keyboard **930** and computer mouse **934**. The device drivers **840**, R/W drive or

interface **832** and network adapter or interface **836** comprise hardware and software (stored in storage device **830** and/or ROM **824**).

(40) It is understood in advance that although this disclosure includes a detailed description on cloud computing, implementation of the teachings recited herein are not limited to a cloud computing environment. Rather, some embodiments are capable of being implemented in conjunction with any other type of computing environment now known or later developed.

(41) Cloud computing is a model of service delivery for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, network bandwidth, servers, processing, memory, storage, applications, virtual machines, and services) that can be rapidly provisioned and released with minimal management effort or interaction with a provider of the service. This cloud model may include at least five characteristics, at least three service models, and at least four deployment models.

(42) Characteristics are as follows:

(43) On-demand self-service: a cloud consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with the service's provider.

(44) Broad network access: capabilities are available over a network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, laptops, and PDAs).

(45) Resource pooling: the provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to demand. There is a sense of location independence in that the consumer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter).

(46) Rapid elasticity: capabilities can be rapidly and elastically provisioned, in some cases automatically, to quickly scale out and rapidly released to quickly scale in. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be purchased in any quantity at any time.

(47) Measured service: cloud systems automatically control and optimize resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported providing transparency for both the provider and consumer of the utilized service.

(48) Service Models are as follows:

(49) Software as a Service (SaaS): the capability provided to the consumer is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through a thin client interface such as a web browser (e.g., web-based e-mail). The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings.

(50) Platform as a Service (PaaS): the capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including networks, servers, operating systems, or storage, but has control over the deployed applications and possibly application hosting environment configurations.

(51) Infrastructure as a Service (IaaS): the capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over

operating systems, storage, deployed applications, and possibly limited control of select networking components (e.g., host firewalls).
(52) Deployment Models are as follows:
(53) Private cloud: the cloud infrastructure is operated solely for an organization. It may be managed by the organization or a third party and may exist on-premises or off-premises.
(54) Community cloud: the cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be managed by the organizations or a third party and may exist on-premises or off-premises.
(55) Public cloud: the cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.
(56) Hybrid cloud: the cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load-balancing between clouds).
(57) A cloud computing environment is service oriented with a focus on statelessness, low coupling, modularity, and semantic interoperability. At the heart of cloud computing is an infrastructure comprising a network of interconnected nodes.
(58) Referring to FIG. **5**, illustrative cloud computing environment **500** is depicted. As shown, cloud computing environment **500** comprises one or more cloud computing nodes **10** with which local computing devices used by cloud consumers, such as, for example, personal digital assistant (PDA) or cellular telephone **54**A, desktop computer **54**B, laptop computer **54**C, and/or automobile computer system **54**N may communicate. Cloud computing nodes **10** may communicate with one another. They may be grouped (not shown) physically or virtually, in one or more networks, such as Private, Community, Public, or Hybrid clouds as described hereinabove, or a combination thereof. This allows cloud computing environment **500** to offer infrastructure, platforms and/or software as services for which a cloud consumer does not need to maintain resources on a local computing device. It is understood that the types of computing devices **54**A-N shown in FIG. **5** are intended to be illustrative only and that cloud computing nodes **10** and cloud computing environment **500** can communicate with any type of computerized device over any type of network and/or network addressable connection (e.g., using a web browser).
(59) Referring to FIG. **6**, a set of functional abstraction layers **600** provided by cloud computing environment **500** (FIG. **5**) is shown. It should be understood in advance that the components, layers, and functions shown in FIG. **6** are intended to be illustrative only and embodiments are not limited thereto. As depicted, the following layers and corresponding functions are provided:
(60) Hardware and software layer **60** includes hardware and software components. Examples of hardware components include: mainframes **61**; RISC (Reduced Instruction Set Computer) architecture based servers **62**; servers **63**; blade servers **64**; storage devices **65**; and networks and networking components **66**. In some embodiments, software components include network application server software **67** and database software **68**.
(61) Virtualization layer **70** provides an abstraction layer from which the following examples of virtual entities may be provided: virtual servers **71**; virtual storage **72**; virtual networks **73**, including virtual private networks; virtual applications and operating systems **74**; and virtual clients **75**.
(62) In one example, management layer **80** may provide the functions described below. Resource provisioning **81** provides dynamic procurement of computing resources and other resources that are utilized to perform tasks within the cloud computing environment. Metering and Pricing **82** provide cost tracking as resources are utilized within the cloud computing environment, and billing or invoicing for consumption of these resources. In one example, these resources may comprise application software licenses. Security provides identity verification for cloud consumers and tasks,

as well as protection for data and other resources. User portal **83** provides access to the cloud computing environment for consumers and system administrators. Service level management **84** provides cloud computing resource allocation and management such that required service levels are met. Service Level Agreement (SLA) planning and fulfillment **85** provide pre-arrangement for, and procurement of, cloud computing resources for which a future requirement is anticipated in accordance with an SLA.

(63) Workloads layer **90** provides examples of functionality for which the cloud computing environment may be utilized. Examples of workloads and functions which may be provided from this layer include: mapping and navigation **91**; software development and lifecycle management **92**; virtual classroom education delivery **93**; data analytics processing **94**; transaction processing **95**; and Video Encoding **96**. Video Encoding **96** may encode video data based on multiple linear models for chroma-from-luma prediction.

(64) Some embodiments may relate to a system, a method, and/or a computer readable medium at any possible technical detail level of integration. The computer readable medium may include a computer-readable non-transitory storage medium (or media) having computer readable program instructions thereon for causing a processor to carry out operations.

(65) The computer readable storage medium can be a tangible device that can retain and store instructions for use by an instruction execution device. The computer readable storage medium may be, for example, but is not limited to, an electronic storage device, a magnetic storage device, an optical storage device, an electromagnetic storage device, a semiconductor storage device, or any suitable combination of the foregoing. A non-exhaustive list of more specific examples of the computer readable storage medium includes the following: a portable computer diskette, a hard disk, a random access memory (RAM), a read-only memory (ROM), an erasable programmable read-only memory (EPROM or Flash memory), a static random access memory (SRAM), a portable compact disc read-only memory (CD-ROM), a digital versatile disk (DVD), a memory stick, a floppy disk, a mechanically encoded device such as punch-cards or raised structures in a groove having instructions recorded thereon, and any suitable combination of the foregoing. A computer readable storage medium, as used herein, is not to be construed as being transitory signals per se, such as radio waves or other freely propagating electromagnetic waves, electromagnetic waves propagating through a waveguide or other transmission media (e.g., light pulses passing through a fiber-optic cable), or electrical signals transmitted through a wire.

(66) Computer readable program instructions described herein can be downloaded to respective computing/processing devices from a computer readable storage medium or to an external computer or external storage device via a network, for example, the Internet, a local area network, a wide area network and/or a wireless network. The network may comprise copper transmission cables, optical transmission fibers, wireless transmission, routers, firewalls, switches, gateway computers and/or edge servers. A network adapter card or network interface in each computing/processing device receives computer readable program instructions from the network and forwards the computer readable program instructions for storage in a computer readable storage medium within the respective computing/processing device.

(67) Computer readable program code/instructions for carrying out operations may be assembler instructions, instruction-set-architecture (ISA) instructions, machine instructions, machine dependent instructions, microcode, firmware instructions, state-setting data, configuration data for integrated circuitry, or either source code or object code written in any combination of one or more programming languages, including an object oriented programming language such as Smalltalk, C++, or the like, and procedural programming languages, such as the "C" programming language or similar programming languages. The computer readable program instructions may execute entirely on the user's computer, partly on the user's computer, as a stand-alone software package, partly on the user's computer and partly on a remote computer or entirely on the remote computer or server. In the latter scenario, the remote computer may be connected to the user's computer

through any type of network, including a local area network (LAN) or a wide area network (WAN), or the connection may be made to an external computer (for example, through the Internet using an Internet Service Provider). In some embodiments, electronic circuitry including, for example, programmable logic circuitry, field-programmable gate arrays (FPGA), or programmable logic arrays (PLA) may execute the computer readable program instructions by utilizing state information of the computer readable program instructions to personalize the electronic circuitry, in order to perform aspects or operations.

(68) These computer readable program instructions may be provided to a processor of a general purpose computer, special purpose computer, or other programmable data processing apparatus to produce a machine, such that the instructions, which execute via the processor of the computer or other programmable data processing apparatus, create means for implementing the functions/acts specified in the flowchart and/or block diagram block or blocks. These computer readable program instructions may also be stored in a computer readable storage medium that can direct a computer, a programmable data processing apparatus, and/or other devices to function in a particular manner, such that the computer readable storage medium having instructions stored therein comprises an article of manufacture including instructions which implement aspects of the function/act specified in the flowchart and/or block diagram block or blocks.

(69) The computer readable program instructions may also be loaded onto a computer, other programmable data processing apparatus, or other device to cause a series of operational steps to be performed on the computer, other programmable apparatus or other device to produce a computer implemented process, such that the instructions which execute on the computer, other programmable apparatus, or other device implement the functions/acts specified in the flowchart and/or block diagram block or blocks.

(70) The flowchart and block diagrams in the Figures illustrate the architecture, functionality, and operation of possible implementations of systems, methods, and computer readable media according to various embodiments. In this regard, each block in the flowchart or block diagrams may represent a module, segment, or portion of instructions, which comprises one or more executable instructions for implementing the specified logical function(s). The method, computer system, and computer readable medium may include additional blocks, fewer blocks, different blocks, or differently arranged blocks than those depicted in the Figures. In some alternative implementations, the functions noted in the blocks may occur out of the order noted in the Figures. For example, two blocks shown in succession may in fact, be executed concurrently or substantially concurrently, or the blocks may sometimes be executed in the reverse order, depending upon the functionality involved. It will also be noted that each block of the block diagrams and/or flowchart illustration, and combinations of blocks in the block diagrams and/or flowchart illustration, can be implemented by special purpose hardware-based systems that perform the specified functions or acts or carry out combinations of special purpose hardware and computer instructions.

(71) It will be apparent that systems and/or methods, described herein, may be implemented in different forms of hardware, firmware, or a combination of hardware and software. The actual specialized control hardware or software code used to implement these systems and/or methods is not limiting of the implementations. Thus, the operation and behavior of the systems and/or methods were described herein without reference to specific software code—it being understood that software and hardware may be designed to implement the systems and/or methods based on the description herein.

(72) No element, act, or instruction used herein should be construed as critical or essential unless explicitly described as such. Also, as used herein, the articles “a” and “an” are intended to include one or more items, and may be used interchangeably with “one or more.” Furthermore, as used herein, the term “set” is intended to include one or more items (e.g., related items, unrelated items, a combination of related and unrelated items, etc.), and may be used interchangeably with “one or

more.” Where only one item is intended, the term “one” or similar language is used. Also, as used herein, the terms “has,” “have,” “having,” or the like are intended to be open-ended terms. Further, the phrase “based on” is intended to mean “based, at least in part, on” unless explicitly stated otherwise.

(73) The descriptions of the various aspects and embodiments have been presented for purposes of illustration, but are not intended to be exhaustive or limited to the embodiments disclosed. Even though combinations of features are recited in the claims and/or disclosed in the specification, these combinations are not intended to limit the disclosure of possible implementations. In fact, many of these features may be combined in ways not specifically recited in the claims and/or disclosed in the specification. Although each dependent claim listed below may directly depend on only one claim, the disclosure of possible implementations includes each dependent claim in combination with every other claim in the claim set. Many modifications and variations will be apparent to those of ordinary skill in the art without departing from the scope of the described embodiments. The terminology used herein was chosen to best explain the principles of the embodiments, the practical application or technical improvement over technologies found in the marketplace, or to enable others of ordinary skill in the art to understand the embodiments disclosed herein.

Claims

1. A method of video encoding, comprising: classifying at least one chroma sample of a current block to be encoded into a first category of plural categories based on at least one of (i) a position of the at least one chroma sample or (ii) values of luma samples corresponding to the at least one chroma sample; selecting a first linear model of plural linear models according to the first category into which the at least one chroma sample is classified; and encoding the at least one chroma sample of the current block using a chroma-from-luma prediction mode based on the selected first linear model, wherein the at least one chroma sample is predicted from at least one luma sample based on a DC contribution from a chroma component associated with the current block and an AC contribution from a luma component associated with the current block.
2. The method of claim 1, wherein the at least one chroma sample is classified into the first category based on coordinates associated with the at least one chroma sample.
3. The method of claim 2, wherein chroma samples corresponding to even rows or columns in the current block are classified into the first category and chroma samples corresponding to odd rows or columns in the current block are classified into a second category.
4. The method of claim 1, wherein the at least one chroma sample is classified into the first category based on values of corresponding reconstructed luma samples.
5. The method of claim 1, wherein parameters of one linear model from among the plural linear models are directly signaled, and a difference between the directly signaled parameters of the one linear model and parameters of remaining linear models of the plural linear models are signaled.
6. The method of claim 1, wherein one or more average luma values for a respective linear model of the plural linear models are computed by using samples belonging to a category corresponding to the respective linear model.
7. The method of claim 1, wherein a first flag signals a cross-component mode enabling use of the plural linear models and the chroma-from-luma prediction mode and a second flag signals whether a current mode corresponds to the use of the plural linear models and the chroma-from-luma prediction mode.
8. The method of claim 1, wherein one or more luma samples outside of boundaries of the current block are not used for calculating average luma values for the chroma-from-luma prediction mode.
9. The method of claim 1, wherein one or more average luma values are computed by using the values of luma samples corresponding to the at least one chroma sample in predefined positions when the chroma-from-luma prediction mode is used.

10. The method of claim 1, wherein parameters of the plural linear models are derived using neighboring reconstructed luma and chroma samples.
11. A method for video decoding, comprising: receiving coded information for a current block in a coded video bitstream, wherein the coded information indicates use of plural linear models in a chroma-from-luma prediction mode; classifying at least one chroma sample of the current block into a first category of plural categories based on at least one of (i) a position of the at least one chroma sample or (ii) values of luma samples corresponding to the at least one chroma sample; selecting a first linear model of the plural linear models according to the first category into which the at least one chroma sample is classified; and reconstructing the at least one chroma sample of the current block using the chroma-from-luma prediction mode based on the selected first linear model, wherein the at least one chroma sample is reconstructed from at least one luma sample based on a DC contribution from a chroma component associated with the current block and an AC contribution from a luma component associated with the current block.
12. The method of claim 11, wherein the at least one chroma sample is classified into the first category based on coordinates associated with the at least one chroma sample.
13. The method of claim 12, wherein chroma samples corresponding to even rows or columns in the current block are classified into the first category and chroma samples corresponding to odd rows or columns in the current block are classified into a second category.
14. The method of claim 11, wherein the at least one chroma sample is classified into the first category based on values of corresponding reconstructed luma samples.
15. The method of claim 11, wherein parameters of one linear model from among the plural linear models are directly signaled in the coded video bitstream, and a difference between the directly signaled parameters of the one linear model and parameters of remaining linear models of the plural linear models are signaled in the coded video bitstream.
16. The method of claim 11, wherein one or more average luma values for a respective linear model of the plural linear models are computed by using samples belonging to a category corresponding to the respective linear model.
17. The method of claim 11, wherein a first flag in the coded video bitstream signals a cross-component mode enabling use of the plural linear models and the chroma-from-luma prediction mode and a second flag in the coded video bitstream signals whether a current mode corresponds to the use of the plural linear models and the chroma-from-luma prediction mode.
18. The method of claim 11, wherein one or more luma samples outside of boundaries of the current block are not used for calculating average luma values for the chroma-from-luma prediction mode.
19. The method of claim 11, wherein one or more average luma values are computed by using the values of luma samples corresponding to the at least one chroma sample in predefined positions when the chroma-from-luma prediction mode is used.
20. A method of processing visual media data, the method comprising: processing a bitstream that includes the visual media data according to a format rule, wherein the bitstream includes coded information for a current block in the bitstream, wherein the coded information indicates use of plural linear models in a chroma-from-luma prediction mode; the format rule specifies that at least one chroma sample of the current block is classified into a first category of plural categories based on at least one of (i) a position of the at least one chroma sample or (ii) values of luma samples corresponding to the at least one chroma sample; the format rule specifies that a first linear model of the plural linear models is selected according to the first category into which the at least one chroma sample is classified; and the format rule specifies that the at least one chroma sample of the current block is reconstructed using the chroma-from-luma prediction mode based on the selected first linear model, the at least one chroma sample being reconstructed from at least one luma sample based on a DC contribution from a chroma component associated with the current block and an AC contribution from a luma component associated with the current block.