

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7585500号
(P7585500)

(45)発行日 令和6年11月18日(2024.11.18)

(24)登録日 令和6年11月8日(2024.11.8)

(51)国際特許分類

F I

H O 4 N 19/132 (2014.01)

H O 4 N 19/132

請求項の数 70 (全114頁)

(21)出願番号	特願2023-536909(P2023-536909)	(73)特許権者	503433420
(86)(22)出願日	令和2年12月18日(2020.12.18)		華為技術有限公司
(65)公表番号	特表2024-500744(P2024-500744 A)		HUAWEI TECHNOLOGIES CO., LTD.
(43)公表日	令和6年1月10日(2024.1.10)		中華人民共和國 5 1 8 1 2 9 広東省深
(86)国際出願番号	PCT/EP2020/087333		▲チェン▼市龍崗区坂田 華為總部▲ベ
(87)国際公開番号	WO2022/128138		ン▼公樓
(87)国際公開日	令和4年6月23日(2022.6.23)		Huawei Administrat
審査請求日	令和5年6月16日(2023.6.16)		ion Building, Banti
			an, Longgang Distri
			ct, Shenzhen, Guang
			dong 5 1 8 1 2 9, P. R. C
			hina
		(74)代理人	100107766
			弁理士 伊東 忠重

最終頁に続く

(54)【発明の名称】 ニューラルネットワークによりピクチャを符号化又は復号する方法及び装置

(57)【特許請求の範囲】

【請求項1】

ニューラルネットワーク（NN）によりピクチャを符号化する方法であって、前記NNは、少なくとも1つの次元でサイズSを有するマトリクスを表す入力にダウンサンプリングを適用する少なくとも1つのダウンサンプリングレイヤを含む複数のレイヤを有する、前記方法において、

前記少なくとも1つのダウンサンプリングレイヤにより前記入力を処理する前に、前記入力にリスケーリングを適用することであり、前記リスケーリングは、

【数1】

$$\bar{S} = n \cdot r$$

なお、 $n \in \mathbb{N}$ 及び $r > 1, r \in \mathbb{N}$

であるように、前記少なくとも1つの次元での前記サイズSを変更することを含み、ここで、rは、前記ダウンサンプリングレイヤのダウンサンプリング比である、前記適用することと、

前記リスケーリングの後に、前記ダウンサンプリングレイヤによって前記入力を処理し、サイズ $S_1 < S$ を有する出力を供給することと、

前記NNにより前記ピクチャを処理した後、ビットストリームを前記NNの出力として供給することと

を有する方法。

【請求項2】

前記NNは、数

【数2】

$$M \in \mathbb{N}$$

10

個のダウンサンプリングレイヤを含み、

【数3】

$$m \leq M, m \in \mathbb{N}$$

として、各ダウンサンプリングレイヤ m は、関連するダウンサンプリング比

【数4】

20

$$r_m > 1, r_m \in \mathbb{N}$$

を有し、当該方法は、

少なくとも1つの次元でのサイズ S_m を有するマトリクスを表す入力をダウンサンプリングレイヤ m により処理する前に、前記サイズ S_m が前記ダウンサンプリングレイヤ m の前記ダウンサンプリング比 r_m の整数倍でない場合に、前記入力にリスケーリングを適用することであり、前記リスケーリングは、

【数5】

30

$$\overline{S_m} = n \cdot r_m, n \in \mathbb{N}$$

であるように、前記少なくとも1つの次元での前記サイズ S_m を変更することを含む、ことを更に有する、

請求項1に記載の方法。

【請求項3】

40

前記サイズ S_m を有する前記入力に前記リスケーリングを適用することの前に、 S_m が前記ダウンサンプリング比 r_m の整数倍であるかどうか決定され、 S_m が前記ダウンサンプリング比 r_m の整数倍でない場合に、前記リスケーリングが前記入力に適用され、それにより、前記サイズ S_m は、

【数6】

$$\overline{S_m} = n \cdot r_m, n \in \mathbb{N}$$

50

であるように、前記少なくとも 1 つの次元で変更される、
請求項 2 に記載の方法。

【請求項 4】

前記入力の前記サイズ S_m が前記ダウンサンプリングレイヤ m の前記ダウンサンプリング比 r_m の整数倍である場合に、サイズ

【数 7】

$$\overline{S_m} \neq S_m$$

10

へのリスケーリングは、前記ダウンサンプリングレイヤ m によって前記入力を処理する前に前記入力に適用されない、

請求項 3 に記載の方法。

【請求項 5】

S_m が前記ダウンサンプリング比 r_m の整数倍であるかどうかの前記決定は、前記サイズ S_m を前記ダウンサンプリングレイヤ m の許容入力サイズと比較することを含む、

請求項 3 又は 4 に記載の方法。

【請求項 6】

前記ダウンサンプリングレイヤ m の前記許容入力サイズは、前記ダウンサンプリング比 r_m 及び前記サイズ S_m のうちの少なくとも一方に基づき計算される、

20

請求項 5 に記載の方法。

【請求項 7】

前記比較することは、 S_m と前記ダウンサンプリングレイヤ m の前記許容入力サイズとの間の差を計算することを含む、

請求項 5 又は 6 に記載の方法。

【請求項 8】

前記比較することは、

【数 8】

$$\text{ceil}\left(\frac{S_m}{r_m}\right) \cdot r_m - S_m \text{ 又は } \text{floor}\left(\frac{S_m}{r_m}\right) \cdot r_m - S_m$$

30

を決定することを含む、

請求項 5 又は 7 に記載の方法。

【請求項 9】

【数 9】

$$\text{ceil}\left(\frac{S_m}{r_m}\right) \cdot r_m - S_m$$

40

が決定され、

【数 10】

$$\text{ceil}\left(\frac{S_m}{r_m}\right) \cdot r_m - S_m \neq 0$$

50

である場合に、前記リスケーリングは、前記サイズ S_m を有する前記入力に適用される、
請求項 8 に記載の方法。

【請求項 10】

【数 11】

$$\text{floor}\left(\frac{S_m}{r_m}\right) \cdot r_m - S_m$$

10

が決定され、

【数 12】

$$\text{floor}\left(\frac{S_m}{r_m}\right) \cdot r_m - S_m \neq 0$$

である場合に、前記リスケーリングは、前記サイズ S_m を有する前記入力に適用される、
請求項 8 に記載の方法。

【請求項 11】

前記サイズ

【数 13】

$$\overline{S_m}$$

20

は、前記ダウンサンプリング比 r_m 及び前記サイズ S_m のうちの少なくとも一方を用いて
決定される、

請求項 10 に記載の方法。

【請求項 12】

前記サイズ

【数 14】

$$\overline{S_m}$$

30

は、前記サイズ S_m を、前記ダウンサンプリング比 r_m の最も近い小さい方の整数倍に又
は前記ダウンサンプリング比 r_m の最も近い大きい方の整数倍に変更することによって、
決定される、

請求項 11 に記載の方法。

【請求項 13】

－ 前記サイズ

【数 15】

$$\overline{S_m}$$

40

は、

50

【数 1 6】

$$\text{floor}\left(\frac{S_m}{r_m}\right) \cdot r_m = \overline{S_m}$$

を用いて決定されるか、あるいは、

－ 前記サイズ

【数 1 7】

$$\overline{S_m}$$

10

は、

【数 1 8】

$$\text{ceil}\left(\frac{S_m}{r_m}\right) \cdot r_m = \overline{S_m}$$

20

を用いて決定される、

請求項 1 1 に記載の方法。

【請求項 1 4】

ダウンサンプリングレイヤ m の入力に適用される前記リスケーリングは、他のダウンサンプリングレイヤのダウンサンプリング比 r_k , $k \neq m$ と無関係である、

請求項 2 乃至 1 3 のうちいずれか一項に記載の方法。

【請求項 1 5】

ダウンサンプリングレイヤ m への前記入力は、前記ダウンサンプリングレイヤ m の前記ダウンサンプリング比であるダウンサンプリング比 r_m の最も近い小さい方の整数倍と、前記ダウンサンプリングレイヤ m の前記ダウンサンプリング比 r_m の最も近い大きい方の

30

整数倍との間にある値を有する前記少なくとも 1 つの次元でのサイズ S_m を有し、
条件に応じて、前記入力の前記サイズ S_m は、前記ダウンサンプリング比 r_m の前記最も近い小さい方の整数倍に一致するか又は前記ダウンサンプリング比 r_m の前記最も近い大きい方の整数倍に一致するかのどちらかであるよう前記リスケーリング中に変更される、

請求項 1 乃至 1 4 のうちいずれか一項に記載の方法。

【請求項 1 6】

ダウンサンプリングレイヤ m への前記入力は、前記ダウンサンプリングレイヤ m の前記ダウンサンプリング比であるダウンサンプリング比 r_m の整数倍でない値を有する前記少なくとも 1 つの次元でのサイズ S_m を有し、

前記入力の前記サイズ S_m は、前記ダウンサンプリング比 r_m の最も近い小さい方の整数倍に一致するか又は前記ダウンサンプリング比 r_m の最も近い大きい方の整数倍に一致するかのどちらかであるよう前記リスケーリング中に変更される、

40

請求項 1 乃至 1 4 のうちいずれか一項に記載の方法。

【請求項 1 7】

ダウンサンプリングレイヤ m への前記入力は、前記少なくとも 1 つの次元でのサイズ S_m を有し、ここで、

【数 2 1】

$$lr_m \leq S_m \leq r_m(l+1), l, m \in \mathbb{N}$$

50

であり、 r_m は、前記ダウンサンプリングレイヤ m の前記ダウンサンプリング比であり、
サイズ S_m は、条件に応じて、

【数 2 2】

$$\overline{S_m} = lr_m \text{ 又は } \overline{S_m} = r_m(l+1)$$

のどちらかにリスケーリングされる、

請求項 1 乃至 1 4 のうちいずれか一項に記載の方法。

10

【請求項 1 8】

ダウンサンプリングレイヤ m への前記入力は、前記少なくとも 1 つの次元でのサイズ S_m を有し、前記入力の前記サイズ S_m が、前記ダウンサンプリングレイヤ m の前記ダウンサンプリング比であるダウンサンプリング比 r_m の最も近い大きい方の整数倍よりも前記ダウンサンプリングレイヤ m の前記ダウンサンプリング比 r_m の最も近い小さい方の整数倍に近い場合に、前記入力の前記サイズ S_m は、前記ダウンサンプリング比 r_m の前記最も近い小さい方の整数倍に一致するサイズ

【数 2 3】

$$\overline{S_m}$$

20

に低減される、

請求項 1 乃至 1 4 のうちいずれか一項に記載の方法。

【請求項 1 9】

前記入力の前記サイズ S_m を前記サイズ

【数 2 4】

$$\overline{S_m}$$

30

に低減させることは、前記入力をクロッピングすることを含む、

請求項 1 8 に記載の方法。

【請求項 2 0】

ダウンサンプリングレイヤ m への前記入力は、前記少なくとも 1 つの次元でのサイズ S_m を有し、前記入力の前記サイズ S_m が、前記ダウンサンプリングレイヤ m の前記ダウンサンプリング比であるダウンサンプリング比 r_m の最も近い小さい方の整数倍よりも前記ダウンサンプリングレイヤ m の前記ダウンサンプリング比 r_m の最も近い大きい方の整数倍に近い場合に、前記入力の前記サイズ S_m は、前記ダウンサンプリング比 r_m の前記最も近い大きい方の整数倍に一致するサイズ

【数 2 5】

$$\overline{S_m}$$

40

に増大される、

請求項 1 乃至 1 4 のうちいずれか一項に記載の方法。

【請求項 2 1】

前記入力の前記サイズ S_m を前記サイズ

50

【数 2 6】

$$\overline{S_m}$$

に増大させることは、前記サイズ S_m を有する前記入力にゼロを又は前記サイズ S_m を有する前記入力から得られたパディング情報をパディングすることを含む、

請求項 20 に記載の方法。

【請求項 22】

前記サイズ S_m を有する前記入力から得られた前記パディング情報は、前記入力の前記サイズ S_m を前記サイズ

10

【数 2 7】

$$\overline{S_m}$$

に増大させるための冗長なパディング情報として適用される、

請求項 21 に記載の方法。

【請求項 23】

冗長なパディング情報による前記パディングすることは、反射パディング又は繰り返しパディングのうちの少なくとも一方を含む、

20

請求項 22 に記載の方法。

【請求項 24】

前記パディング情報は、前記冗長なパディング情報が加えられるべき前記入力内の領域に最も近い、前記サイズ S_m を有する前記入力の少なくとも 1 つの値であるか又は該少なくとも 1 つの値を含む、

請求項 22 又は 23 に記載の方法。

【請求項 25】

ダウンサンプリングレイヤ m への前記入力のサイズ S_m は、前記ダウンサンプリングレイヤ m の前記ダウンサンプリング比であるダウンサンプリング比 r_m の最も近い大きい方の整数倍に一致するサイズ

30

【数 2 8】

$$\overline{S_m}$$

に増大される、

請求項 1 又は 13 に記載の方法。

【請求項 26】

前記条件は、 $\text{Min}(|S - l r_m|, |S - r_m(l + 1)|)$ を使用し、前記条件は、 Min が $|S_m - l r_m|$ を選択する場合に、前記入力の前記サイズ S_m は

40

【数 2 9】

$$\overline{S_m} = l r_m$$

に低減され、 Min が $|S_m - (l + 1) r_m|$ を選択する場合に、前記入力の前記サイズ S_m は、

【数 30】

50

$$\overline{S_m} = (l + 1)r_m$$

に増大されることを含む、
請求項 17 に記載の方法。

【請求項 27】

l は、前記ダウンサンプリングレイヤ m への前記入力の前記サイズ S_m 及び前記ダウンサンプリングレイヤ m の前記ダウンサンプリング比であるダウンサンプリング比 r_m のうちの少なくとも一方を用いて決定される、

10

請求項 26 に記載の方法。

【請求項 28】

l は、

【数 31】

$$l = \text{floor} \left(\frac{S_m}{r_m} \right)$$

20

によって決定され、かつ／あるいは、

$l + 1$ は、

【数 32】

$$l + 1 = \text{ceil} \left(\frac{S_m}{r_m} \right)$$

30

によって決定される、

請求項 27 に記載の方法。

【請求項 29】

前記ダウンサンプリングレイヤの少なくとも 1 つは、第 1 次元及び第 2 次元で前記入力にダウンサンプリングを適用し、前記第 1 次元でのダウンサンプリング比は、前記第 2 次元でのダウンサンプリング比に等しい、

請求項 1 乃至 28 のうちいずれか一項に記載の方法。

【請求項 30】

全てのダウンサンプリングレイヤの前記ダウンサンプリング比は等しい、

請求項 1 乃至 29 のうちいずれか一項に記載の方法。

40

【請求項 31】

ニューラルネットワーク (NN) により、ピクチャを表すビットストリームを復号する方法であって、前記 NN は、少なくとも 1 つの次元でサイズ T を有するマトリクスを表す入力にアップサンプリングを適用する少なくとも第 1 アップサンプリングレイヤ及び第 2 アップサンプリングレイヤを含む複数のレイヤを有する、前記方法において、

前記第 1 アップサンプリングレイヤによって前記入力を処理し、前記第 1 アップサンプリングレイヤの出力を供給することであり、前記出力は、サイズ

【数 33】

50

$$\bar{T}_1 = T \cdot u_1, u_1 > 1, u_1 \in \mathbb{N}$$

を有し、 u_1 は、前記第1アップサンプリングレイヤのアップサンプリング比である、ことと、

前記第2アップサンプリングレイヤによって前記第1アップサンプリングレイヤの前記出力を処理する前に、前記第1アップサンプリングレイヤの前記出力にリスケージングを適用することであり、前記リスケージングは、前記少なくとも1つの次元での前記出力の前記サイズ

10

【数34】

$$\bar{T}_1$$

を、取得された情報に基づいて前記少なくとも1つの次元でのサイズ

【数35】

$$\hat{T}_1$$

20

に変更することを含む、ことと、

前記リスケージングされた出力を前記第2アップサンプリングレイヤによって処理し、前記第2アップサンプリングレイヤの出力を供給することであり、前記出力は、前記少なくとも1つの次元でサイズ

【数36】

$$\bar{T}_2 = \hat{T}_1 \cdot u_2, u_2 > 1, u_2 \in \mathbb{N}$$

30

を有し、 u_2 は、前記第2アップサンプリングレイヤのアップサンプリング比である、ことと、

前記NNを用いて前記ビットストリームを処理した後、復号されたピクチャを前記NNの出力として供給することと

を有する方法。

【請求項32】

前記第1アップサンプリングレイヤ及び前記第2アップサンプリングレイヤの夫々は、転置畳み込み又は畳み込みレイヤを有する、

請求項31に記載の方法。

40

【請求項33】

前記情報は、前記復号されたピクチャの高さH及び前記復号されたピクチャの幅Wのうちの少なくとも一方を含む前記復号されたピクチャの目標サイズ、前記アップサンプリング比 u_1 、前記アップサンプリング比 u_2 、前記第2アップサンプリングレイヤの目標出力サイズ

【数37】

$$\bar{T}_2$$

50

、前記サイズ

【数 3 8】

$$\hat{T}_1$$

のうちの少なくとも 1 つを含む、

請求項 3 1 又は 3 2 に記載の方法。

【請求項 3 4】

前記情報は、前記ビットストリーム、第 2 ビットストリーム、デコーダで利用可能な情報、のうちの少なくとも 1 つから取得される、

10

請求項 3 1 乃至 3 3 のうちいずれか一項に記載の方法。

【請求項 3 5】

前記少なくとも 1 つの次元での前記サイズ

【数 3 9】

$$\bar{T}_1$$

をサイズ

20

【数 4 0】

$$\hat{T}_1$$

に変更する前記リスケーリングは、

【数 4 1】

$$\bar{T}_2 \text{ 及び } u_2$$

30

に依存する式に基づいて決定され、

【数 4 2】

$$\bar{T}_2$$

は、前記第 2 アップサンプリングレイヤの前記出力の目標出力サイズであり、 u_2 は、前記第 2 アップサンプリングレイヤの前記アップサンプリング比である、

40

請求項 3 1 乃至 3 4 のうちいずれか一項に記載の方法。

【請求項 3 6】

前記少なくとも 1 つの次元での前記サイズ

【数 4 3】

$$\bar{T}_1$$

をサイズ

50

【数 4 4】

$$\hat{T}_1$$

に変更する前記リスケーリングは、 u_2 及び N に依存する式に基づいて決定され、 N は、前記 NN による前記ビットストリームの処理順序において前記第 1 アップサンプリングレイヤに続くアップサンプリングレイヤの総数である、

請求項 3 1 乃至 3 5 のうちいずれか一項に記載の方法。

【請求項 3 7】

前記式は、

【数 4 5】

$$\hat{T}_1 = \text{ceil} \left(\frac{T_{\text{output}}}{u_2^N} \right)$$

によって与えられ、 T_{output} は、前記 NN の前記出力の目標サイズである、

請求項 3 6 に記載の方法。

【請求項 3 8】

前記式は、

【数 4 6】

$$\hat{T}_1 = \text{ceil} \left(\frac{T_{\text{output}}}{u^N} \right)$$

によって与えられ、 T_{output} は、前記 NN の前記出力の前記目標サイズであり、 u はアップサンプリング比である、

請求項 3 7 に記載の方法。

【請求項 3 9】

前記目標サイズを示す指示は、前記ビットストリームに含まれる、

請求項 3 8 に記載の方法。

【請求項 4 0】

前記式は、

【数 4 7】

$$\hat{T}_1 = \text{ceil} \left(\frac{\bar{T}_2}{u_2} \right)$$

によって与えられるか、あるいは、前記式は、

【数 4 8】

$$\hat{T}_1 = \text{floor} \left(\frac{\bar{T}_2}{u_2} \right)$$

によって与えられる、

請求項 3 5 に記載の方法。

10

20

30

40

50

【請求項 4 1】

複数の事前定義された式のうちのどれが選択されるかを示す指示は、前記ビットストリームに含まれる、

請求項 4 0 に記載の方法。

【請求項 4 2】

当該方法は、前記サイズ

【数 4 9】

$$\overline{T}_1$$

10

を有する前記出力をリスケーリングすることの前に、前記出力の前記サイズ

【数 5 0】

$$\overline{T}_1$$

が前記サイズ

【数 5 1】

20

$$\hat{T}_1$$

と一致するかどうかを決定することを更に有する、

請求項 3 1 乃至 4 1 のうちいずれか一項に記載の方法。

【請求項 4 3】

前記サイズ

【数 5 2】

30

$$\overline{T}_1$$

が前記サイズ

【数 5 3】

$$\hat{T}_1$$

と一致すると決定される場合に、前記サイズ

【数 5 4】

40

$$\overline{T}_1$$

を変更するリスケーリングは適用されない、

請求項 4 2 に記載の方法。

【請求項 4 4】

当該方法は、前記サイズ

【数 5 5】

50

$$\overline{T_1}$$

が前記サイズ
【数 5 6】

$$\widehat{T_1}$$

よりも大きいかどうか又は前記サイズ
【数 5 7】

$$\overline{T_1}$$

が前記サイズ
【数 5 8】

$$\widehat{T_1}$$

よりも小さいかどうかを決定することを更に有する、
請求項 4 2 又は 4 3 に記載の方法。
【請求項 4 5】
前記サイズ
【数 5 9】

$$\overline{T_1}$$

が前記サイズ
【数 6 0】

$$\widehat{T_1}$$

よりも大きいと決定される場合に、前記リスケーリングは、前記サイズ
【数 6 1】

$$\overline{T_1}$$

を有する前記出力にクロッピングを適用して、前記サイズ
【数 6 2】

$$\overline{T_1}$$

が前記サイズ

【数 6 3】

$$\widehat{T}_1$$

に低減されるようにする、
請求項 4 4 に記載の方法。

【請求項 4 6】

前記サイズ

【数 6 4】

10

$$\overline{T}_1$$

が前記サイズ

【数 6 5】

$$\widehat{T}_1$$

20

よりも小さいと決定される場合に、前記リスケーリングは、前記サイズ

【数 6 6】

$$\overline{T}_1$$

を有する前記出力にパディングを適用して、前記サイズ

【数 6 7】

$$\overline{T}_1$$

30

が前記サイズ

【数 6 8】

$$\widehat{T}_1$$

に増大されるようにする、
請求項 4 4 又は 4 5 に記載の方法。

40

【請求項 4 7】

前記パディングは、前記サイズ

【数 6 9】

$$\overline{T}_1$$

を有する前記出力にゼロを又は前記サイズ

【数 7 0】

50

$\overline{T}_1$

を有する前記出力から得られたパディング情報をパディングすることを有する、
請求項 46 に記載の方法。

【請求項 48】

前記サイズ

【数 71】

10

 $\overline{T}_1$

を有する前記出力から得られた前記パディング情報は、前記出力の前記サイズ
【数 72】

 $\overline{T}_1$

を前記サイズ

【数 73】

20

 $\widehat{T}_1$

に増大させる冗長なパディング情報として適用される、
請求項 47 に記載の方法。

【請求項 49】

前記パディングは、反射パディング又は繰り返しパディングを有する、
請求項 48 に記載の方法。

30

【請求項 50】

前記パディング情報は、前記冗長なパディング情報が加えられるべき前記出力内の領域
に最も近い、前記サイズ

【数 74】

 $\overline{T}_1$

を有する前記出力の少なくとも 1 つの値であるか又は該少なくとも 1 つの値を含む、
請求項 48 又は 49 に記載の方法。

40

【請求項 51】

【数 75】

 $\overline{T}_1$

が

【数 76】

50

$\hat{T}_1$

に等しくないと決定される場合に、前記リスケーリングは、補間フィルタを適用することを含む、

請求項 4 4 に記載の方法。

【請求項 5 2】

前記情報は、前記ビットストリーム又は更なるビットストリームで提供され、前記ビットストリームを符号化したエンコーダの1つ以上のダウンサンプリングレイヤのダウンサンプリング比 r_m を含む、

10

前記ダウンサンプリングレイヤの夫々は、前記入力を処理する順序において、デコーダの前記第 1 アップサンプリングレイヤ及び前記第 2 アップサンプリングレイヤを含むアップサンプリングレイヤのうちの各 1 つに対応する、

請求項 3 1 乃至 5 1 のうちいずれか一項に記載の方法。

【請求項 5 3】

前記第 1 アップサンプリングレイヤ及び前記第 2 アップサンプリングレイヤを含むアップサンプリングレイヤのうちの少なくとも 1 つは、第 1 次元及び第 2 次元でアップサンプリングを適用し、前記第 1 次元でのアップサンプリング比は、前記第 2 次元でのアップサンプリング比に等しい、

20

請求項 3 1 乃至 5 2 のうちいずれか一項に記載の方法。

【請求項 5 4】

全てのアップサンプリングレイヤの前記アップサンプリング比は等しい、

請求項 3 1 乃至 5 3 のうちいずれか一項に記載の方法。

【請求項 5 5】

前記 NN は、前記 NN による前記ビットストリームの処理順序において、前記少なくとも 1 つの次元での前記入力のサイズを変更しない前記入力への変換を適用する更なるユニットを含み、

当該方法は、前記更なるユニットによる前記入力の処理の後、後続のレイヤによって前記入力を処理する前に、前記リスケーリングが前記少なくとも 1 つの次元での前記入力のサイズの増大をもたらす場合に、前記リスケーリングを適用することを有し、かつ／あるいは、当該方法は、前記更なるユニットによる前記入力の処理の前に、前記リスケーリングが前記少なくとも 1 つの次元での前記入力のサイズの低減を含む場合に、前記リスケーリングを適用することを有する、

30

請求項 1 乃至 3 0 のうちいずれか一項に記載の方法。

【請求項 5 6】

前記更なるユニットは、バッチ正規化部及び／又は正規化線形ユニット（ReLU）であるか又はそれを有する、

請求項 5 5 に記載の方法。

【請求項 5 7】

前記ビットストリームは、前記ピクチャの異なるカラーチャネルに対応するサブビットストリームを有し、

40

前記 NN は、サブニューラルネットワーク（sNN）を有し、該 sNN は夫々、請求項 1 乃至 5 1 のうちいずれか一項に記載の方法を、前記 sNN への入力として供給された前記サブビットストリームに適用するよう構成される、

請求項 1 乃至 3 0 又は 5 5 乃至 5 6 のうちいずれか一項に記載の方法。

【請求項 5 8】

前記 NN は、前記 NN による前記ビットストリームの処理順序において、前記少なくとも 1 つの次元での前記入力のサイズを変更しない前記入力への変換を適用する更なるユニットを含み、

50

当該方法は、前記更なるユニットによる前記入力の処理の後、後続のレイヤによって前記入力を処理する前に、前記リスケーリングが前記少なくとも1つの次元での前記入力のサイズの増大をもたらす場合に、前記リスケーリングを適用することを有し、かつ／あるいは、当該方法は、前記更なるユニットによる前記入力の処理の前に、前記リスケーリングが前記少なくとも1つの次元での前記入力のサイズの低減を含む場合に、前記リスケーリングを適用することを有する、

請求項31乃至54のうちいずれか一項に記載の方法。

【請求項59】

前記更なるユニットは、バッチ正規化部及び／又は正規化線形ユニット（ReLU）であるか又はそれを有する、

10

請求項58に記載の方法。

【請求項60】

前記ビットストリームは、前記ピクチャの異なるカラーチャネルに対応するサブビットストリームを有し、

前記NNは、サブニューラルネットワーク（sNN）を有し、該sNNは夫々、請求項1乃至51のうちいずれか一項に記載の方法を、前記sNNへの入力として供給された前記サブビットストリームに適用するよう構成される、

請求項31乃至54又は58乃至59のうちいずれか一項に記載の方法。

【請求項61】

前記リスケーリングが前記サイズSをサイズ

20

【数77】

$$\bar{S}$$

に増大させることを含む場合に、前記サイズ

【数78】

$$\bar{S}$$

30

は、

【数79】

$$\bar{S} = \text{Int} \left(\frac{S + r - 1}{r} \right) r$$

によって与えられ、

40

前記リスケーリングが前記サイズSを前記サイズ

【数80】

$$\bar{S}$$

に低減させることを含む場合に、前記サイズ

【数81】

50

$$\bar{S}$$

は、

【数 8 2】

$$\bar{S} = \text{Int} \left(\frac{S}{r} \right) r$$

10

によって与えられる、

請求項 1 乃至 30 のうちいずれか一項に記載の方法。

【請求項 6 2】

前記リスケーリングが前記サイズ

【数 8 3】

$$\bar{T}_1$$

20

を前記サイズ

【数 8 4】

$$\hat{T}_1$$

に増大させることを含む場合に、前記サイズ

【数 8 5】

$$\hat{T}_1$$

30

は、

【数 8 6】

$$\hat{T}_1 = \text{Int} \left(\frac{\bar{T}_1 + u_2 - 1}{u_2} \right) u_2$$

40

によって与えられ、

前記リスケーリングが前記サイズ

【数 8 7】

$$\bar{T}_1$$

を前記サイズ

【数 8 8】

50

$$\hat{T}_1$$

に低減させることを含む場合に、前記サイズ
【数 8 9】

$$\hat{T}_1$$

は、
【数 9 0】

$$\hat{T}_1 = \text{Int} \left(\frac{\bar{T}_1}{u_2} \right) u_2$$

によって与えられる、

請求項 3 1 乃至 5 4 のうちいずれか一項に記載の方法。

【請求項 6 3】

ピクチャを符号化するエンコーダであって、

前記ピクチャを受信する受信器と、

ニューラルネットワーク（NN）を実装するよう構成され、前記 NN は、前記 NN による前記ピクチャの処理順序において、入力にダウンサンプリングを適用するよう構成される少なくとも 1 つのダウンサンプリングレイヤを含む複数のレイヤを有する、1 つ以上のプロセッサと、

ビットストリームを出力する送信器と

を有し、

当該エンコーダは、請求項 1 乃至 3 0 又は 5 5 乃至 5 7 又は 6 1 のうちいずれか一項に記載の方法を実行するよう構成される、

エンコーダ。

【請求項 6 4】

ピクチャを符号化するエンコーダであって、

ニューラルネットワーク（NN）を実装する 1 つ以上のプロセッサを有し、

前記 1 つ以上のプロセッサは、請求項 1 乃至 3 0 又は 5 5 乃至 5 7 又は 6 1 のうちいずれか一項に記載の方法を実行するよう構成される、

エンコーダ。

【請求項 6 5】

ニューラルネットワーク（NN）によりピクチャを符号化するエンコーダであって、

前記 NN を実装する 1 つ以上のプロセッサを有し、前記 NN は、少なくとも 1 つの次元でサイズ S を有するマトリクスを表す入力にダウンサンプリングを適用する少なくとも 1 つのダウンサンプリングレイヤを含む複数のレイヤを有し、

前記 1 つ以上のプロセッサは、

前記少なくとも 1 つのダウンサンプリングレイヤにより前記入力を処理する前に、前記入力にリスケーリングを適用することであり、前記リスケーリングは、

【数 9 1】

10

20

30

40

50

$$\bar{S} = n \cdot r$$

なお、 $n \in \mathbb{N}$ 及び $r > 1, r \in \mathbb{N}$

であるように、前記少なくとも1つの次元での前記サイズ S を変更することを含み、ここで、 r は、前記レイヤのダウンサンプリング比である、前記適用することと、

前記リスケーリングの後に、前記ダウンサンプリングレイヤによって前記入力を処理し、サイズ $S_1 < S$ を有する出力を供給することと、

前記 NN により前記ピクチャを処理した後、ビットストリームを前記 NN の出力として供給することと

を実行するよう構成される、

エンコーダ。

【請求項 66】

ピクチャを表すビットストリームを復号するデコーダであって、

前記ビットストリームを受信する受信器と、

ニューラルネットワーク (NN) を実装するよう構成され、前記 NN は、前記 NN による前記ビットストリームの処理順序において、入力にアップサンプリングを適用するよう構成される少なくとも2つのアップサンプリングレイヤを含む複数のレイヤを有する、1つ以上のプロセッサと、

復号されたピクチャを出力する送信器と

を有し、

当該デコーダは、請求項 31 乃至 54 又は 58 乃至 60 又は 62 のうちいずれか一項に記載の方法を実行するよう構成される、

デコーダ。

【請求項 67】

ピクチャを表すビットストリームを復号するデコーダであって、

ニューラルネットワーク (NN) を実装する1つ以上のプロセッサを有し、

前記1つ以上のプロセッサは、請求項 31 乃至 54 又は 58 乃至 60 又は 62 のうちいずれか一項に記載の方法を実行するよう構成される、

デコーダ。

【請求項 68】

ピクチャを表すビットストリームを復号するデコーダであって、

ニューラルネットワーク (NN) を実装する1つ以上のプロセッサを有し、前記 NN は、少なくとも1つの次元でサイズ T を有するマトリクスを表す入力にアップサンプリングを適用する少なくとも第1アップサンプリングレイヤ及び第2アップサンプリングレイヤを含む複数のレイヤを有し、

前記1つ以上のプロセッサは、

前記第1アップサンプリングレイヤによって前記入力を処理し、前記第1アップサンプリングレイヤの出力を供給することであり、前記出力は、サイズ

【数 92】

$$\bar{T}_1 = T \cdot u_1, u_1 > 1, u_1 \in \mathbb{N}$$

を有し、 u_1 は、前記第1アップサンプリングレイヤのアップサンプリング比である、ことと、

前記第2アップサンプリングレイヤによって前記第1アップサンプリングレイヤの前記出力を処理する前に、前記第1アップサンプリングレイヤの前記出力にリスケーリングを

適用することであり、前記リスケーリングは、前記少なくとも 1 つの次元での前記出力の前記サイズ

【数 9 3】

$$\overline{T}_1$$

を、取得された情報に基づいて前記少なくとも 1 つの次元でのサイズ

【数 9 4】

$$\widehat{T}_1$$

に変更することを含む、ことと、

前記リスケーリングされた出力を前記第 2 アップサンプリングレイヤによって処理し、前記第 2 アップサンプリングレイヤの出力を供給することであり、前記出力は、前記少なくとも 1 つの次元でサイズ

【数 9 5】

$$\overline{T}_2 = \widehat{T}_1 \cdot u_2, u_2 > 1, u_2 \in \mathbb{N}$$

を有し、 u_2 は、前記第 2 アップサンプリングレイヤのアップサンプリング比である、ことと、

前記 NN を用いて前記ビットストリームを処理した後、復号されたピクチャを前記 NN の出力として供給することと

を実行するよう構成される、

デコーダ。

【請求項 6 9】

コンピューティングシステムで実行される場合に、前記コンピューティングシステムに、請求項 1 乃至 3 0 又は 5 5 乃至 5 7 又は 6 1 のうちいずれか一項に記載の方法を実行させるコンピュータ実行可能命令を有するコンピュータ可読記憶媒体。

【請求項 7 0】

コンピューティングシステムで実行される場合に、前記コンピューティングシステムに、請求項 3 1 乃至 5 4 又は 5 8 乃至 6 0 又は 6 2 のうちいずれか一項に記載の方法を実行させるコンピュータ実行可能命令を有するコンピュータ可読記憶媒体。

【発明の詳細な説明】

【技術分野】

【0 0 0 1】

本願は、ニューラルネットワークによりピクチャを符号化する方法及びニューラルネットワークによりピクチャを復号する方法に関する。本願は更に、ピクチャを符号化するエンコーダ及びピクチャを復号するデコーダ、並びにコンピュータ可読記憶媒体に関する。

【背景技術】

【0 0 0 2】

ビデオコーディング（ビデオ符号化及び復号化）は、広範囲のデジタルビデオアプリケーション、例えば、デジタル TV 放送、インターネット及び移動体ネットワークを介したビデオ伝送、ビデオチャットなどのリアルタイムの会話アプリケーション、ビデオ会議、

10

20

30

40

50

DVD及びBlu-rayディスク、ビデオコンテンツ取得及び編集システム、並びにセキュリティアプリケーションのカムコードにおいて使用されている。

【0003】

比較的短いビデオでさえ描画するのに必要なビデオデータの量はかなりの量である可能性があり、その結果、データが有限な帯域幅容量で通信ネットワークにわたってストリーミングされるか又は別なふうに通信される場合に問題が生じる。よって、ビデオデータは、今日のテレコミュニケーションズネットワークにわたって通信される前に、一般的には圧縮される。メモリ資源が有限である可能性があるために、ビデオが記憶デバイスに記憶される場合に、ビデオのサイズも問題になることがある。ビデオ圧縮デバイスは、伝送又は記憶の前にビデオデータをコーディングするようソース側でソフトウェア及び／又はハードウェアをしばしば使用して、デジタルビデオ画像を表すのに必要なデータの量を減らす。圧縮されたデータは、次いで、ビデオデータを復号するビデオ圧縮解除デバイスによってあて先側で受信される。ネットワークリソースが限られておりかつより高いビデオ品質要求が更に高まるにつれて、ピクチャ品質をほとんど又は全く犠牲にせずに圧縮比を向上させる改善された圧縮及び圧縮解除技術が望ましい。

10

【0004】

ニューラルネットワーク及びニューラルネットワークを使用するディープラーニング技術が、現在、ビデオ、画像などの符号化及び復号化の技術分野でも、かねてより使用されてきた。

【0005】

20

かような場合に、ビットストリームは通常、値の2次元行列によって合理的に表現されるデータを表すか、又はデータそのものである。例えば、これは、画像、ビデオシーケンス、又は同様のデータを表すか又はそのようなものであるビットストリームに当てはまる。2Dデータは別として、本開示で言及されているニューラルネットワーク及びフレームワークは、通常は1D信号として表されるオーディオ信号、又は他の信号などの更なるソース信号に適用されてもよい。

【0006】

例えば、複数のダウンサンプリングレイヤを有するニューラルネットワークは、ピクチャのような符号化されるべき入力にダウンサンプリング（ダウンサンプリングレイヤが畳み込みレイヤである場合に、畳み込み）を適用し得る。このダウンサンプリングを入力ピクチャに適用することによってサイズが低減され、これは、最終的なサイズが得られるまで繰り返され得る。そのようなニューラルネットワークは、ディープラーニングニューラルネットワークによる画像認識及びピクチャの符号化の両方に使用され得る。相応して、そのようなネットワークは、符号化されたピクチャを復号するために使用され得る。2つよりも少ない又は多い次元を有する信号などの他のソース信号も、同様のネットワークによって処理され得る。

30

【0007】

サイズが異なる可能性がある様々な異なる信号に効率的に適用され得るニューラルネットワークフレームワークを提供することが望ましい場合がある。

【発明の概要】

40

【0008】

本開示のいくつかの実施形態は、情報を運ぶビットストリームのサイズを低減することを可能にしながら、情報の損失を可能な限り最小限に抑えて元の情報が再構成され得ることを依然として保証し得る。

【0009】

いくつかの実施形態は、独立請求項1に係る、ニューラルネットワークによりピクチャを符号化する方法、及び請求項31に係る、ニューラルネットワークによりビットストリームを復号する方法と、請求項60又は61又は62に係る、ピクチャを符号化するエンコーダ、及び請求項63又は64又は65に係る、ビットストリームを復号するデコーダと、請求項66に係るコンピュータ可読記憶媒体とを提供する。更なる好ましい実施形態

50

は、従属請求項で提供される。

【0010】

ピクチャを符号化する方法は、ニューラルネットワークNNを使用して提供され、NNは、少なくとも1つの次元でサイズSを有するマトリクスを表す入力にダウンサンプリングを適用する少なくとも1つのダウンサンプリングレイヤを含む複数のレイヤを有し、方法は、少なくとも1つのダウンサンプリングレイヤにより入力を処理する前に、入力にリスケールリングを適用することであり、リスケールリングは、

【数1】

$$\bar{S} = n \cdot r$$

なお、 $n \in \mathbb{N}$ 及び $r > 1, r \in \mathbb{N}$

10

であるように、少なくとも1つの次元でのサイズSを変更することを含み、ここで、rは、ダウンサンプリングレイヤのダウンサンプリング比である、ことと、

リスケールリングの後に、ダウンサンプリングレイヤによって入力を処理し、サイズ $S_1 < S$ を有する出力を供給することと、

NNによりピクチャを処理した後（例えば、NNの各レイヤによりピクチャを処理した後）、ビットストリームをNNの出力として供給することと

を有する。

20

【0011】

本開示の文脈中、ピクチャは、ビデオ又はビデオシーケンスという意味で静止画又は動画として理解され得る。更に、本開示でのピクチャは、より大きい（全体の）ピクチャのセクションのみに関連してもよい。更に、ピクチャは画像又はフレームとも呼ばれ得る。いずれの場合にも、ピクチャは、マトリクスの形で値（通常、サンプルと呼ばれる）の2又はそれ以上の次元の配列であると又はそのようなものによって表現可能であると見なされてよく、それは、次いでニューラルネットワークによって、具体的には、上述されたような方法でダウンサンプリングレイヤによって処理され得る。

【0012】

本開示に従うビットストリームは、符号化されたピクチャであっても又はそれを有してもよい。ビットストリームは、例えば、ビットストリームを復号してピクチャを再構成するのに必要な情報に関連する追加情報（サイド情報）を有することができる。

30

【0013】

これに関連して、ビットストリームは、マトリクスとしてビットストリームの表現で見られる場合に、少なくとも1つの次元で元々も入力されたピクチャのサイズよりも小さいサイズを有してもよい。ここでのサイズという用語は、例えば、1つ以上の次元（例えば、幅及び高さ）におけるサンプルの数を指す。サイズはまた、ピクチャ又はピクチャエリアごとのサンプルの数によって通常特徴付けられる解像度表すこともある。

【0014】

本開示は、入力ピクチャの次元の全てがエンコーダの出力で低減される場合に制限されない、ことが留意されるべきである。例えば、入力ピクチャは、その次元のただ1つでのサイズにおいて低減され得る。

40

【0015】

更に、ビットストリームは、例えば、第3の次元を有しもよく、第3の次元は、入力ピクチャの対応する次元よりも大きいサイズを有してもよい。第3の次元は、チャネルとも呼ばれ得る特徴マップの数を表し得る。1つの具体例では、第3の次元のサイズは、入力（ピクチャ）では3であり、出力（ビットストリーム）では192であってよい。特徴マップのサイズは、通常は、入力をそれらの重要度に応じてより効率的に分類するために、エンコーダによって増大される。

50

【0016】

ダウンサンプリングは、任意の既知の又は技術的に妥当な方法で達成されてよい。それは、例えば、各々のダウンサンプリングレイヤの入力に畳み込みを適用することによるダウンサンプリングを有してもよい。それに関連して、ダウンサンプリングレイヤは、具体的に、畳み込みを入力に適用する畳み込みレイヤとして実装されてよい。

【0017】

ダウンサンプリングは、マトリクスの形で表される場合に、入力ピクチャのただ1つの次元でしか実行されなくてもよい。それはまた、マトリクスの1つよりも多い又は全ての次元に適用されることも可能である。しかし、本開示は、如何なる特定のダウンサンプリングにも限定されない。処理レイヤの1つ以上は、例えば、入力ピクチャ又は入力特徴マップの2番目、3番目、などの行及び／又は列ごとに削除（除外）することによって、畳み込み以外のダウンサンプリングを適用してもよい。

【0018】

ここで提供される本開示実施形態は、各々のダウンサンプリングレイヤによる入力の処理の直前にリスケーリングが適用されるように理解されるべきである。ニューラルネットワークに1つよりも多いダウンサンプリングレイヤがある場合に、リスケーリングは、各々のダウンサンプリングレイヤの夫々の前に、しかし、前のレイヤによる入力の処理の後に適用されることになる。スケーリングは、このようにして、上記に沿って、各々のダウンサンプリングレイヤのダウンサンプリング比に依存することになる。

【0019】

リスケーリングは、いくつかの実施形態で、事前の判定によりこのリスケーリングが必要であることが示される場合にのみ入力に適用される、ことが留意されるべきである。リスケーリング操作を入力に適用する前に、リスケーリングが適用されるべきかどうかチェックされ得る。例えば、 $S = n \cdot r$ であるかどうかチェックされ得る。その場合に、ダウンサンプリングレイヤへの入力がダウンサンプリング比の整数倍であるサイズを既に有しているので、リスケーリングは不要であると次に決定され得る。

【0020】

レイヤごとにリスケーリングを適用することによって、リスケーリングは、入力に適用される変更が可能な限り少なくなるように実行される。それによって、品質の低下及び／又は入力のサイズの増大は最小限に抑えられ得る。

【0021】

一実施形態で、 NN は、数

【数2】

$$M \in \mathbb{N}$$

個のダウンサンプリングレイヤを含み、

【数3】

$$m \leq M, m \in \mathbb{N}$$

として、各ダウンサンプリングレイヤ m は、関連するダウンサンプリング比

【数4】

10

20

30

40

50

$$r_m > 1, r_m \in \mathbb{N}$$

を有し、方法は、

少なくとも1つの次元でのサイズ S_m を有するマトリクスを表す入力をダウンサンプリングレイヤ m により処理する前に、サイズ S_m がダウンサンプリングレイヤ m のダウンサンプリング比 r_m の整数倍でない場合に、入力にリスケーリングを適用することであり、リスケーリングは、

【数5】

$$\overline{S_m} = n \cdot r_m, n \in \mathbb{N}$$

であるように、少なくとも1つの次元でのサイズ S_m を変更することを含む、ことを更に有する。より具体的には、ダウンサンプリングレイヤ m の入力に適用されるリスケーリングは、他のダウンサンプリングレイヤのダウンサンプリング比 r_k , $k \neq m$ と無関係であり得る。

【0022】

更なる実施形態で、ダウンサンプリングレイヤ m への入力は、ダウンサンプリングレイヤ m のダウンサンプリング比 r_m の整数倍でない値を有する少なくとも1つの次元でのサイズ S_m を有し、入力のサイズ S_m は、ダウンサンプリング比 r_m の最も近い小さい方の整数倍に一致するか又はダウンサンプリング比 r_m の最も近い大きい方の整数倍に一致するかのどちらかであるようリスケーリング中に変更される。

【0023】

この実施形態によれば、直ぐ後の（又は後続の若しくは連続する、ここでは同義的に使用される）ダウンサンプリングレイヤのダウンサンプリング比に依存するリスケーリングが、この特定のダウンサンプリングレイヤの前にのみ実行されることが指定される。これは、符号化されたビットストリームのサイズが、リスケーリングが入力のサイズの増大を有する場合でさえ最小限に低減されることを保証し得る。

【0024】

上記の実施形態に関して、ダウンサンプリングレイヤへの入力にリスケーリングを適用する前に、リスケーリングが必要であるかどうか（明示的に又は暗黙的に）決定又はチェックされることが提供され得る。リスケーリングが（例えば、入力のサイズが既に各々のダウンサンプリング比の整数倍であるために）不要であると決定される場合に、リスケーリングは適用されないことが提供され得る。

【0025】

一実施形態で、サイズ S_m を有する入力にリスケーリングを適用することの前に、 S_m がダウンサンプリング比 r_m の整数倍であるかどうか決定され、 S_m がダウンサンプリング比 r_m の整数倍でない場合に、リスケーリングが入力に適用され、それにより、サイズ S_m は、

【数6】

$$\overline{S_m} = n \cdot r_m, n \in \mathbb{N}$$

であるように、少なくとも1つの次元で変更される。この決定は、方法の追加のステップという意味で明示的な決定であることができ、あるいは、この決定は、別なふうに行われた計算の部分であることができるので、暗黙的な決定であることができる。結果（サイ

10

20

30

40

50

ズ S_m と整数倍との間の差を示す数という意味で) がこの決定によって実際に取得されることは不要である。

【0026】

リスケーリングを実行する前にかような決定を適用することは、必要な出ないリスケーリングが回避され得るということで、符号化の計算複雑性を低減することができる。

【0027】

一実施形態で、入力サイズ S_m がダウンサンプリングレイヤ m のダウンサンプリング比 r_m の整数倍である場合に、サイズ

【数7】

$$\overline{S_m} \neq S_m$$

10

へのリスケーリングは、ダウンサンプリングレイヤ m によって入力を処理する前に入力に適用されない。この実施形態は、サイズ S_m を変更しないリスケーリングステップが適用されることを除外するものではない。例えば、サイズ $S_m = n \cdot r_m$ である場合に、”リスケーリングされた”サイズは、リスケーリングが

【数8】

$$\overline{S_m} = \text{ceil}\left(\frac{S_m}{r_m}\right)r_m$$

20

に基づいてサイズ S_m を増大させることを有する場合に、ここでは、

【数9】

$$S_m = \overline{S_m}$$

30

であるため、

【数10】

$$\overline{S_m}$$

となる。リスケーリングはこのようにして、この実施形態では何らかのデフォルトステップとして実行され得る一方で、リスケーリングは、入力サイズ S_m の変化を生じさせない。 S_m が既にダウンサンプリング比の整数倍である場合にサイズ

【数11】

$$\overline{S_m} \neq S_m$$

40

へのリスケーリングを止めることによって、サイズの不必要な低減による情報の損失又はサイズの不必要な増大による冗長若しくは無関係な情報の追加は回避される。

【0028】

更なる実施形態で、 S_m がダウンサンプリング比 r_m の整数倍であるかどうかの決定は、サイズ S_m をダウンサンプリングレイヤ m の許容入力サイズと比較することを含む。許

50

容入力サイズは事前定義されてよく、あるいは、それは、このサイズがダウンサンプリング比の整数倍である限りは如何なるサイズであってもよい。具体的に、許容入力サイズは、サイズ S_m に依存してよく、あるいは、それが事前定義されない場合には、 S_m を用いて（動的に）決定されてもよい。

【0029】

一実施形態で、ダウンサンプリングレイヤ m の許容入力サイズは、ダウンサンプリング比 r_m 及びサイズ S_m のうちの少なくとも一方に基づき計算される。

【0030】

これにより、決定は、各々のダウンサンプリングレイヤ及び／又はその入力に特有であり、それによって、潜在的に必要なリスケーリングの決定を信頼できるものとし、具体的には、様々なダウンサンプリング比を有するダウンサンプリングレイヤを備えたニューラルネットワークに適したものとする。

10

【0031】

より具体的に、比較することは、 S_m とダウンサンプリングレイヤ m の許容入力サイズとの間の差を計算することを含んでよい。

【0032】

一実施形態で、比較することは、 $\text{ceil}(S_m/r_m) \cdot r_m - S_m$ 又は $\text{floor}(S_m/r_m) \cdot r_m - S_m$ を決定することを含む。本文脈中、 $\text{ceil}(S_m/r_m) \cdot r_m$ 及び $\text{floor}(S_m/r_m)$ は、上述された「許容入力サイズ」と見なされてよい。

【0033】

より具体的な実施形態では、 $\text{ceil}(S_m/r_m) \cdot r_m - S_m$ が決定され、 $\text{ceil}(S_m/r_m) \cdot r_m - S_m \neq 0$ である場合に、リスケーリングは、サイズ S_m を有する入力に適用される。

20

【0034】

代替的に、又は追加的に、 $\text{floor}(S_m/r_m) \cdot r_m - S_m$ が決定され、 $\text{floor}(S_m/r_m) \cdot r_m - S_m \neq 0$ である場合に、リスケーリングは、サイズ S_m を有する入力に適用される。上記の式に従って決定を行うことによって、 $\text{ceil}(S_m/r_m) \cdot r_m$ 又は $\text{floor}(S_m/r_m)$ が許容入力サイズを構成し、

【数12】

30

$$\overline{S_m}$$

として使用される場合に、同時に、リスケーリング中に、 S_m に加えられるか又は S_m から除かれる必要がある必要量を求めることも可能である。これは、リスケーリングを求める信頼できる方法をもたらし、更には、入力サイズ S_m を可能な限りほとんど変化させないリスケーリングをもたらし得る。

【0035】

更なる実施形態で、サイズ

40

【数13】

$$\overline{S_m}$$

は、ダウンサンプリング比 r_m 及びサイズ S_m のうちの少なくとも一方を用いて決定される。

【0036】

具体的に、サイズ

50

【数 1 4】

$$\overline{S_m}$$

は、`ceil`、`int`、`floor`のうちの少なくとも1つを含む関数を用いて決定され得る。

【0 0 3 7】

より具体的には、サイズ

【数 1 5】

10

$$\overline{S_m}$$

は、次の方法のうちの1つで決定されてよい：

- － サイズ

【数 1 6】

$$\overline{S_m}$$

20

は、

【数 1 7】

$$\text{floor}\left(\frac{S_m}{r_m}\right) \cdot r_m = \overline{S_m}$$

を用いて決定されるか、あるいは、

- － サイズ

【数 1 8】

30

$$\overline{S_m}$$

は、

【数 1 9】

$$\text{ceil}\left(\frac{S_m}{r_m}\right) \cdot r_m = \overline{S_m}$$

40

を用いて決定されるか、あるいは、

- － サイズ

【数 2 0】

50

$$\overline{S_m}$$

は、

【数 2 1】

$$\text{int}\left(\frac{S_m}{r_m}\right) \cdot r_m = \overline{S_m}$$

10

を用いて決定される。

【0 0 3 8】

これらの方法を用いて

【数 2 2】

$$\overline{S_m}$$

20

を計算することは、ダウンサンプリング比の最も近い小さい方の整数倍又はダウンサンプリング比の最も近い大きい方の整数倍がサイズ

【数 2 3】

$$\overline{S_m}$$

を構成するということで、サイズ S_m を有する入力への変更を最小限に抑えることになる。それによって、スケーリング中に入力に加えられる冗長な情報が最小限となり（例えば、リスケーリングが、入力のサイズを増大させるために入力にパディングすることによって行われる場合）、あるいは、リケーリング中に除かれる関連する情報は最小限となる（例えば、入力サイズを低減させるために入力をクロッピングすることによる）。

30

【0 0 3 9】

一実施形態で、ダウンサンプリングレイヤ m への入力は、ダウンサンプリングレイヤ m のダウンサンプリング比 r_m の最も近い小さい方の整数倍と、ダウンサンプリングレイヤ m のダウンサンプリング比 r_m の最も近い大きい方の整数倍との間にある値を有する少なくとも 1 つの次元でのサイズ S_m を有し、条件に応じて、入力のサイズ S_m は、ダウンサンプリング比 r_m の最も近い小さい方の整数倍に一致するか又はダウンサンプリング比 r_m の最も近い大きい方の整数倍に一致するかのどちらかであるようリスケーリング中に変更される。条件は、任意の適切な条件であることができ、例えば、入力からの情報の削除の回避又は入力への冗長な又は意味のない情報の最低限の付加を有し得る。かような条件でのリスケーリング中にサイズを増大させるべきか又はサイズを低減させるべきかどうかの決定に基づくことによって、数学的なだけでなく潜在的に計算上の制約又は要件が満足され得る。更に、これにより、符号化の品質が所与の要件を、例えば、符号化されたピクチャの復号化中の達成可能な品質に関して満足することを保証することが可能である。

40

【0 0 4 0】

一実施形態で、ダウンサンプリングレイヤ m への入力は、少なくとも 1 つの次元でのサイズ S_m を有し、ここで、

50

【数 2 4】

$$lr_m \leq S_m \leq r_m(l+1), l, m \in \mathbb{N}$$

であり、 r_m は、ダウンサンプリングレイヤ m のダウンサンプリング比であり、サイズ S_m は、条件に応じて、

【数 2 5】

$$\overline{S_m} = lr_m \text{ 又は } \overline{S_m} = r_m(l+1)$$

10

のどちらかにリスケーリングされる。望ましくは、 l は、 l 及び $l+1$ がサイズ S に関してダウンサンプリング比の“最も近い”整数倍を指すような小さい自然数である。

【0041】

リスケーリングの計算効率は、元の入力サイズを増大されるか又はサイズを低減されるかどうかによって異なる。具体的に、サイズを lr_m に低減することは、符号化されたビットストリームが最終的により小さくなるとは言え、情報の損失を生じさせる可能性がある。サイズを $r_m(l+1)$ に増大させることは、更なるエントロピが入力に与えられること、及びそれほど有意でない情報により符号化されたピクチャのサイズを増大させることを生じさせる。しかし、少なくとも情報が失われないので、符号化されたピクチャを高品質で復号することが可能である。

20

【0042】

一実施形態で、入力サイズ S_m が、ダウンサンプリング比 r_m の最も近い大きい方の整数倍よりもダウンサンプリングレイヤ m のダウンサンプリング比 r_m の最も近い小さい方の整数倍に近い場合に、入力サイズ S_m は、ダウンサンプリング比 r_m の最も近い小さい方の整数倍に一致するサイズ

【数 2 6】

$$\overline{S_m}$$

30

に低減される。それによって、ほんの少しの情報しか入力から削除されない一方で、著しく大量の冗長又は無駄な情報を追加することは回避される。

【0043】

一実施形態で、入力サイズ S_m をサイズ

【数 2 7】

$$\overline{S_m}$$

40

に低減させることは、入力をクロッピングすることを含む。これは、情報の削除を生じさせるが、符号化方法において効率的に実施され得る。更に、入力の境界は、ピクチャの関連性が高い情報を提供しないので、ピクチャの再構成中に大幅な品質の損失無しに削除され得ることが可能である。

【0044】

50

更なる実施形態で、入力のサイズ S_m が、ダウンサンプリング比 r_m の最も近い小さい方の整数倍よりもダウンサンプリングレイヤ m のダウンサンプリング比 r_m の最も近い大きい方の整数倍に近い場合に、入力のサイズ S_m は、ダウンサンプリング比 r_m の最も近い大きい方の整数倍に一致するサイズ

【数 2 8】

$$\overline{S_m}$$

10

に増大される、ことが提供される。これにより、冗長な又は意味のない情報の量は小さいままであることができ、かつ、入力の情報は失われない。

【0 0 4 5】

具体的に、入力のサイズ S_m をサイズ

【数 2 9】

$$\overline{S_m}$$

に増大させることは、サイズ S_m を有する入力にゼロを又はサイズ S_m を有する入力から得られたパディング情報をパディングすることを含んでもよい。どちらのパディング方法も計算効率よく実装され得る。

【0 0 4 6】

より具体的な実施形態では、サイズ S_m を有する入力から得られたパディング情報は、入力のサイズ S_m をサイズ

【数 3 0】

$$\overline{S_m}$$

20

に増大させるための冗長なパディング情報として適用される。

【0 0 4 7】

より具体的には、冗長なパディング情報によるパディングすることは、反射パディング又は繰り返しパディングのうちの少なくとも一方を含んでよい。これらのパディング方法によれば、既に入力の一部となっている情報を使用することができるので、ピクチャを再構成するときのピクチャの境界でのひずみは軽減される。

【0 0 4 8】

一実施形態で、パディング情報は、冗長なパディング情報が加えられるべき入力内の領域に最も近い、サイズ S_m を有する入力の少なくとも 1 つの値であるか又は該少なくとも 1 つの値を含む。

【0 0 4 9】

一実施形態で、ダウンサンプリングレイヤ m への入力のサイズ S_m は、ダウンサンプリング比 r_m の最も近い大きい方の整数倍に一致するサイズ

【数 3 1】

$$\overline{S_m}$$

30

40

に増大される。これはデフォルトして提供され得、つまり、入力のサイズ S_m がダウンサ

50

ンプリング比 r_m の最も近い大きい方の整数倍よりもレイヤ m のダウンサンプリング比の小さい方の整数倍により近いかどうかに関わらず、サイズ S_m は、例えば、上記の実施形態で見られるようにパディングを使用することによって、増大される。それによって、入力に含まれている情報の損失は回避される。

【0050】

一実施形態で、条件は、 $\text{Min}(|S - lr_m|, |S - r_m(l+1)|)$ を使用し、条件は、 Min が $|S_m - lr_m|$ を投じるか又は選択する場合に、入力のサイズ S_m は

【数32】

$$\overline{S_m} = lr_m$$

10

に低減され、 Min が $|S_m - (l+1)r_m|$ を投じるか又は選択する場合に、入力のサイズ S_m は、

【数33】

$$\overline{S_m} = (l+1)r_m$$

20

に増大されることを含む。この条件を使用することによって、計算上より効率的なリスケーリングが適用されることが確かにされる。

【0051】

具体的に、 l は、ダウンサンプリングレイヤ m への入力のサイズ S_m 及びダウンサンプリングレイヤ m のダウンサンプリング比 r_m のうちの少なくとも一方を用いて決定されてよい。

【0052】

1つの具体的な実施形態では、 l は、 $l = \text{floor}(S_m / r_m)$ によって決定され、かつ／あるいは、 $l+1$ は、 $l+1 = \text{ceil}(S_m / r_m)$ によって決定される。これは、計算上効率の良い方法で上記の値を求めることを可能にする。上記の実施形態から分かるように、 l は、必ずしも何らかの計算の出力の形で明示的に求められず、

30

【数34】

$$\overline{S_m}$$

の計算中に直接使用され得る。

【0053】

これの他の具体的な実現では、条件は、

40

【数35】

$$|\bar{S} - S|$$

を最小化するために値の組 $\{k r_m\}$ からの

【数36】

50

$$\bar{S}$$

の選択であり、ここで、 k は正の整数であり、 S はリスケーリング前の入力サイズである。この条件を使用することによって、計算上より効率的なリスケーリングが適用されることが確かにされ得る。

【0054】

一実施形態で、ダウンサンプリングレイヤの少なくとも1つは、2つの次元で入力（に
ダウンサンプリングを適用し、第1次元でのダウンサンプリング比は、第2次元でのダウ
ンサンプリング比に等しい。この実施形態は、ピクチャのサイズの有意な低減を実現する
。ピクチャのサイズは、例えば、その幅及び高さによって、例えば、幅及び高さの方向に
おけるサンプルの数によって定義される。

【0055】

更なる実施形態では、全てのダウンサンプリングレイヤのダウンサンプリング比は等し
い。全てのダウンサンプリングのダウンサンプリング比が等しい場合に、必要であるリス
ケーリングを決定することは、計算上より容易である。

【0056】

ニューラルネットワークNNにより、ピクチャを表すビットストリームを復号する方法
によれば、一実施形態に従って、NNは、少なくとも1つの次元でサイズ T を有するマト
リクスを表す入力にアップサンプリングを適用する少なくとも2つのアップサンプリング
レイヤを含む複数のレイヤを有し、方法は、第1アップサンプリングレイヤによって入力
を処理し、第1アップサンプリングレイヤの出力を供給することであり、出力は、サイズ
【数37】

$$\bar{T}_1 = T \cdot u_1, u_1 > 1, u_1 \in \mathbb{N}$$

を有し、 u_1 は、第1アップサンプリングレイヤのアップサンプリング比である、ことと、
第2アップサンプリングレイヤによって第1アップサンプリングレイヤの出力を処理す
る前に、第1アップサンプリングレイヤの出力にリスケーリングを適用することであり、
リスケーリングは、少なくとも1つの次元での出力のサイズ

【数38】

$$\bar{T}_1$$

を、取得された情報に基づいて少なくとも1つの次元でのサイズ

【数39】

$$\hat{T}_1$$

に変更することを含む、ことと、

リスケーリング（リサイズ）された出力を第2アップサンプリングレイヤによって処理
し、第2アップサンプリングレイヤの出力を供給することであり、ここで又はそれにより

、出力は、少なくとも 1 つの次元でサイズ
【数 4 0】

$$\overline{T}_2 = \widehat{T}_1 \cdot u_2, u_2 > 1, u_2 \in \mathbb{N}$$

を有し、 u_2 は、第 2 アップサンプリングレイヤのアップサンプリング比である、ことと、
NN を用いてビットストリームを処理した後（例えば、NN の各レイヤでビットストリームを処理した後）、復号されたピクチャを NN の出力として供給することと
を有する。

10

【0 0 5 7】

復号されたピクチャが、ビットストリームを復号する本方法により再構成されるということで、復号されたピクチャは、再構成されたピクチャとも呼ばれ得る。復号されたピクチャは、マトリクスで提示されるか又は表現可能であることができ、その意味で、幅 W 及び高さ H を有し得る。

【0 0 5 8】

更に、

【数 4 1】

20

$$\widehat{T}_1$$

は、第 1 アップサンプリングレイヤの出力の目標出力サイズ及び第 2 アップサンプリングレイヤの目標入力サイズと呼ばれ得る。

【0 0 5 9】

取得された情報は、例えば、復号されたビットストリーム又はそれが表すマトリクスの目標出力サイズに関連する情報を有することができる。この情報は、デコーダで復号化を実行するのに必要であるか又は復号化を支援する情報であることができる。また、出力

【数 4 2】

30

$$\overline{T}_2$$

に必要な特定のサイズのような他の情報も、取得された情報又はその部分であってよい。本開示は更に、どのように情報が取得されるかという具体的な方法に制限されない。例えば、情報は、デコーダに既に利用可能であってよく、あるいは、それは、符号化されたビットストリーム又は追加のビットストリームによりデコーダへ供給されてもよい。また、それらの組み合わせも可能である。

【0 0 6 0】

40

符号化の場合と同様に、ビットストリームは、任意の適当な形式で供給されてよい。この方法によれば、サイズを低減されているビットストリームからビットストリームにおいて符号化されている情報を再構成するためにビットストリームを復号することが可能である。

【0 0 6 1】

アップサンプリングレイヤは、転置畳み込み（逆畳み込み）レイヤを有する、ことが提供され得る。

【0 0 6 2】

転置畳み込みは、「逆畳み込み」とも呼ばれ得る。これは、エンコーダによって適用され得る畳み込みと比較して逆の操作であり、復号されたビットストリームを最終的に得る

50

ために入力の離散表現を含む。

【0063】

一実施形態で、情報は、ビットストリームによって表されるマトリクス of 目標サイズ、ビットストリームによって表現されるマトリクス of 高さ H 、ビットストリームによって表現されるマトリクス of 幅 W 、アップサンプリング比 u_1 、アップサンプリング比 u_2 、第2アップサンプリングレイヤの目標出力サイズ

【数43】

$$\overline{T}_2$$

10

、サイズ

【数44】

$$\widehat{T}_1$$

のうちの少なくとも1つを含む。

20

【0064】

更なる実施形態では、情報は、ビットストリーム、第2ビットストリーム、デコーダで利用可能な情報、のうちの少なくとも1つから取得される。情報は、デコーダに既に知られているので、例えば、アップスケーリング比に関する限り、デコーダで利用可能であることができる。目標サイズのような更なる情報も、デコーダに既に利用可能であることができる。しかし、例えば、中間ステップのサイズは、デコーダに知られていない場合があるので、それがデコーダの詳細に依存し得るということで、ビットストリーム自体によって又は別個のビットストリームで供給されてもよい。これは、使用されるアップサンプリング及びダウンサンプリングレイヤに関してもデコーダがエンコーダの“逆”として実装される場合に特に関係があり、あるいは、そのような場合に特に有利であり得る。これにより、デコーダに必要である情報の量が供給され得、計算効率が良い一方で、同時に、ビットストリームのサイズを最小限に保つ又は低減することができる。

30

【0065】

少なくとも1つの次元でのサイズ

【数45】

$$\overline{T}_1$$

40

をサイズ

【数46】

$$\widehat{T}_1$$

に変更するリスケーリングは、

【数47】

50

$\bar{T}_2$ 及び u_2

に依存する式に基づいて決定されることも企図され得、ここで、
【数 4 8】

$$\bar{T}_2$$

10

は、第 2 アップサンプリングレイヤの出力の目標出力サイズであり、 u_2 は、第 2 アップサンプリングレイヤのアップサンプリング比である。

【0066】

アップサンプリング比 u_2 は固定であるが、この第 2 アップサンプリングレイヤの入力のサイズはリスケーリングに依存するということで、アップサンプリング後の出力サイズも、実際に適用されたリスケーリングに依存する。翻って、出力の目標サイズが固定である（例えば、後続レイヤにおける入力要件による）場合に、これらの条件はリスケーリングを決定するために使用され得る。それによって、通常はデコーダそれ自体に利用可能である情報（具体的に、最終的な目標サイズ）は、ビットストリームに含まれる必要がある情報の量を更に減らすことができ、それによってそのサイズを低減する必須のリスケーリングを決定するために使用され得る。

20

【0067】

更なる実施形態では、少なくとも 1 つの次元でのサイズ

【数 4 9】

$$\bar{T}_1$$

をサイズ

30

【数 5 0】

$$\hat{T}_1$$

に変更するリスケーリングは、 u_2 及び N に依存する式に基づいて決定され、 N は、 NN によるビットストリームの処理順序において第 1 アップサンプリングレイヤに続くアップサンプリングレイヤの総数である。

【0068】

40

この実施形態では、アップサンプリングレイヤの夫々は、同じアップサンプリング比を有し、この実施形態により、必要とされるリスケーリングの計算効率の良い計算が可能である。具体的に、各レイヤのリスケーリングは、アップサンプリング比、最終的な目標サイズ、及び処理される必要がある残りのレイヤの数 N にのみ依存する。しかし、この実施形態は、全てのアップサンプリングレイヤが同じアップサンプリング比を有する場合に制限されない。一例では、第 1 アップサンプリングレイヤに続くレイヤのアップサンプリング比は、2 及び 8 であってよい。この場合に、計算の単純化のために、2 に等しい残りレイヤの数 N 及び 4 に等しいアップサンプリング比 u_2 を考慮に入れることが可能である。

【0069】

一実施形態で、式は、

50

【数 5 1】

$$\hat{T}_1 = \text{ceil} \left(\frac{T_{\text{output}}}{u_2^N} \right)$$

によって与えられ、 T_{output} は、 NN の出力の目標サイズである。

【0070】

式は、後続レイヤのための合理的に処理可能なリスケーリングされた入力を得るために、リスケーリングが常に、前のレイヤの出力に加えられるか又は該出力から減じられ得る整数値であることを保証する。最後のアップサンプリングレイヤについては、アップサンプリングが、いくつかの実施形態で、 NN の出力の目標サイズ T_{output} を常にもたらし、ことが留意されるべきである、具体的に、デコーダの NN の最後のレイヤについては、よって、最後のレイヤへの入力が適切なサイズを有するかどうか又は最後のレイヤの出力が適切なサイズを有するかどうかをチェックしなくてもよい。

【0071】

より具体的には、サイズ T_3 を示す指示が、ビットストリームに含まれてもよい。

【0072】

代替の実施形態では、式は、

【数 5 2】

$$\hat{T}_1 = \text{ceil} \left(\frac{\bar{T}_2}{u_2} \right)$$

によって与えられるか、あるいは、式は、

【数 5 3】

$$\hat{T}_1 = \text{floor} \left(\frac{\bar{T}_2}{u_2} \right)$$

によって与えられる。それによって、リスケーリングは、後続レイヤの出力サイズにのみ依存するように提供される。これは、必要なリスケーリングを決定する際の計算複雑性を低減することができる。

【0073】

複数の事前定義された式のうちのどれが選択されるかを示す指示がビットストリームから取得されることも提供され得る。複数の事前定義された式は、上述されたようにかつ以下で言及されるように、

【数 5 4】

$$\hat{T}_1$$

を決定するための式を含んでもよい。

【0074】

一実施形態で、方法は、サイズ
【数 5 5】

$$\overline{T}_1$$

を有する出力に対するリスケーリングの前に、出力の前イ
【数 5 6】

$$\overline{T}_1$$

10

がサイズ
【数 5 7】

$$\widehat{T}_1$$

20

と一致するかどうかを決定することを更に有する。かような決定は、サイズ
【数 5 8】

$$\overline{T}_1$$

を有する出力の不必要なリスケーリングを潜在的に回避するために使用され得る。
【 0 0 7 5】

30

更には、サイズ
【数 5 9】

$$\overline{T}_1$$

がサイズ
【数 6 0】

40

$$\widehat{T}_1$$

と一致すると決定される場合に、サイズ
【数 6 1】

$$\overline{T}_1$$

50

を変更するリスケーリングは適用されないことが提供され得る。これは、サイズ
【数 6 2】

$$\hat{T}_1 \neq \bar{T}_1$$

へのリスケーリングを生じさせるサイズ
【数 6 3】

$$\bar{T}_1$$

10

にリスケーリングが適用されないことを意味する。しかし、サイズを変更せずに形式的に
リスケーリングを適用する「同一の変換」が、たとえサイズ
【数 6 4】

$$\bar{T}_1$$

20

がサイズ
【数 6 5】

$$\hat{T}_1$$

と一致すると決定されるとしても依然として起用されてよい。例えば、サイズ
【数 6 6】

30

$$\hat{T}_1 = \text{ceil}\left(\frac{\bar{T}_2}{u_2}\right) \text{ 及び } \bar{T}_1 = \frac{\bar{T}_2}{u_2},$$

である場合に、これは
【数 6 7】

$$\hat{T}_1 = \bar{T}_1$$

40

をもたらし、これに基づき適用されたリスケーリングは
【数 6 8】

$$\bar{T}_1$$

を変更しない。
【0 0 7 6】

更なる実施形態では、方法は、サイズ

50

【数 6 9】

$$\overline{T_1}$$

がサイズ
【数 7 0】

$$\widehat{T_1}$$

10

よりも大きいかどうか又はサイズ
【数 7 1】

$$\overline{T_1}$$

20

がサイズ
【数 7 2】

$$\widehat{T_1}$$

よりも小さいかどうかを決定することを更に有する。この決定に基づいて、リスケーリングに関して、更なる動作がとられてもよい。例えば、この決定に基づいて、サイズ
【数 7 3】

30

$$\overline{T_1}$$

は、1つの特定の技術を用いて低減されてよく、あるいは、
【数 7 4】

$$\overline{T_1}$$

40

は、他の技術を用いて増大されてもよい。
【 0 0 7 7】
更なる実施形態では、サイズ
【数 7 5】

$$\overline{T_1}$$

50

がサイズ
【数 7 6】

$$\hat{T}_1$$

よりも大きいと決定される場合に、リスケーリングは、サイズ
【数 7 7】

$$\bar{T}_1$$

10

を有する出力にクロッピングを適用して、サイズ
【数 7 8】

$$\bar{T}_1$$

20

がサイズ
【数 7 9】

$$\hat{T}_1$$

に低減されるようにすることが提供され得る。クロッピングは情報の損失を生じさせるが
、サイズ
【数 8 0】

$$\bar{T}_1$$

30

を低減させるための計算効率の良い方法である。
【0 0 7 8】

一実施形態で、サイズ

【数 8 1】

$$\bar{T}_1$$

40

がサイズ
【数 8 2】

$$\hat{T}_1$$

よりも小さいと決定される場合に、リスケーリングは、サイズ
【数 8 3】

50

$$\overline{T}_1$$

を有する出力にパディングを適用して、サイズ
【数 8 4】

$$\overline{T}_1$$

10

がサイズ
【数 8 5】

$$\widehat{T}_1$$

に増大されるようにすることが提供され得る。パディングは、種々の方法で実現され得る。
具体的に、パディングは、追加情報がサイズ
【数 8 6】

20

$$\overline{T}_1$$

を有する出力の1つ以上の境界にのみ加えられるように提供されてよい。それによって、
再構成されたピクチャ内で関連性が低い可能性のある領域のみパディングの影響を受け得
るので、再構成されたピクチャの品質は向上する。

30

【0 0 7 9】

より具体的な実施形態では、パディングは、サイズ
【数 8 7】

$$\overline{T}_1$$

を有する出力にゼロを又はサイズ
【数 8 8】

40

$$\overline{T}_1$$

を有する出力から得られたパディング情報をパディングすることを有する。いずれの代替
案にも利点がある。ゼロによるパディングは、出力の情報から独立しているということで
、計算効率の良い方法で実現され得る。サイズ
【数 8 9】

50

$$\overline{T}_1$$

を有する出力から得られた情報を出力にパディングすることは、計算上通常はより複雑であるが、出力における実際の情報と付加された情報との間のトランジションを制御することを可能にするので、再構成されたピクチャでの処理アーチファクトを軽減することを可能にする。

【0080】

更なるより具体的な実施形態では、サイズ

【数90】

10

$$\overline{T}_1$$

を有する出力から得られたパディング情報は、出力のサイズ

【数91】

$$\overline{T}_1$$

20

をサイズ

【数92】

$$\hat{T}_1$$

に増大させる冗長なパディング情報として適用される。

【0081】

一実施形態で、パディングは、反射パディング又は繰り返しパディングを有する。

【0082】

更なる実施形態では、パディング情報は、冗長なパディング情報が加えられるべき出力内の領域に最も近い、サイズ

【数93】

30

$$\overline{T}_1$$

40

を有する出力の少なくとも1つの値であるか又は該少なくとも1つの値を含む。これは、実際の出力とパディングによって付加された情報との間のトランジションでのひずみを低減させる。

【0083】

更には、

【数94】

50

$$\overline{T}_1$$

が

【数 9 5】

$$\hat{T}_1$$

10

に等しくないと決定される場合に、リスケーリングは補間フィルタを適用することを含むことが更に提供され得る。

【0 0 8 4】

一般的に、サイズ

【数 9 6】

$$\hat{T}_1$$

20

を計算するためのリサイジング式は、次の形式：

【数 9 7】

$$\hat{T}_1 = f(\overline{T}_2, u_2)$$

を有し得る。例えば、

【数 9 8】

$$\hat{T}_1 = \text{floor}\left(\frac{\overline{T}_2}{u_2}\right)$$

30

である。

【0 0 8 5】

一例で、各アップサンプリングレイヤ後にサイズを計算するための式は異なってもよい。例えば：

【数 9 9】

$$\hat{T}_1 = \text{ceil}\left(\frac{\overline{T}_2}{u_2}\right), \text{ 及び } \hat{T}_2 = \text{floor}\left(\frac{\overline{T}_3}{u_3}\right)$$

40

である。

【0 0 8 6】

情報は、ビットストリーム又は更なるビットストリームで提供され、ビットストリームを符号化したエンコーダの少なくとも1つのダウンサンプリングレイヤのダウンサンプリング比 r_m を含み、ダウンサンプリングレイヤは、入力进行处理する順序において、デコーダのアップサンプリングレイヤに対応することも提供され得る。

50

【0087】

この実施形態は、エンコーダ及びデコーダが互いの厳密に反転であることによって互いに対応している場合に、特に有利である。

【0088】

一実施形態で、アップサンプリングレイヤの少なくとも1つは、2つの次元でアップサンプリングを適用し、第1次元でのアップサンプリング比は、第2次元でのアップサンプリング比に等しい。

【0089】

更なる実施形態では、全てのアップサンプリングレイヤのアップサンプリング比は等しい。

【0090】

これは、デコーダの計算複雑性を軽減する。

【0091】

更に、ビットストリームは、幅Wを有する第1次元及び幅Hを有する第2次元を有するマトリクスを表すか、及び／又は復号されたビットストリームは再構成されたピクチャ、例えば、静止画及び／又はビデオピクチャであることが提供され得る。かようなビットストリームについては、当該復号化方法を適用することが特に有利である。

【0092】

更なる実施形態では、リスケーリングは、リスケーリングに従って入力の変更するために補間、パディングクロッピングを入力に適用することの少なくとも1つであるか又はその少なくとも1つを含むリスケーリング操作を有する。これらのリスケーリング操作は、リスケーリングが元の入力のサイズの増大又は元の入力のサイズの低減を有する場合に有利に適用され得る。

【0093】

一実施形態で、NNは、NNによるビットストリームの処理順序において、少なくとも1つの次元での入力のサイズを変更しない入力への変換を適用する更なるユニットを含み、方法は、更なるユニットによる入力の処理の後、後続のレイヤによって入力を処理する前に、リスケーリングが少なくとも1つの次元での入力のサイズの増大をもたらす場合に、リスケーリングを適用することを有し、かつ／あるいは、方法は、更なるユニットによる入力の処理の前に、リスケーリングが少なくとも1つの次元での入力のサイズの低減を含む場合に、リスケーリングを適用することを有する。

【0094】

それによって、ニューラルネットワークの各レイヤ又はステップで必要な処理は、最小限に保たれる。

【0095】

更なるユニットは、バッチ正規化部及び／又は正規化線形ユニットReLUであることができる、あるいは、それを有することができる。

【0096】

かような更なるユニットは、これらのユニットの後又はそれらの前にリスケーリングが行われる場合に計算効率を向上させることができるマトリクスのエントリへの変更を適用する。それによって、冗長な又は意味のない情報の不必要な処理は回避される。

【0097】

また更には、ビットストリームは、ビットストリームの異なるカラーチャネルに対応するサブビットストリームを有してもよく、NNは、サブニューラルネットワークsNNを有し、sNNは夫々、上記の実施形態のいずれかに係る方法を、sNNへの入力として供給されたサブビットストリームに適用するよう構成される。

【0098】

これは、本開示を、例えば、基礎を成すビットストリームの異なるカラーチャネルが考慮されるべきであるが、これらが互いに独立して符号化される場合にも適用可能にする。

【0099】

10

20

30

40

50

更なる実施形態では、リスケーリングがサイズ S をサイズ
【数 1 0 0】

$$\bar{S}$$

に増大させることを含む場合に、サイズ
【数 1 0 1】

$$\bar{S}$$

10

は、
【数 1 0 2】

$$\bar{S} = \text{Int} \left(\frac{S + r - 1}{r} \right) r$$

によって与えられ、リスケーリングがサイズ S をサイズ
【数 1 0 3】

20

$$\bar{S}$$

に低減させることを含む場合に、サイズ
【数 1 0 4】

$$\bar{S}$$

30

は、
【数 1 0 5】

$$\bar{S} = \text{Int} \left(\frac{S}{r} \right) r$$

40

によって与えられることが提供される。これらのリスケーリングによれば、後続のダウン
サンプリングレイヤへのリスケーリングされた入力のサイズは常に、サイズ S を有する入
力を処理する後続のダウンサンプリングレイヤのダウンサンプリング比 r の整数倍である
ことが保証される。

【0 1 0 0】

他の実施形態では、リスケーリングがサイズ

【数 1 0 6】

50

$$\overline{T}_1$$

をサイズ
【数 1 0 7】

$$\widehat{T}_1$$

10

に増大させることを含む場合に、サイズ
【数 1 0 8】

$$\widehat{T}_1$$

は、
【数 1 0 9】

20

$$\widehat{T}_1 = \text{Int} \left(\frac{\overline{T}_1 + u_2 - 1}{u_2} \right) u_2$$

によって与えられ、リスケーリングがサイズ
【数 1 1 0】

$$\overline{T}_1$$

30

をサイズ
【数 1 1 1】

$$\widehat{T}_1$$

40

に低減させることを含む場合に、サイズ
【数 1 1 2】

$$\widehat{T}_1$$

は、
【数 1 1 3】

50

$$\hat{T}_1 = \text{Int} \left(\frac{\bar{T}_1}{u_2} \right) u_2$$

によって与えられることが提供される。これによれば、アップサンプリングレイヤの出力は、続くアップサンプリングレイヤによって合理的に処理され得るサイズを有することが保証される。

【0101】

一実施形態に従って、ピクチャを符号化するエンコーダが提供され、エンコーダは、ピクチャを受信（取得）する受信器又はインターフェースと、ニューラルネットワークNNを実装するよう構成され、NNは、NNによるピクチャの処理順序において、入力にダウンサンプリングを適用するよう構成される少なくとも1つのダウンサンプリングレイヤを含む複数のレイヤを有する、1つ以上のプロセッサと、ビットストリームを出力する送信器とを有し、エンコーダは、上記の実施形態のいずれかに係る方法を実行するよう構成される。このエンコーダは、ピクチャを符号化するための上記の実施形態に係る方法を有利に実施する。

【0102】

更なる実施形態では、ピクチャを符号化するエンコーダが提供され、エンコーダは、ニューラルネットワークNNを実装する1つ以上のプロセッサを有し、1つ以上のプロセッサは、上記の実施形態のいずれかに係る方法を実行するよう構成される。

【0103】

従って、例えば、ニューラルネットワークNNによりピクチャを符号化するエンコーダが提供され、エンコーダは、NNを実装する1つ以上のプロセッサを有し、NNは、少なくとも1つの次元でサイズSを有するマトリクスを表す入力にダウンサンプリングを適用する少なくとも1つのダウンサンプリングレイヤを含む複数のレイヤを有し、1つ以上のプロセッサは、少なくとも1つのダウンサンプリングレイヤにより入力を処理する前に、入力にリスケージングを適用することであり、リスケージングは、

【数114】

$$\bar{S} = n \cdot r$$

なお、 $n \in \mathbb{N}$ 及び $r > 1, r \in \mathbb{N}$

であるように、少なくとも1つの次元でのサイズSを変更することを含み、ここで、rは、レイヤのダウンサンプリング比である、ことと、リスケージングの後に、ダウンサンプリングレイヤによって入力を処理し、サイズ $S_1 < S$ を有する出力を供給することと、NNによりピクチャを処理した後（例えば、NNの各レイヤでピクチャを処理した後）、ビットストリームをNNの出力として供給することとを実行するよう構成される。

【0104】

一実施形態で、ピクチャを表すビットストリームを復号するデコーダが提供され、デコーダは、ビットストリームを受信する受信器又はインターフェースと、ニューラルネットワークNNを実装するよう構成され、NNは、NNによるビットストリームの処理順序において、入力にアップサンプリングを適用するよう構成される少なくとも2つのアップサンプリングレイヤを含む複数のレイヤを有する、1つ以上のプロセッサと、復号されたピクチャを出力する送信器とを有し、デコーダは、上記の実施形態の方法のいずれかを実行するよう構成される。このデコーダは、ピクチャを表すビットストリームを復号する上記の方法の利点を実現する。

【0105】

一実施形態で、ピクチャを表すビットストリームを復号するデコーダが提供され、デコーダは、ニューラルネットワークNNを実装する1つ以上のプロセッサを有し、1つ以上のプロセッサは、上記の実施形態のいずれかに係る方法を実行するよう構成される。

【0106】

従って、例えば、ピクチャを表すビットストリームを復号するデコーダが提供され、デコーダは、NNを実装する1つ以上のプロセッサを有し、NNは、少なくとも1つの次元でサイズTを有するマトリクスを表す入力にアップサンプリングを適用する少なくとも第1アップサンプリングレイヤ及び第2アップサンプリングレイヤを含む複数のレイヤを有し、1つ以上のプロセッサは、第1アップサンプリングレイヤによって入力を処理し、第1アップサンプリングレイヤの出力を供給することであり、出力は、サイズ

10

【数115】

$$\overline{T}_1 = T \cdot u_1, u_1 > 1, u_1 \in \mathbb{N}$$

を有し、 u_1 は、第1アップサンプリングレイヤのアップサンプリング比である、ことと、第2アップサンプリングレイヤによって第1アップサンプリングレイヤの出力を処理する前に、第1アップサンプリングレイヤの出力にリスケーリングを適用することであり、リスケーリングは、少なくとも1つの次元での出力のサイズ

20

【数116】

$$\overline{T}_1$$

を、取得された情報に基づいて少なくとも1つの次元でのサイズ

【数117】

$$\widehat{T}_1$$

30

に変更することを含む、ことと、リスケーリングされた出力を第2アップサンプリングレイヤによって処理し、第2アップサンプリングレイヤの出力を供給することであり、出力は、少なくとも1つの次元でサイズ

【数118】

$$\overline{T}_2 = \widehat{T}_1 \cdot u_2, u_2 > 1, u_2 \in \mathbb{N}$$

40

を有し、 u_2 は、第2アップサンプリングレイヤのアップサンプリング比である、ことと、NNを用いてビットストリームを処理した後（例えば、NNの各レイヤでビットストリームを処理した後）、復号されたピクチャをNNの出力として供給することを実行するよう構成される。

【0107】

実施形態において、コンピュータプログラムが提供され、コンピュータプログラムは、例えば、コンピューティングシステムで又は1つ以上のプロセッサによって実行される場合に、上記の実施形態のいずれかに係る方法を実行するよう構成される。

50

【0108】

他の実施形態では、コンピュータ可読記憶媒体が提供され、コンピュータ可読記憶媒体は、コンピューティングシステムで実行される場合に、コンピューティングシステムに、上記の実施形態のいずれかに係る方法を実行させるコンピュータ実行可能命令を有する。具体的に、コンピュータ可読記憶媒体は非一時的な媒体であることができる。

【図面の簡単な説明】

【0109】

【図1A】本開示の実施形態を実施するよう構成されるビデオコーディングシステムの例を示すブロック図である。

【図1B】本開示のいくつかの実施形態を実施するよう構成されるビデオコーディングシステムの他の例を示すブロック図である。

【図2】符号化装置又は復号化装置の例を表すブロック図である。

【図3】符号化装置又は復号化装置の他の例を表すブロック図である。

【図4】一実施形態に係るエンコーダ及びデコーダを一緒に示す。

【図5】入力の符号化及び復号化の模式図を示す。

【図6】VAEフレームワークに従うエンコーダ及びデコーダを示す。

【図7】一実施形態に従って図4に係るエンコーダのコンポーネントを示す。

【図8】一実施形態に従って図4に係るデコーダのコンポーネントを示す。

【図9】入力のリスケーリング及び処理を示す。

【図10】エンコーダ及びデコーダを示す。

【図11】更なるエンコーダ及び更なるデコーダを示す。

【図12】一実施形態に従って入力のリスケーリング及び処理を示す。

【図13】一実施形態に従ってリスケーリングオプションのシグナリングの実施形態を示す。

【図14】図13に係る実施形態のより具体的な実現を示す。

【図15】図14に係る実施形態のより具体的な実現を示す。

【図16】パディング操作の種々の可能性の比較を示す。

【図17】パディング操作の種々の可能性の更なる比較を示す。

【図18】一実施形態に従って、エンコーダ及びデコーダと、エンコーダ及びデコーダへの入力の処理における関係とを示す。

【図19】一実施形態に係るエンコーダの模式図を示す。

【図20】一実施形態に係るデコーダの模式図を示す。

【図21】符号化の実施形態を示すフロー図を示す。

【図22】復号化の実施形態のフロー図を示す。

【図23】リスケーリングするとの決定が条件に応じてどのように行われるかの実施形態を示す。

【発明を実施するための形態】

【0110】

以下で、いくつかの実施形態は、図を参照して記載される。図1から3は、更なる図で記載される本発明のより具体的な実施形態とともに使用され得るビデオコーディングのシステム及び方法を参照する。具体的に、図1から3に関して記載される実施形態は、ビットストリームの符号化及び／又はビットストリームの復号化のためにニューラルネットワークを利用する、以下で更に記載される符号化／復号化技術とともに使用されてよい。

【0111】

以下の記載では、本開示の一部を形成し、実例として、本開示の具体的な態様、又は本開示の実施形態が使用される可能性がある具体的な態様を示す添付の図面が参照される。実施形態は、他の態様で使用され、図に表されていない構造上又は論理上の変更を含んでもよいことが理解される。従って、以下の詳細な説明は、限定の意味で解釈されるべきではなく、本開示の範囲は、添付の特許請求の範囲によって定義される。

【0112】

例えば、開示は、記載される方法に関連して、方法を実行するよう構成される対応するデバイス又はシステムにも当てはまり、逆もまた然りであることが理解される。例えば、1つ又は複数の具体的な方法ステップが記載される場合に、対応する装置は、記載されている1つ又は複数の方法ステップを実行するための1つ又は複数のユニット、例えば、機能ユニット（例えば、1つ又は複数のステップを実行する1つのユニット、あるいは、複数のステップのうちの1つ以上を夫々実行する複数のユニット）を、たとえそのような1つ以上のユニットが明示的に記載又は図示されていないとしても、含んでよい。他方で、例えば、具体的な装置が1つ又は複数のユニット、例えば、機能ユニットに基づいて記載される場合に、対応する方法は、1つ又は複数のユニットの機能を実行するための1つ又は複数のステップ（例えば、1つ又は複数のユニットの機能を実行する1つのステップ、あるいは、複数のユニットのうちの1つ以上の機能を夫々実行する複数のステップ）を、たとえそのような1つ又は複数のステップが明示的に記載又は図示されていないとしても、含んでよい。更に、ここで記載されている様々な例示的な実施形態及び／又は態様の特徴は、特に別なふうに述べられない限りは、互いに結合されてもよいことが理解される。

【0113】

ビデオコーディングは、通常、ビデオ又はビデオシーケンスを形成するピクチャの連続の処理を指す。「ピクチャ」という用語の代わりに、「フレーム」又は「画像」という用語が、ビデオコーディングの分野では同義語として使用される場合がある。ビデオコーディング（又はコーディング全般）は、2つの部分、つまり、ビデオ符号化及びビデオ復号化を含む。ビデオ符号化はソース側で実行され、通常、（より効率的な記憶及び／又は伝送のために）ビデオピクチャを表現するのに必要なデータの量を減らすよう元のビデオピクチャを（例えば、圧縮によって）処理することを含む。ビデオ復号化はあて先側で実行され、通常、ビデオピクチャを再構成するためにエンコーダと比較して逆の処理を含む。ビデオピクチャ（又はピクチャ全般）の“コーディング”に言及する実施形態は、ビデオピクチャ又は各々のビデオシーケンスの“符号化”又は“復号化”に関係があると理解されるべきである。符号化部分及び復号化部分の結合は、CODEC（Coding and Decoding）とも呼ばれる。

【0114】

可逆ビデオコーディングの場合に、元のビデオピクチャは再構成可能であり、つまり、再構成されたビデオピクチャは元のビデオピクチャと同じ品質を有する（記憶又は伝送の間に伝送損失又は他のデータ損失がないとする）。不可逆ビデオコーディングの場合に、例えば、量子化による、更なる圧縮が、ビデオピクチャを表すデータの量を減らすために実行され、これにより、ビデオピクチャはデコーダで完全には再構成され得ず、つまり、再構成されたビデオピクチャの品質は、元のビデオピクチャの品質と比べて低いか又は悪い。

【0115】

いくつかのビデオコーディング標準規格は、“不可逆ハイブリッドビデオコーデック”（つまり、同じ領域での空間及び時間予測と、変換領域で量子化を適用する2D変換コーディングとを組み合わせること）のグループに属する。ビデオシーケンスの各ピクチャは、通常、重なり合わないブロックの組にパーティション化され、コーディングは、通常、ブロックレベルで実行される。換言すれば、エンコーダでは、ビデオは、通常、ブロック（ビデオブロック）レベルで、例えば、空間（イントラピクチャ）予測及び／又は時間（インターピクチャ）予測を用いて予測ブロックを生成し、予測ブロックを現在ブロック（現在処理中の／処理されることになっているブロック）から引き算して残差ブロックを取得し、残差ブロックを変換し、残差ブロックを変換領域で量子化して、伝送されるべきデータの量を減らすこと（圧縮）によって、処理、つまり、符号化され、一方、デコーダでは、エンコーダと比較して逆の処理が、表示のために現在ブロックを再構成するよう、符号化又は圧縮されたブロックに適用される。更に、エンコーダはデコーダ処理ループを複製し、それにより、両方ともが、後続のブロックを処理、つまり、コーディングするために、同じ予測（例えば、イントラ及びインター予測）及び／又は再構成を生成することにな

る。近年、いくつかの部分又は符号化及び復号化チェーンの全体は、ニューラルネットワーク又は、一般に、任意の機械学習若しくはディープラーニングフレームワークを使用することによって実施されている。

【0116】

ビデオコーディングシステム10の以下の実施形態では、ビデオエンコーダ20及びビデオデコーダ30が図1に基づいて記載される。

【0117】

図1Aは、例となるコーディングシステム10、例えば、本願の技術を利用し得るビデオコーディングシステム（略してコーディングシステム10）を表す略ブロック図である。ビデオコーディングシステム10のビデオエンコーダ20（略してエンコーダ20）及びビデオデコーダ30（略してデコーダ30）は、本願で記載される様々な例に従う技術を実行するよう構成され得るデバイスの例を表す。

【0118】

図1Aに示されるように、コーディングシステム10は、例えば、符号化されたピクチャデータ21を復号するあて先デバイス14へ、符号化されたピクチャデータ21を供給するよう構成されるソースデバイス12を有する。

【0119】

ソースデバイス12はエンコーダ20を有し、更には、つまり、任意に、ピクチャソース16、プリプロセッサ（又は前処理ユニット）18、例えば、ピクチャプリプロセッサ18、及び通信インターフェース又は通信ユニット22を有してもよい。本開示のいくつかの実施形態（例えば、初期リスケージング又は2つの連続するレイヤ間のリスケージングに関係がある）は、エンコーダ20によって実施されてよい。いくつかの実施形態（例えば、初期リスケージングに関係がある）は、ピクチャプリプロセッサ18によって実施されてよい。

【0120】

ピクチャソース16は、任意の種類のピクチャ捕捉デバイス、例えば、現実世界のピクチャを捕捉するカメラ、及び／又は任意の種類のピクチャ生成デバイス、例えば、コンピュータアニメーション化されたピクチャを生成するコンピュータグラフィックスプロセッサ、あるいは、現実世界のピクチャ、コンピュータにより生成されたピクチャ（例えば、スクリーンコンテンツ、仮想現実（VR）ピクチャ）及び／又はそれらの任意の組み合わせ（例えば、拡張現実（AR）ピクチャ）を取得及び／又は供給する任意の種類の他のデバイスを有しても又はそのようなものであってもよい。ピクチャソースは、上記のピクチャのいずれかを記憶する任意の種類のメモリ又はストレージであってもよい。

【0121】

プリプロセッサ18及び前処理ユニット18によって実行される処理と区別して、ピクチャ又はピクチャデータ17は、生ピクチャ又は生ピクチャデータ17とも呼ばれ得る。

【0122】

プリプロセッサ18は、（生）ピクチャデータ17を受け取り、ピクチャデータ17に前処理を実行して、前処理されたピクチャ19又は前処理されたピクチャデータ19を取得するよう構成される。プリプロセッサ18によって実行される前処理は、例えば、トリミング、（例えば、RGBからYCbCrへの）カラーフォーマット変換、色補正、又はノイズ除去を含んでもよい。前処理ユニット18は任意のコンポーネントであってもよい、ことが理解され得る。

【0123】

ビデオエンコーダ20は、前処理されたピクチャデータ19を受け取り、符号化されたピクチャデータ21を供給するよう構成される。

【0124】

ソースデバイス12の通信インターフェース22は、符号化されたピクチャデータ21を受け取り、符号化されたピクチャデータ21（又はその任意の更に処理されたバージョン）を通信チャネル13上で他のデバイス、例えば、あて先デバイス14又は任意の他の

10

20

30

40

50

デバイスへ記憶又は直接の再構成のために送信するよう構成されてよい。

【0125】

あて先デバイス14はデコーダ30（例えば、ビデオデコーダ30）を有し、更には、つまり任意に、通信インターフェース又は通信ユニット28、ポストプロセッサ32（又は後処理ユニット32）、及び表示デバイス34を有してもよい。

【0126】

あて先デバイス14の通信インターフェース28は、符号化されたピクチャデータ21（又はその任意の更に処理されたバージョン）を、例えば、ソースデバイス12から直接又は、任意の他のソース、例えば、記憶デバイス、例えば、符号化ピクチャデータ記憶デバイスから受け取り、符号化されたピクチャデータ21をデコーダ30へ供給するよう構成される。

10

【0127】

通信インターフェース22及び通信インターフェース28は、符号化されたピクチャデータ21又は符号化されたデータ21を、ソースデバイス12とあて先デバイス14との間の直接通信リンク、例えば、直接有線若しくは無線接続を介して、あるいは、任意の種類のネットワーク、例えば、有線若しくは無線ネットワーク又はそれらの任意の組み合わせ、又は任意の種類のプライベート及びパブリックネットワーク、あるいは、それらの任意の組み合わせを介して、送信又は受信するよう構成されてよい。

【0128】

通信インターフェース22は、例えば、符号化されたピクチャデータ21を適切なフォーマット、例えば、パケットにパッケージ化するよう、及び／又は通信リンク若しくは通信ネットワーク上での伝送のために任意の種類の伝送符号化若しくは処理を使用して、符号化されたピクチャデータを処理するよう構成されてよい。

20

【0129】

通信インターフェース28は、通信インターフェース22の対応物を形成し、例えば、送信されたデータを受信し、任意の種類の対応する伝送復号化若しくは処理及び／又はデパッキングにより伝送データを処理して、符号化されたピクチャデータ21を取得するよう構成されてよい。

【0130】

通信インターフェース22及び通信インターフェース28のいずれも、図1Aでソースデバイス12からあて先デバイス14に向けた通信チャネル13の矢印によって示されるような一方向の通信インターフェース、又は双方向の通信インターフェースとして構成されてよく、例えば、メッセージを送信又は受信するよう、例えば、通信リンク及び／又はデータ伝送、例えば、符号化されたピクチャデータ伝送に関係がある任意の他の情報を承認及び交換するために、接続をセットアップするよう構成されてよい。

30

【0131】

デコーダ30は、符号化されたピクチャデータ21を受け取り、復号されたピクチャデータ31又は復号されたピクチャ31を供給するよう構成される（更なる詳細は、例えば、図3に基づいて、以下で記載される）。

【0132】

あて先デバイス14のポストプロセッサ32は、復号されたピクチャデータ31（再構成されたピクチャデータとも呼ばれる）、例えば、復号されたピクチャ31を後処理して、後処理されたピクチャデータ33、例えば、後処理されたピクチャ33を取得するよう構成される。後処理ユニット32によって実行される後処理は、例えば、（例えば、YCbCrからRGBへの）カラーフォーマット変換、色補正、トリミング、若しくはリサンプリング、又は、例えば、復号されたピクチャデータ31を、例えば、表示デバイス34による表示のために準備するための任意の他の処理を有してもよい。

40

【0133】

本開示のいくつかの実施形態は、デコーダ30によって又はポストプロセッサ32によって実施されてよい。

50

【0134】

あて先デバイス14の表示デバイス34は、ピクチャを、例えば、ユーザ又はビューアに表示するために、後処理されたピクチャデータを受け取るよう構成される。表示デバイス34は、再構成されたピクチャを表示するための任意の種類のディスプレイ、例えば、内蔵された又は外付けのディスプレイ又はモニタであっても、あるいは、そのようなものを有してもよい。ディスプレイは、例えば、液晶ディスプレイ（LCD）、有機発光ダイオード（OLED）ディスプレイ、プラズマディスプレイ、プロジェクタ、マイクロLEDディスプレイ、リキッドクリスタル・オン・シリコン（LCOS）、デジタルライトプロセッサ（DLP）又は任意の種類の他のディスプレイを有してもよい。

【0135】

図1Aはソースデバイス12及びあて先デバイス14を別個のデバイスとして表しているが、デバイスの実施形態は、両方又は両方の機能、ソースデバイス12又は対応する機能及びあて先デバイス14又は対応する機能を有してもよい。かような実施形態では、ソースデバイス又は対応する機能及びあて先デバイス14又は対応する機能は、同じハードウェア及び／又はソフトウェアを用いて、あるいは、別個のハードウェア及び／又はソフトウェア又はそれらの任意の組み合わせによって、実施されてもよい。

【0136】

記載に基づき当業者に明らかなように、異なるユニットの機能、つまり、図1Aに示されるソースデバイス12及び／又はあて先デバイス14内の機能の存在及び（厳密な）分割は、実際のデバイス及びアプリケーションに応じて様々であってよい。

【0137】

エンコーダ20（例えば、ビデオエンコーダ20）若しくはデコーダ30（例えば、ビデオデコーダ30）又はエンコーダ20及びデコーダ30の両方は、1つ以上のマイクロプロセッサ、デジタルシグナルプロセッサ（DSP）、特定用途向け集積回路（ASIC）、フィールドプログラマブルゲートアレイ（FPGA）、ディスクリートロジック、ハードウェア、ビデオコーディング専用、又はそれらの任意の組み合わせなどの、図1Bに示されている処理回路により実施されてもよい。エンコーダ20は、ここで記載されている様々なモジュール及び／又は任意の他のエンコーダシステム若しくはサブシステムを具現化するよう処理回路46により実施されてもよい。デコーダ30は、ここで記載されている様々なモジュール及び／又は任意の他のデコーダシステム若しくはサブシステムを具現化するよう処理回路46により実施されてもよい。処理回路は、後述される様々な動作を実行するよう構成されてよい。図3に示されるように、技術が部分的にソフトウェアで実施される場合に、デバイスは、適切な、非一時的なコンピュータ可読記憶媒体にソフトウェア用の命令を記憶してよく、1つ以上のプロセッサを使用してハードウェアで命令を実行して、本開示の技術を実行してよい。ビデオエンコーダ20及びビデオデコーダ30のいずれも、例えば、図1Bに示されるように、単一のデバイスで複合エンコーダ／デコーダ（CODEC）の部分として組み込まれてもよい。

【0138】

ソースデバイス12及びあて先デバイス14は、任意の種類の携帯又は固定デバイス、例えば、ノートブック若しくはラップトップコンピュータ、携帯電話、スマートフォン、タブレット若しくはタブレットコンピュータ、カメラ、デスクトップコンピュータ、セットトップボックス、テレビジョン、表示デバイス、デジタルメディアプレイヤー、ビデオゲーム機、ビデオストリーミングデバイス（例えば、コンテンツサービスサーバ若しくはコンテンツ配信サーバ）、放送受信機、放送送信機、などを含む、広範なデバイスのうちのいずれかを有してよく、オペレーティングシステムを全く使用しなくても又は任意の種類のオペレーティングシステムを使用してもよい。いくつかの場合に、ソースデバイス12及びあて先デバイス14は、無線通信のために装備されてもよい。よって、ソースデバイス12及びあて先デバイス14は、無線通信デバイスであってもよい。

【0139】

いくつかの場合に、図1Aに表されているビデオコーディングシステム10は、一例に

すぎず、本願の技術は、符号化デバイスと復号化デバイスとの間の如何なるデータ通信も必ずしも含むわけではないビデオコーディング設定（例えば、ビデオ符号化又はビデオ復号化）に適用されてもよい。他の例では、データは、ローカルメモリから読み出されるか、ネットワーク上でストリーミングされるか、などである。ビデオ符号化デバイスは、データを符号化してメモリに記憶してよく、かつ／あるいは、ビデオ復号化デバイスは、メモリからデータを読み出して復号してよい。いくつかの例では、符号化及び復号化は、互いに通信しないデバイスによって実行されるが、単純にデータを符号化してメモリに記憶しかつ／あるいはデータをメモリから読み出して復号する。

【0140】

記載の便宜上、いくつかの実施形態は、ここでは、例えば、High-Efficiency Video Coding (HEVC) を参照して、又はITU-T Video Coding Experts Group (VCEG) 及ISO/IEC Motion Picture Experts Group (MPEG) のJoint Collaboration Team on Video Coding (JCT-VC) によって開発された次世代のビデオコーディング標準であるVersatile Video coding (VVC) の参照ソフトウェアを参照して、記載される。当業者であれば、本発明の実施形態がHEVC又はVVCに制限されないことを理解するだろう。

【0141】

図2は、本開示の実施形態に従うビデオコーディングデバイス400の模式図である。ビデオコーディングデバイス400は、ここで記載されている開示された実施形態を実装するのに適している。実施形態において、ビデオコーディングデバイス400は、図1Aのビデオデコーダなどのデコーダ、又は図1Aのビデオエンコーダ20などのエンコーダであってよい。

【0142】

ビデオコーディングデバイス400は、データを受け取る入力ポート410（又は入力ポート410）及び受信器ユニット（Rx）420と、データを処理するためのプロセッサ、ロジックユニット、又は中央演算処理装置（CPU）430と、データを送信する送信器ユニット（Tx）440及び出口ポート450（又は出力ポート450）と、データを記憶するメモリ460とを有する。ビデオコーディングデバイス400はまた、光又は電気信号の出口又は入口のために入力ポート410、受信器ユニット420、送信器ユニット440、及び出口ポート450へ結合された光電気（OE）コンポーネント及び電気光（EO）コンポーネントを有してもよい。

【0143】

プロセッサ430は、ハードウェア及びソフトウェアによって実装される。プロセッサ430は、1つ以上のCPUチップ、コア（例えば、マルチコアプロセッサとして）、FPGA、ASIC、及びDSPとして実装されてよい。プロセッサ430は、入力ポート410、受信器ユニット420、送信器ユニット440、出口ポート450、及びメモリ460と通信する。プロセッサ430はコーディングモジュール470を有する。コーディングモジュール470は、上記の開示された実施形態を実装する。例えば、コーディングモジュール470は、様々なコーデック操作を実装、処理、準備、又は提供する。コーディングモジュール470の包含は、従って、ビデオコーディングデバイス400の機能性を実質的な改善をもたらし、異なる状態へのビデオコーディングデバイス400の変形を達成する。代替的に、コーディングモジュール470は、メモリ460に記憶されてプロセッサ430によって実行される命令として実装される。

【0144】

メモリ460は、1つ以上のディスク、テープドライブ、及びソリッドステートドライブを有してよく、プログラムが実行のために選択される場合にかようなプログラムを記憶するために、かつ、プログラム実行中に読み出される命令及びデータを記憶するために、オーバーフローデータ記憶デバイスとして使用されてもよい。メモリ460は、例えば、揮発性及び／又は不揮発性であってよく、リードオンリーメモリ（ROM）、ランダムア

クスメモリ（RAM）、三値連想メモリ（TCAM）、及び／又は静的ランダムアクセスメモリ（SRAM）であってもよい。

【0145】

図3は、例示的な実施形態に従って、ソースデバイス12及びアークデバイス14の一方又は両方として使用され得る装置500の略ブロック図である。

【0146】

装置500のプロセッサ502は、中央演算処理装置であることができる。代替的に、プロセッサ502は、現在存在しているか又は今後開発される、情報を操作又は処理可能な任意の他のタイプのデバイス又は複数のデバイスであることができる。開示されている実施は、示されるように単一のプロセッサ、例えば、プロセッサ502で実施することができるが、速度及び効率における利点は、1よりも多いプロセッサを用いて達成することができる。

10

【0147】

装置500のメモリ504は、実施においてランダムアクセスメモリ（ROM）デバイス又はランダムアクセスメモリ（RAM）デバイスであることができる。任意の他の適切なタイプの記憶デバイスがメモリ504として使用され得る。メモリ504は、バス512を用いてプロセッサ502によってアクセスされるコード及びデータ506を含むことができる。メモリ504は、オペレーティングシステム508及びアプリケーションプログラム510を更に含むことができ、アプリケーションプログラム510は、プロセッサ502がここで記載されている方法を実行することを可能にする少なくとも1つのプログラムを含む。例えば、アプリケーションプログラム510は、アプリケーション1乃至Nを含むことができ、更には、ここで記載されている方法を実行するビデオコーディングアプリケーションを含む。

20

【0148】

装置500はまた、ディスプレイ518などの1つ以上の出力デバイスも含むことができる。ディスプレイ518は、一例では、タッチ入力を検知するよう動作するタッチ検知素子とディスプレイを組み合わせるタッチ検知ディスプレイであってもよい。ディスプレイ518は、バス512を介してプロセッサ502へ結合され得る。

【0149】

単一のバスとしてここでは表されているが、装置500のバス512は、複数のバスから成ることができる。更に、二次ストレージ514が、装置500の他のコンポーネントへ直接に結合され得るか、あるいは、ネットワークを介してアクセス可能であり、メモリカードなどの単一の集積ユニット又は複数のメモリカードなどの複数のユニットを有することができる。装置500は、このように、多種多様な構成で実施され得る。

30

【0150】

以下では、本発明のより具体的な、非限定的な、例示的な実施形態が記載される。その前に、いくつかの説明が、本開示の理解を助けるよう与えられる。

【0151】

人工ニューラルネットワーク（ANN）又はコネクショニストシステムは、動物の脳を構成する生体ニューラルネットワークから漠然とインスピレーションを受けたコンピューティングシステムである。ANNは、生物学的な脳のニューロンを大まかにモデル化した、人工ニューロンと呼ばれる接続されたユニット又はノードの集合に基づいている。生物学的な脳のシナプスと同様に、各接続は他のニューロンに信号を送信することができる。信号を受信した人工ニューロンはそれを処理し、自身に接続されているニューロンに信号を送ることができる。ANNの実装では、接続における「信号」は実数であり、各ニューロンの出力は、その入力の合計に関する何らかの非線形関数によって計算することができる。この接続はエッジと呼ばれている。ニューロン及びエッジは通常、学習の進行に応じて調整される重みを有する。重みにより、接続における信号の強度が増減する。ニューロンは閾値を有する場合があり、それにより、集合信号がその閾値を超えた場合にのみ信号が送信される。通常、ニューロンはレイヤに集約される。異なるレイヤは、それらの入力

40

50

に対して異なる変換を実行してもよい。信号は、場合により複数回レイヤを通過した後、最初のレイヤ（入力レイヤ）から最後のレイヤ（出力レイヤ）まで伝わる。

【0152】

ANNアプローチの当初の目標は、人間の脳と同じ方法で問題を解決することであった。時間が経つにつれて、特定のタスクを実行することに注意が移り、生物学から逸脱するようになった。ANNは、コンピュータビジョンを含む様々なタスクに使用されている。

【0153】

「畳み込みニューラルネットワーク」（CNN）という名前は、ネットワークが畳み込みと呼ばれる数学的演算を使用していることを示している。畳み込みは、特殊な種類の線形演算である。畳み込みネットワークは、単に、少なくとも1つのレイヤで一般的な行列乗算の代わりに畳み込みを使用するニューラルネットワークである。畳み込みニューラルネットワークは、入力レイヤ及び出力レイヤ、並びに複数の隠れレイヤから成る。入力レイヤは、処理のために入力が入力されるレイヤである。例えば、図6のニューラルネットワークはCNNである。CNNの隠れレイヤは通常、乗算又は他のドット積を使って畳み込む一連の畳み込みレイヤから成る。レイヤの結果は、チャンネルとも呼ばれることがある1つ以上の特徴マップである。一部又は全てのレイヤにサブサンプリングが含まれてもよい。結果として、特徴マップはより小さくなり得る。CNNの活性化関数は、上で既に例示したRELU（Rectified Linear Unit）レイヤ又はGDNレイヤであり、その後、入力及び出力が活性化関数及び最終的な畳み込みによってマスクされるために隠れレイヤと呼ばれる、プーリングレイヤ、全結合レイヤ、及び正規化レイヤなどの追加の畳み込みが続く。レイヤは口語的に畳み込みと呼ばれるが、これは単に慣例によるものである。数学的には、それは技術的にスライディングドット積又は相互相関である。これは、特定のインデックスポイントでの重みの決定方法に影響を与えるため、マトリクスのインデックスにとって重要である。

【0154】

ピクチャ又は画像を処理するためにCNNをプログラミングする場合、入力は形状（画像の数）×（画像の幅）×（画像の高さ）×（画像の深さ）を持つテンソルである。次に、畳み込みレイヤを通過した後、画像は、形状（画像の数）×（特徴マップの幅）×（特徴マップの高さ）×（特徴マップチャンネル）を持つ特徴マップに抽象化される。ニューラルネットワーク内の畳み込みレイヤには次の属性が必要である。幅及び高さ（ハイパーパラメータ）によって定義された畳み込みカーネル。入力チャンネル及び出力チャンネルの数（ハイパーパラメータ）。畳み込みフィルタの深さ（入力チャンネル）は、入力特徴マップのチャンネル数（深さ）と同じである必要がある。

【0155】

これまで、画像認識には従来のマルチレイヤパーセプトロン（MLP）モデルが使用されてきた。しかし、ノード間の完全な接続により、高次元性が問題となり、高解像度の画像ではうまく拡張できなかった。RGBカラーチャンネルを含む1000×1000ピクセルの画像には300万の重みがあり、完全な接続で大規模に効率的に処理するには高すぎる。また、このようなネットワークアーキテクチャはデータの空間構造を考慮しておらず、遠く離れた入力ピクセルを、互いに近いピクセルと同じように扱う。これは、計算的にも意味的にも、画像データ内の参照の局所性を無視している。従って、ニューロンの完全な接続は、空間的に局所的な入力パターンによって支配される画像認識などの目的にとっては無駄である。CNNモデルは、自然画像に存在する強力な空間的局所相関を利用することで、MLPアーキテクチャによってもたらされる課題を軽減する。畳み込みレイヤは、CNNの中核となる構成要素である。このレイヤのパラメータは、ひと組の学習可能なフィルタ（前述のカーネル）から成り、受容野は小さいが、入力ボリュームの深さ全体に広がる。順方向パス中に、各フィルタは入力ボリュームの幅及び高さによって畳み込まれ、フィルタのエントリと入力との間のドット積が計算され、そのフィルタの2次元活性化マップが生成される。その結果、ネットワークは、入力内のある空間位置である特定の種類の特徴を検出したときにアクティブになるフィルタを学習する。

【0156】

全てのフィルタの活性化マップを深さ次元に沿って積み重ねると、畳み込みレイヤの完全な出力ボリュームが形成される。従って、出力ボリューム内の全てのエントリは、入力内の小さな領域を調べ、同じ活性化マップ内のニューロンとパラメータを共有するニューロンの出力として解釈することもできる。機能マップ又は活性化マップは、所与のフィルタの出力活性化である。機能マップと活性化とは同じ意味である。いくつかの論文では、それは、画像の種々の部分の活性化に対応するマッピングであるため、活性化マップと呼ばれたり、画像内で特定の種類の特徴が見つかる場所のマッピングでもあるため、特徴マップとも呼ばれたりする。活性化が高いということは、特定の機能が見つかったことを意味する。

10

【0157】

CNNのもう1つの重要な概念は、非線形ダウンサンプリングの形式であるプーリングである。プーリングを実装する非線形関数はいくつかあるが、その中で最も一般的なのは最大値プーリングである。それは入力画像をひと組の重なり合わない長方形にパーティション化し、そのようなサブ領域ごとに最大値を出力する。直感的には、特徴の正確な位置は、他の特徴と比較した大まかな位置ほど重要ではない。これは、畳み込みニューラルネットワークでのプーリングの使用の背後にある考え方である。プーリングレイヤは、表現の空間サイズを段階的に縮小し、パラメータの数、メモリフットプリント、及びネットワーク内の計算量を削減し、それによってオーバーフィッティングを制御する役割も果たす。CNNアーキテクチャでは、連続した畳み込みレイヤの間にプーリングレイヤを周期的に挿入するのが一般的である。プーリング操作は別の形式の並進不変をもたらす。

20

【0158】

上記ReLUは、Rectified Linear Unitの略称であり、非飽和活性化関数を適用する。それは、負の値をゼロに設定することで、活性化マップから負の値を有効に除く。それは、畳み込みレイヤの受容野に影響を与えることなく、決定関数及びネットワーク全体の非線形特性を高める。非線形性を高めるために、例えば、飽和双曲線正接及びシグモイド関数などの他の関数も使用される。ReLUは、一般化の精度に重大な影響を与えることなく、ニューラルネットワークを数倍高速にトレーニングできるため、他の関数よりもしばしば好まれる。

【0159】

いくつかの畳み込みレイヤ及び最大値プーリングレイヤの後、ニューラルネットワークの高度な推論が全結合レイヤを介して実行される。通常の（非畳み込み）人工ニューラルネットワークに見られるように、全結合レイヤ内のニューロンは、前のレイヤの全ての活性化と接続している。従って、それらの活性化は、行列乗算の後にバイアスオフセット（学習又は固定バイアス項のベクトル加算）を行うアフィン変換として計算することができる。

30

【0160】

オートエンコーダは、教師なし方式で効率的なデータコーディングを学習するために使用される人工ニューラルネットワークの一種である。オートエンコーダの目的は、信号の「ノイズ」を無視するようにネットワークをトレーニングすることにより、通常は次元削減のために、データの組の表現（符号化）を学習することである。削減側とともに、再構築側も学習され、そこでは、オートエンコーダは、その元の入力にできるだけ近い表現を削減された符号化から生成しようとする。これがその名前の由来である。

40

【0161】

ピクチャサイズ：ピクチャの幅若しくは高さ又は幅と高さの組を指す。画像の幅及び高さは、通常、ルーマサンプルの数で測定される。

【0162】

ダウンサンプリング：ダウンサンプリングは、離散入力信号のサンプリングレート（サンプリングインターバル）が低減されるプロセスである。例えば、入力信号が、高さ h 及び幅 w （又は、以下、 H 及び W も同様）のサイズを有する画像であり、ダウンサンプリン

50

グの出力が高さ h_2 及び幅 w_2 である場合に、次のうちの少なくとも1つが当てはまる：

- $h_2 < h$
- $w_2 < w$

【0163】

1つの実施例では、ダウンサンプリングは、各 m 番目のサンプルのみを保持し、入力信号（本発明の文脈中、基本的にピクチャである）残りを破棄するものとして実施され得る。

【0164】

アップサンプリング：アップサンプリングは、離散入力信号のサンプリングレート（サンプリングインターバル）が増大するプロセスである。例えば、入力画像が h 及び w （又は、以下、 H 及び W も同様）のサイズを有し、アップサンプリングの出力が h_2 及び w_2 である場合に、次のうちの少なくとも1つが当てはまる：

- $h < h_2$
- $w < w_2$

【0165】

リサンプリング：ダウンサンプリングプロセス及びアップサンプリングプロセスは両方ともリサンプリングの例である。リサンプリングは、入力信号のサンプリングレート（サンプリングインターバル）が変更されるプロセスである。

【0166】

補間フィルタリング：アップサンプリング又はダウンサンプリングプロセス中に、フィルタリングは、リサンプリングされた信号の精度を控除させかつエイリアシングの影響を軽減するために適用することができる。補間フィルタは、通常、リサンプリング位置の周りのサンプル位置でのサンプル値の重み付けされた結合を含む。それは：

$$f(x_r, y_r) = \sum s(x, y) C(k)$$

として実施され得る。ここで、 $f()$ はリサンプリングされた信号であり、 (x_r, y_r) はリサンプリング座標であり、 $C(k)$ は補間フィルタ係数であり、 $s(x, y)$ は入力信号である。加算演算が、 (x_r, y_r) の近傍にある (x, y) に対して行われる。

【0167】

クロッピング：デジタル画像の外縁をトリミングする。クロッピングは、画像を（サンプルの数において）より小さくするために、及び／又は画像のアスペクト比（幅に対する長さ）を変えるために使用され得る。

【0168】

パディング：パディングは、画像の境界で新しいサンプルを生成することによって、入力画像（又は画像）のサイズを増大させることを指す。これは、例えば、事前定義されたサンプル値を使用するか、又は入力画像内の位置のサンプル値を使用するかのどちらかによって、行われ得る。

【0169】

リサイジング：リサイジングは、入力画像のサイズが変更される一般用語である。それは、パディング又はクロッピングの方法の1つを使用して行われてもよい。それは、補間を使用するリサイジング操作によって行われ得る。以下で、リサイジングは、リスケージングとも呼ばれ得る。

【0170】

整数除算：整数除算は、小数部（余り）が捨てられる除算である。

【0171】

畳み込み：畳み込みは、次の一般式によって与えられる。以下で、 $f()$ は、入力信号として定義することができ、 $g()$ は、フィルタとして定義することができる。

【数119】

10

20

30

40

50

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f[m]g[n-m]$$

【0172】

ダウンサンプリングレイヤ：入力の次元の少なくとも1つの削減をもたらすニューラルネットワークのレイヤなどの処理レイヤ。一般に、入力は3以上の次元を有する可能性があり、次元はチャンネルの数、幅及び高さを含み得る。しかし、本開示は、そのような信号に制限されない。むしろ、1つ又は2つの次元を有し得る（例えば、オーディオ信号又は複数のチャンネルを有するオーディオ信号）信号が処理されてもよい。ダウンサンプリングレイヤは、通常、幅及び高さの次元の削減を指す。それは、畳み込み、平均化、最大値プーリングなどの操作で実施することができる。また、他のダウンサンプリング方法が可能であり、本発明がこれに関して制限されない。

10

【0173】

アップサンプリングレイヤ：入力の次元の1つの増加をもたらすニューラルネットワークのレイヤなどの処理レイヤ。一般に、入力は3以上の次元を有する可能性があり、次元はチャンネルの数、幅及び高さを含み得る。アップサンプリングレイヤは、通常、幅及び高さの次元の増加を指す。それは、逆畳み込み、複製などの操作で実施することができる。また、他のアップサンプリング方法が可能であり、本発明はこれに関して制限されない。

20

【0174】

いくつかのディープラーニングに基づいた画像及びビデオ圧縮アルゴリズムは、変分オートエンコーダフレームワーク（VAE）、例えば、<https://arxiv.org/abs/2003.02012>から入手することができるG-VAE: A Continuously Variable Rate Deep Image Compression Framework, (Ze Cui, Jing Wang, Bo Bai, Tiansheng Guo, Yihui Feng)に従う。

【0175】

VAEフレームワークは、非線形変換コーディングモデルとして数えられ得る。

【0176】

変換プロセスは、主に、4つの部分に分割することができる。図4はVAEフレームワークを例示する。図4で、エンコーダ601は、入力画像 x を、関数 $y = f(x)$ により、潜在表現（ y で表記される）にマッピングする。この潜在表現は、以下で「潜在空間」内の点又は部分とも呼ばれ得る。関数 $f()$ は、入力信号 x を圧縮しやすい表現 y に変換する変換関数である。量子化器602は、潜在表現 y を、

30

【数120】

$$\hat{y} = Q(y)$$

40

によって、（離散）値を有する量子化された潜在表現

【数121】

$$\hat{y}$$

に変換し、 Q は量子化関数を表す。エントロピモデル、又はハイパーエンコーダ／デコーダ（ハイパープライアとしても知られる）603は、可逆エントロピソースコーディング

50

により達成可能な最小レートを得るよう、量子化された潜在表現
【数 1 2 2】

$$\hat{y}$$

の分布を推定する。

【0 1 7 7】

潜在空間は、類似したデータ点が潜在空間において互いにより近くにある圧縮されたデータの表現として理解することができる。潜在空間は、データ特徴を学習すること及び解析のためにデータのより簡単な表現を見つけることにとって有用である。ハイパープライア 3 の量子化された潜在表現 T、

【数 1 2 3】

$$\hat{y}$$

及びサイド情報

【数 1 2 4】

$$\hat{z}$$

は、算術コーディング (A E) を用いてビットストリーム 2 に含まれる (二値化される)。

【0 1 7 8】

更に、デコーダ 6 0 4 が設けられ、量子化された潜在表現を、再構成された画像

【数 1 2 5】

$$\hat{x}, \hat{x} = g(\hat{y})$$

に変換する。信号

【数 1 2 6】

$$\hat{x}$$

は、入力画像 x の推定である。x は可能な限り

【数 1 2 7】

$$\hat{x}$$

に近いことが望ましく、換言すれば、再構成品質は可能な限り高い。しかし、

【数 1 2 8】

$\hat{x}$

と x との間の類似度が高ければ高いほど、伝送される必要があるサイド情報の量はますます多くなる。サイド情報は、図 4 に示されるビットストリーム 1 及びビットストリーム 2 を含み、これらはエンコーダによって生成されて、デコーダへ送信される。通常、サイド情報の量が多ければ多いほど、再構成品質はますます高くなる。しかし、サイド情報の量が多いとは、圧縮比が低いことを意味する。従って、図 4 で記載されるシステムの 1 つの目的は、再構成品質と、ビットストリームで運ばれるサイド情報の量とのバランスをとることである。

10

【0 1 7 9】

図 4 で、コンポーネント A E 6 0 5 は、算術符号化モジュールであり、量子化された潜在表現

【数 1 2 9】

$\hat{y}$

20

及びサイド情報

【数 1 3 0】

$\hat{z}$

のサンプルを二値表現ビットストリーム 1 に変換する。

【数 1 3 1】

$\hat{y}$ 及び $\hat{z}$

30

のサンプルは、例えば、整数又は浮動小数点数を有してよい。算術符号化モジュールの 1 つの目的は、サンプル値を（二値化のプロセスにより）二進数の列（次いで、符号化された画像に対応する更なる部分又は更なるサイド情報を含み得るビットストリームに含まれる）に変換することである。

【0 1 8 0】

算術復号化（A D）6 0 6 は、二値化プロセスを元に戻すプロセスであり、二進数はサンプル値に逆変換される。算術復号化は、算術復号化モジュール 6 0 6 によって提供される。

40

【0 1 8 1】

本開示はこの特定のフレームワークに制限されないことが留意されるべきである。更に、本開示は、画像又はビデオ圧縮に制限されず、オブジェクト検出、画像生成、及び認識システムなどにも適用することができる。

【0 1 8 2】

図 4 には、互いに連結された 2 つのサブネットワークがある。本文脈中のサブネットワークは、ネットワーク全体の部分の間の論理的な分割である。例えば、図 4 で、モジュール 6 0 1、6 0 2、6 0 4、6 0 6、及び 6 0 6 は、「エンコーダ／デコーダ」サブネットワークと呼ばれる。「エンコーダ／デコーダ」サブネットワークは、第 1 ビットストリ

50

ーム“ビットストリーム1”の符号化（生成）及び復号化（パージング）に關与する。図4の第2サブネットワークは、モジュール603、608、609、610、及び607を含み、「ハイパーエンコーダ／デコーダ」サブネットワークと呼ばれる。第2サブネットワークは、第2ビットストリーム“ビットストリーム2”を生成することに関与する。2つのサブネットワークの目的は異なる。第1サブネットワークは：

- 入力画像 x のその潜在表現 y (x よりも圧縮するのが容易である) への変換 601,
- 潜在表現 y を量子化された潜在表現

【数132】

$$\hat{y}$$

10

に量子化すること 602,

- ビットストリーム“ビットストリーム1”を得るよう、算術符号化モジュール605によってAEを用いて、量子化された潜在表現

【数133】

$$\hat{y}$$

20

を圧縮すること,

- 算術復号化モジュール606を用いてADによりビットストリーム1をパースすること, 及び

- パースされたデータを用いて、再構成された画像

【数134】

$$(\hat{x})$$

を再構成すること 604

30

に關与する。第2サブネットワークの目的は、第1サブネットワークによるビットストリーム1の圧縮がより効率的であるように、“ビットストリーム1”のサンプルの統計的性質（例えば、ビットストリーム1のサンプル間の平均値、分散、及び相関）を得ることである。第2サブネットワークは、上記の情報（例えば、ビットストリーム1のサンプル間の平均値、分散、及び相関）を含む第2ビットストリーム“ビットストリーム2”を生成する。

【0183】

第2サブネットワークは、量子化された潜在表現

【数135】

$$\hat{y}$$

40

のサイド情報 z への変換 603、サイド情報 z を量子化されたサイド情報

【数136】

$$\hat{z}$$

に量子化すること、量子化されたサイド情報

50

【数 1 3 7】

$$\hat{z}$$

をビットストリーム 2 に符号化すること（例えば、二値化すること）6 0 9 を含む符号化部分を含む。この例では、二値化は算術符号化（A E）によって行われる。第 2 サブネットワークの復号化部分は、入力ビットストリーム 2 を復号された量子化されたサイド情報【数 1 3 8】

10

$$\hat{z}'$$

に変換する算術復号化（A D）6 1 0 を含む。算術符号化及び復号化操作は可逆圧縮方法であるから、
【数 1 3 9】

$$\hat{z}'$$

20

は、
【数 1 4 0】

$$\hat{z}$$

と同じであることができる。復号された量子化されたサイド情報
【数 1 4 1】

$$\hat{z}'$$

30

は、次いで、復号されたサイド情報
【数 1 4 2】

$$\hat{y}'$$

に変換 6 0 7 される。
【数 1 4 3】

40

$$\hat{y}'$$

は、
【数 1 4 4】

$$\hat{y}$$

50

の統計的性質（例えば、
【数 1 4 5】

$$\hat{y}$$

のサンプルの平均値、又はサンプル値の分散、など）を表す。復号された潜在表現
【数 1 4 6】

$$\hat{y}'$$

10

は、次いで、
【数 1 4 7】

$$\hat{y}$$

の確率モデルを制御するよう上記の算術エンコーダ 6 0 5 及び算術デコーダ 6 0 6 へ供給
される。

20

【0 1 8 4】

図 4 は、VAE（変分オートエンコーダ）の一例を記載しており、その詳細は、異なる
実施において異なる可能性がある。例えば、具体的な実施では、追加のコンポーネントが
、ビットストリーム 1 のサンプルの統計的性質より効率的に得るために存在してもよい。
1 つのかような実施において、コンテキストモデラが存在してもよく、ビットストリーム
1 の相互相関情報を抽出することを目標とする。第 2 サブネットワークによって提供され
る統計情報は、AE（算術エンコーダ）6 0 5 及びAD（算術デコーダ）6 0 6 のコンポ
ーネントによって使用されてもよい。

【0 1 8 5】

図 4 は、単一の図でエンコーダ及びデコーダを表している。当業者には明らかなように
、エンコーダ及びデコーダは、相互に異なるデバイスに埋め込まれてもよく、埋め込まれ
ていることが非常に多い。

30

【0 1 8 6】

図 7 は、VAE フレームワークのエンコーダコンポーネントを表し、分離して、図 8 は
、VAE フレームワークのデコーダコンポーネントを表す。入力として、エンコーダは、
いくつかの実施形態に従って、ピクチャを受け取る。入力されたピクチャは、カラーチャ
ネル又は他の種類のチャンネル、例えば、デプスチャンネル又は動き情報チャンネル、などのよ
うな 1 つ以上のチャンネルを含んでよい。エンコーダ（図 7 に図示）の出力はビットストリ
ーム 1 及びビットストリーム 2 である。ビットストリーム 1 は、エンコーダの第 1 サブネ
ットワークの出力であり、ビットストリーム 2 は、エンコーダの第 2 サブネットワークの
出力である。

40

【0 1 8 7】

同様に、図 8 では、2 つのビットストリーム、つまり、ビットストリーム 1 及びビット
ストリーム 2 が入力として受け取られ、再構成された（復号された）画像である

【数 1 4 8】

$$\hat{z}$$

が出力として生成される。

50

【0188】

上述されたように、VAEは、異なる動作を実行する異なる論理的ユニットに分けられ得る。これは図7及び図8で例示されており、図7は、ビデオなどの信号の符号化に關与するコンポーネント及び供給された符号化された情報を示している。この符号化された情報は、次に、例えば復号化のために、図8に示されるデコーダコンポーネントによって受信される。符号9xx及び10xxで示されるエンコーダ及びデコーダのコンポーネントは、その機能において、図4において上で参照されて符号6xxで示されているコンポーネントに対応し得ることに留意されたい。

【0189】

具体的に、図7から分かるように、エンコーダは、入力xを、次に量子化器902へ供給されるyに変換するエンコーダ901を有する。量子化器902は、情報を算術符号化モジュール905及びハイパーエンコーダ903へ供給する。ハイパーエンコーダ903は、既に上で議論されたビットストリーム2をハイパーデコーダ907へ供給し、ハイパーデコーダ907は次いで、情報を算術符号化モジュール905へ伝える。

【0190】

符号化は、図19に關連して以下で更に詳細に説明されるように、畳み込みを利用することができる。

【0191】

算術符号化モジュールの出力はビットストリーム1である。ビットストリーム1及びビットストリーム2は信号の符号化の出力であり、それらは次いで、復号化プロセスへ供給（伝送）される。

【0192】

ユニット901は「エンコーダ」と呼ばれるが、図7に記載されている完全なサブネットワークを「エンコーダ」と呼ぶことも可能である。一般に、符号化のプロセスは、入力を符号化（例えば、圧縮）された出力に変換するユニット（モジュール）を意味する。図7からは、ユニット901が入力xを、xの圧縮された表現であるyに変換するというこ
とで、それがサブネットワーク全体の中核と実際に見なされ得ることが分かる。エンコーダ901の圧縮は、例えば、ニューラルネットワーク、又は一般に、1つ以上のレイヤを備えた任意の処理ネットワークによって、達成されてもよい。そのようなネットワークで、圧縮は、入力チャネルのサイズ及び／又は数を低減するダウンサンプリングを含むカスケード接続された処理によって実行されてもよい。よって、エンコーダは、例えば、ニューラルネットワーク（NN）に基づいたエンコーダなどと呼ばれることがある。

【0193】

図中の残りの部分（量子化ユニット、ハイパーエンコーダ、ハイパーデコーダ、算術エンコーダ／デコーダ）は全て、符号化プロセスの効率を向上させたり、又は圧縮された出力yをビットの連続（ビットストリーム）に変換することに関与したりする部分である。量子化は、不可逆圧縮によってNNエンコーダ901の出力を更に圧縮するよう提供されてよい。AE905は、AE905を構成するために使用されるハイパーエンコーダ903及びハイパーデコーダ907と組み合わせて、量子化された信号を可逆圧縮によって更に圧縮し得る二値化を実行してよい。従って、図7のサブネットワーク全体を「エンコーダ」と呼ぶことも可能である。

【0194】

ディープラーニング（DL）に基づいた画像／ビデオ圧縮システムの大部分は、信号を二進数（ビット）に変換する前に信号の次元を削減する。VAEフレームでは、例えば、非線形変換であるエンコーダは、入力画像xをyにマッピングし、yは、xよりも小さい幅及び高さを有する。yはより小さい幅及び高さ、よって、より小さいサイズを有するので、信号の次元（のサイズ）は削減され、従って、信号yを圧縮する方が容易である。一般に、エンコーダは、両方の（又は一般に全ての）次元のサイズを削減する必要があるが必ずしもないことが留意されるべきである。むしろ、いくつかの例示的な実施は、1つの次元（又は一般的にそのサブセット）でのみサイズを削減するエンコーダを提供し得る。

10

20

30

40

50

【0195】

圧縮の一般的な原理は、図5で例示されている。エンコーダの出力であり、デコーダの入力である潜在空間は、圧縮されたデータを表す。潜在空間のサイズは、入力信号サイズよりもずっと小さくなることに留意されるべきである。ここで、サイズという用語は、解像度を、例えば、エンコーダによって出力される特徴マップのサンプルの数を指すことができる。解像度は、各次元ごとのサンプルの数の積として与えられ得る（例えば、入力画像の又は特徴マップの幅×高さ×チャンネルの数）。

【0196】

入力信号のサイズの低減は、ディープラーニングに基づいたエンコーダ及びデコーダを表す図5で例示されている。図5で、入力画像 x は、エンコーダの入力である入力データに対応する。変換された信号 y は、入力信号よりも少なくとも1つの次元で小さい次元性又はサイズを有する潜在空間に対応する、円の各列は、エンコーダ又はデコーダの処理チェーン内のレイヤを表す。各レイヤにおける円の数は、そのレイヤでの信号のサイズ又は次元性を示す。

10

【0197】

図5から分かるように、符号化操作は、入力信号のサイズの低減に対応し、一方、復号化操作は、画像の元のサイズの再構成に対応する。

【0198】

信号サイズの低減のための方法の1つはダウンサンプリングである。ダウンサンプリングは、入力信号のサンプリングレートが低減されるプロセスである。例えば、入力画像が h 及び w のサイズを有し、ダウンサンプリングの出力が h_2 及び w_2 である場合に、次のうちの少なくとも1つが当てはまる：

20

● $h_2 < h$

● $w_2 < w$

【0199】

信号サイズの低減は、通常は、一度にではなく処理レイヤのチェーンに沿って段階的に起こる。例えば、入力画像 x が（高さ及び幅を示す） h 及び w の次元（又は次元のサイズ）をし、潜在空間 y が次元 $1/16$ 及び $w/16$ を有する場合に、サイズの低減は、符号化中に4つのレイヤで起こる可能性があり、各レイヤは、各次元で信号のサイズを2分の1にする。

30

【0200】

いくつかのディープラーニングに基づいたビデオ／画像圧縮方法は、複数のダウンサンプリングレイヤを用いる。一例として、VAEフレームワーク、図6は、801から806でマークされた6つのダウンサンプリングレイヤを利用する。ダウンサンプリングを含むレイヤは、レイヤの説明において下向き矢印で示されている。レイヤの説明「Conv $N \times 5 \times 5 / 2 \downarrow$ 」は、レイヤが N 個のチャンネルを有する畳み込みレイヤであることを意味し、畳み込みカーネルはサイズが 5×5 である。述べられているように、 $2 \downarrow$ は、係数2によるダウンサンプリングがこのレイヤで実行されることを意味する。係数2によるダウンサンプリングは、入力信号の次元の1つが出力で半分に削減されることをもたらす。図6で、 $2 \downarrow$ は、入力画像の幅及び高さの両方が係数2で低減されることを示す。6つのダウンサンプリングレイヤがあるので、入力画像814（ x でも表記される）の幅及び高さが w 及び h によって与えられる場合に、出力信号

40

【数149】

$$\hat{z}$$

813は、夫々、 $w/64$ 及び $h/64$ に等しい幅及び高さを有する。AE及びADによって示されるモジュールは、図4、図7、及び図8に関して既に上で説明された算術エンコーダ及び算術デコーダである。算術エンコーダ及び算術デコーダは、エントロピコーデ

50

ィングの具体的な実施である。A E 及び A D（コンポーネント 8 1 3 及び 8 1 5 の部分として）は、他のエントロピコーディング手段によって置換することができる。情報理論では、エントロピ符号化は、シンボルの値を二値表現に変換するために使用される可逆データ圧縮スキームであり、これは復帰可能なプロセスである。また、図中の「Q」は、図 4 に関して上でも言及されており、「量子化」のセクションにおいて上で更に説明されている量子化操作に対応する。また、コンポーネント 8 1 3 及び 8 1 5 の部分としての量子化操作及び対応する量子化ユニットは、必ずしも存在するわけではなく、かつ／あるいは、他のユニットで置換することができる。

【0201】

図 6 には、アップサンプリングレイヤ 8 0 7 から 8 1 2 を含むデコーダも示されている。更なるレイヤ 8 2 0 が、入力の処理順においてアップサンプリングレイヤ 8 1 1 及び 8 1 0 の間に設けられており、畳み込みレイヤとして実装されるが、受信した入力にアップサンプリングを施さない。対応する畳み込みレイヤ 8 3 0 もデコーダに対して示されている。かようなレイヤは、入力サイズを変更しないが特定の特性を変更する操作を入力に対して実行するために N N に設けられ得る。

【0202】

デコーダを通るビットストリーム 2 の処理順で見たとき、アップサンプリングレイヤは逆の順序で、つまり、アップサンプリングレイヤ 8 1 2 からアップサンプリングレイヤ 8 0 7 まで実行される。各アップサンプリングレイヤは、ここでは、 $\uparrow$ によって示される 2 のアップサンプリング比によるアップサンプリング提供するように示されている。当然ながら、全てのアップサンプリングレイヤが同じアップサンプリング比を持つとは限らず、3、4、8 などの他のアップサンプリング比も使用されてよい。レイヤ 8 0 7 から 8 1 2 は、畳み込みレイヤ（conv）として実装される。具体的に、それらは、エンコーダの操作とは逆である操作を入力に提供するように意図され得るので、アップサンプリングレイヤは、受け取った入力に逆畳み込み操作を適用することができ、それにより、入力のサイズは、アップサンプリング比に対応する係数だけ増大される。しかし、本開示は、概して、逆畳み込みに制限されず、アップサンプリングは、如何なる他の方法でも、2 つの隣接したサンプル間の双線形補間や最近傍サンプルコピーなどによって実行されてもよい。

【0203】

第 1 サブネットワークにおいて、いくつかの畳み込みレイヤ（8 0 1 から 8 0 3）の後には、エンコーダ側では一般化分割正規化（GDN）が、デコーダ側では逆 GDN（IGDN）が続いている。第 2 サブネットワークにおいて、適用される活性化関数は ReLU である。本開示はそのような実施に制限されず、一般に、他の活性化関数が GDN 又は ReLU の代わりに使用されてもよいことが留意されるべきである。

【0204】

画像及びビデオ圧縮システムは、一般に、任意の入力画像サイズを処理することができない。その理由は、圧縮システム内の処理ユニットの一部（例えば、変換ユニット、又は動き補償ユニット）が最小単位で動作し、入力画像サイズが最小処理単位の整数倍でない場合には、画像を処理することは不可能であるからである。

【0205】

一例として、HEVC は、予測残差をコーディングするよう 4×4 、 8×8 、 16×16 、及び 32×32 の 4 つの変換ユニット（TU）を規定している。最小変換ユニットは 4×4 であるから、HEVC エンコーダ及びデコーダを用いて、 3×3 のサイズを有する入力画像を処理することは不可能である。同様に、画像サイズが 1 つの次元で 4 の倍数でない場合に、画像を、有効な変換ユニット（ 4×4 、 8×8 、 16×16 、及び 32×32 ）によって処理可能であるサイズにパーティション化することは不可能であるから、画像を処理することはやはり不可能である。従って、入力画像が 8×8 である最小コーディング単位サイズの倍数でなければならないことが、HEVC 標準の要件である。さもなければ、入力画像は HEVC によって圧縮可能でない。同様の要件は、他のコーデックによっても課されている。既存のハードウェア又はソフトウェアを利用するために、あるいは

10

20

30

40

50

、既存のコーデックの一部又は一部の相互運用制を維持するために、そのような制限を維持することが望ましい場合がある。しかし、本開示は、如何なる特定の変換ブロックサイズにも制限されない。

【0206】

いくつかのDNN (deep nueral network) 又はNNに基づいた画像及びビデオ圧縮システムは、複数のダウンサンプリングレイヤを利用する。図6では、例えば、4つのダウンサンプリングレイヤが第1サブネットワーク (レイヤ801から804) に含まれ、2つの追加のダウンサンプリングレイヤが第2サブネットワーク (レイヤ805から806) に含まれている。従って、入力画像のサイズがw及びh (幅及び高さを示す) によって夫々与えられる場合に、第1サブネットワーク出力は $w/16$ 及び $h/16$ であり、第2サブネットワークの出力は $w/64$ 及び $h/64$ によって与えられる。

10

【0207】

ディープニューラルネットワークの「ディープ」という用語は、通常、入力に順に適用される処理レイヤの数を指す。レイヤの数が多いとき、ニューラルネットワークはディープニューラルネットワークと呼ばれるが、ネットワークがディープネットワークと呼ばれるべき明りょうな記載又はガイダンスはない。従って、本願の目的のために、DNNとNNとの間には大きな違いはない。DNNは、1よりも多いレイヤを有するNNを指すことができる。

【0208】

ダウンサンプリング中、例えば、入力に畳み込みが適用される場合、場合によっては、符号化されたピクチャの小数 (最終的な) サイズが取得されることがある。このような小数サイズは、ニューラルネットワークの後続のレイヤによって又はデコーダによって合理的に処理されない可能性がある。

20

【0209】

別の言い方をすると、一部のダウンサンプリング操作 (畳み込みなど) は、ニューラルネットワークの特定のレイヤへの入力のサイズが特定の条件を満たすことを (例えば、設計などによって) 期待する場合があるため、ダウンサンプリングを実行するニューラルネットワークのレイヤ内で実行されるか、又はダウンサンプリングに続く演算は、依然として明確に定義された数学的演算である。例えば、少なくとも1つの次元で入力のサイズを比率 r だけ低減するダウンサンプリング比

30

【数150】

$$r > 1, r \in \mathbb{N}$$

を持つダウンサンプリングレイヤの場合、入力がこの次元でダウンサンプリング比 r の整数倍であるサイズを有するならば、妥当な出力が得られる。 r によるダウンサンプリングは、1つの次元 (例えば、幅) 又はより多くの次元 (例えば、幅及び高さ) での入力サンプルの数が出力サンプルの数を求めるために2で除されることを意味する。

【0210】

40

数値例を挙げると、レイヤのダウンサンプリング比は4である。第1の入力は、ダウンサンプリングが適用される次元でサイズ512を有する。 $128 \times 4 = 512$ であるから、512は4の整数倍である。従って、入力の処理はダウンサンプリングレイヤによって実行され、妥当な出力が得らる。第2の入力は、ダウンサンプリングが適用される次元で513のサイズを持つ可能性がある。513は4の整数倍ではないため、この入力はダウンサンプリングレイヤ又は後続のダウンサンプリングレイヤによって、それらが、例えば、設計によって、特定の (例えば、512の) 入力サイズを期待している場合には、合理的に処理できない。これを考慮して、入力のサイズが常に同じではない場合でも、入力がニューラルネットワークの各レイヤによって合理的な方法 (事前定義されたレイヤ入力サイズに従って) で処理され得ることを保証するために、リスケーリングは、ニューラルネ

50

ットワークによって入力を処理する前に適用されてもよい。このリスケーリングは、ニューラルネットワークの全てのダウンサンプリングレイヤに関して上記の条件を満たすように、ニューラルネットワークへの（例えば、ニューラルネットワークの入力レイヤへの）入力の実際のサイズを変更又は適応させることを含む。このリスケーリングは、ダウンサンプリングが適用される次元での入力のサイズを、

【数 1 5 1】

$$S = K \prod_i r_i$$

10

になるように増減することによって行われ、ここで、 r_i は、ダウンサンプリングレイヤのダウンサンプリング比であり、 K は、ゼロよりも大きい整数である。換言すれば、ダウンサンプリング方向における入力ピクチャ（信号）の入力サイズは、ダウンサンプリング方向（次元）におけるネットワーク処理チェーンで入力ピクチャ（信号）に適用される全てのダウンサンプリング比の積の整数倍であるよう適応される。

【0 2 1 1】

それによって、ニューラルネットワークへの入力サイズは、例えば、レイヤの事前定義された入力サイズ設定に従って、各レイヤがその各々の入力を処理することができることを保証するサイズを有する。

20

【0 2 1 2】

そのようなリスケーリングを提供することによって、しかしながら、符号化されるべきピクチャのサイズの低減には制限があり、相応して、例えば、符号化された情報を再構成するために、デコーダに供給され得る符号化されたピクチャのサイズにも、下限がある。更に、これまで提供されたアプローチによれば、相当量のエントロピが（サイズをリスケーリングによって増大させる場合に）ビットストリームに加えられる場合があり、あるいは、相当量の情報損失が（ビットストリームのサイズをリスケーリングによって増大させる場合に）起こる可能性がある。どちらも、復号化後のビットストリームの品質に悪影響を与える可能性がある。

【0 2 1 3】

30

従って、高品質の符号化／復号されたビットストリーム及びそれらが表すデータを得ながら、同時に、低減されたサイズで符号化されたビットストリームを供給することは困難である。

【0 2 1 4】

ネットワーク内のレイヤの出力サイズは小数であることができない（サンプルの整数個の行及び列が必要ある）から、入力画像サイズには制限がある。図 6 では、信頼できる処理を保証するために、入力画像サイズは、水平及び垂直の両方の方向で 64 の整数倍である。さもなければ、第 2 サブネットワークの出力は整数にならない。

【0 2 1 5】

40

この問題を解決するために、入力画像にゼロをパディングしてそれを各方向で 64 サンプルの倍数にする方法を使用することが可能である。この解決法に従って、入力画像サイズは、次の量：

【数 1 5 2】

50

$$w_{diff} = \text{Int} \left(\frac{w + 63}{64} \right) \cdot 64 - w$$

$$h_{diff} = \text{Int} \left(\frac{h + 63}{64} \right) \cdot 64 - h$$

だけ幅及び高さを拡張され得る。ここで、「Int」は整数変換である。整数変換は、第1の値a及び第2の値bの商を計算し、次いで、全ての小数部の桁を無視する出力を供給して、整数だけであるようにすることができる。新たに生成されたサンプル値は0にセットされ得る。

【0216】

上記の問題を解決する他の可能性は、入力画像をクロッピングし、つまり、入力画像の端部からサンプルの行及び列を捨て、入力画像サイズを64サンプルの倍数にすることである。クロッピングされる必要がある行及びサンプルの最低限の量は、次の通りに計算され得る：

【数153】

$$w_{diff} = w - \text{Int} \left(\frac{w}{64} \right) \cdot 64$$

$$h_{diff} = h - \text{Int} \left(\frac{h}{64} \right) \cdot 64$$

ここで、 w_{diff} 及び h_{diff} は、夫々、画像の両側から捨てられる必要があるサンプル行及び列の量に対応する。

【0217】

上記により、水平（ h_{new} ）及び垂直（ w_{new} ）次元での入力画像の新しいサイズは、次の通りである：

パディングの場合に：

- $h_{new} = h + h_{diff}$
- $w_{new} = w + w_{diff}$

クロッピングの場合に：

- $h_{new} = h - h_{diff}$
- $w_{new} = w - w_{diff}$

これは、図10及び図11でも示されている。図10には、エンコーダ及びデコーダ（まとめて1200で表記されている）が多数のダウンサンプリング及びアップサンプリングレイヤを有し得ることが示されている。各レイヤは、係数2によるダウンサンプリング又は係数2によるアップサンプリングを適用する。更に、エンコーダ及びデコーダは、エンコーダ側では一般化分割正規化（GDN）1201のような、デコーダ側では逆GDN（IGDN）1202によって、更なるコンポーネントを有することができる。更に、エンコーダ及びデコーダはいずれも、1つ以上のReLU、具体的には、leaky ReLU 1203を有してもよい。また、エンコーダでは因数分解エントロピモデル1205が、デコーダではガウスエントロピモデル1206が設けられ得る。更に、複数の畳み込みマスク1204が設けられてもよい。更に、エンコーダは、図10及び図11の実施形態において、汎用量子化器（UnivQuan）1207を含み、デコーダはアテンションモジュール1208を有する。参照を簡単にするために、機能的に対応するコンポーネ

10

20

30

40

50

ントは、図 1 1 で対応する符号を有する。

【 0 2 1 8 】

ダウンサンプリング操作及びストライドの総数は、入力チャンネルサイズ、つまり、ニューラルネットワークへの入力のサイズに対する条件を定義する。

【 0 2 1 9 】

ここで、入力チャンネルサイズが $64 = 2 \times 2 \times 2 \times 2 \times 2 \times 2$ の整数倍である場合に、チャンネルサイズは、全ての一連のダウンサンプリング操作の後、整数のままである。対応するアップサンプリング操作をアップサンプリング中にデコーダで適用することによって、及びアップサンプリングレイヤを通る入力の処理の終わりに同じリスケールを適用することによって、出力サイズは、再びエンコーダでの入力サイズと同じである。それによって、元の入力の信頼できる再構成が得られる。

10

【 0 2 2 0 】

図 1 1 において、図 1 0 で説明されているもののより一般的な例が示されている。この例も、まとめて 1 3 0 0 で表記されているエンコーダ及びデコーダを示す。m 個のダウンサンプリングレイヤ（及び対応するアップサンプリングレイヤ）は、ダウンサンプリング比 s_i 及び対応するアップサンプリング比を有する。ここで、入力チャンネルサイズが

【数 1 5 4】

$$S = \prod_{i=1}^m s_i$$

20

の整数倍である場合に、チャンネルサイズは、全ての m 個の一連の（連続した又は後続の又はカスケード接続された）ダウンサンプリング操作の後に整数のままである。エンコーダでニューラルネットワークによって入力を処理する前のその対応するリスケールは、上記の式が満足されることを保証する。換言すれば、ダウンサンプリング方向における入力チャンネルサイズは、（サブ）ネットワークの各々の m 個のダウンサンプリングレイヤによって入力に適用される全てのダウンサンプリング比の積である。上述されたように入力のサイズを変更するこのモードは、依然としていくつかの欠点を有する場合がある。

30

【 0 2 2 1 】

図 6 で、「ビットストリーム」及び「ビットストリーム 2」によって示されるビットストリームは、夫々、 $A(h_{new}/16)(w_{new}/16)$ 及び $B(h_{new}/64)(w_{new}/64)$ である。A 及び B は、圧縮比を記述するスカラーパラメータである。圧縮比が高ければ高いほど、数 A 及び B はますます小さくなる。ビットストリームの全体のサイズは、従って、 $A(h_{new}/16)(w_{new}/16) + B(h_{new}/64)(w_{new}/64)$ として与えられる。

【 0 2 2 2 】

圧縮の目標は、再構成された画像の品質を高く保ちながらビットストリームのサイズを低減することであるから、 h_{new} 及び w_{new} は、ビットレートを下げるよう可能な限り小さくしなければならないことが明らかである。

40

【 0 2 2 3 】

従って、「ゼロでパディングすること」の問題は、入力サイズの増大によるビットレートの増大である。換言すれば、入力画像のサイズは、入力画像に冗長データを加えることによって増大し、つまり、更なるサイド情報が、入力信号の再構成のためにエンコーダからデコーダへ伝送されなければならない。その結果、ビットストリームのサイズは大きくなる。

【 0 2 2 4 】

一例として、図 6 のエンコーダ／デコーダの組を用いると、入力画像が、WQVGA（

50

Wide Quarter Video Graphics Array)として一般的に知られている画像サイズフォーマットであるサイズ416×240を有する場合に、入力画像は、サイズ448×256に等しくなるようパディングされなければならない、これは、冗長データの包含によるビットレートの15%の増大に等しい。

【0225】

第2アプローチ(入力画像のクロッピング)に伴う問題は、情報の損失である。圧縮及び圧縮解除の目標は、忠実度を高く保ちながら入力信号を伝送することであるから、信号の部分を捨てる目的に逆らっている。従って、クロッピングは、通常はそうではないが、入力信号に不要部分があることが分からない限りは、有利ではない。

【0226】

一実施形態に従って、入力画像のサイズ調整は、DNNに基づいたピクチャ又はビデオ圧縮システムの全てのダウンサンプリング又はアップサンプリングレイヤの前に行われる。より具体的には、ダウンサンプリングレイヤがダウンサンプリング比2を有する(レイヤの出力で入力サイズが半分になる)場合に、入力リサイジングは、レイヤの入力に対して、それが奇数のサンプル行又は列を有する場合に適用され、パディングは、サンプル行又は列の数が偶数(2の倍数)である場合に適用されない。

【0227】

更に、リサイジング操作は、対応するダウンサンプリングレイヤが(その)入力でリサイジングを適用されている場合に、最後に、例えば、アップサンプリングレイヤの出力で適用することができる。ダウンサンプリングレイヤの対応するレイヤは、再構成された画像から開始してアップサンプリングレイヤの数を数え、入力画像から開始してダウンサンプリングレイヤの数を数えることによって、見つけることができる。これは図18で例示されており、アップサンプリングレイヤ1及びダウンサンプリングレイヤ1は対応する例やであり、アップサンプリングレイヤ2及びダウンサンプリングレイヤ2は対応するレイヤである、など。

【0228】

ダウンサンプリングレイヤの入力で適用されるリサイジング操作及びアップサンプリングレイヤの出力で適用されるリサイジング操作は、両方の出力でのデータのサイズが同じままであるように相補的である。

【0229】

結果として、ビットストリームのサイズの増大は最小限にされる。例示的な実施形態は、図12を参照して説明することができ、これは、図9とは対照的に、他のアプローチについて記載している。図9で、入力のリサイジングは、入力でDNNへ供給される前に行われ、リサイジングされた入力がDNN全体を通して処理され得るように行われる。図9に示される例は、図6に記載されているエンコーダ/デコーダにより実現(実施)されてもよい。

【0230】

図12では、任意のサイズを有する入力画像がニューラルネットワークへ供給される。この実施形態のニューラルネットワークはN個のダウンサンプリングレイヤを有し、各レイヤ i ($1 \leq i \leq N$)はダウンサンプリング比 r_i を有する。「 $\leq$ 」は、以下を示す。ダウンサンプリング比 r_i は、 i の異なる値ごとに必ずしも同じではなく、いくつかの実施形態では全ての等しく、例えば、全て $r_i = r = 2$ であることができる。図12で、ダウンサンプリングレイヤ1からMは、ダウンサンプリングレイヤのサブネット1としてまとめられる。サブネット1(サブネットワーク1)はビットストリーム1を出力として供給する。ダウンサンプリングレイヤの集約は、本文脈中、しかしながら、記載のためだけである。レイヤM+1からNを含む第2サブネット2(サブネットワーク2)はビットストリーム2を出力として供給する。

【0231】

この実施形態では、ダウンサンプリングレイヤ、例えば、ダウンサンプリングレイヤMはへの入力がダウンサンプリングレイヤへ供給される前であるが、それが前のダウンサン

10

20

30

40

50

プリングレイヤ（この場合に、レイヤ $M-1$ ）によって処理された後に、入力は、リサイジング操作を適用することによってリサイジングされ、それにより、ダウンサンプリングレイヤ M への入力はサイズ
【数 1 5 5】

$$S = nr_M, n \in \mathbb{N}$$

を有する。 r_M は、ダウンサンプリングレイヤ M のダウンサンプリング比を表し、前もってセットされた値であってよく、従って、デコーダで既に利用可能であり得る。この実施形態で、このリサイジング操作は、上記の条件が特定のダウンサンプリングレイヤ及びその各々のダウンサンプリング比について満足されるように、各ダウンサンプリングの前に実行される。換言すれば、サイズ S は、後続の（一連の処理においてダウンサンプリングに続く）レイヤのダウンサンプリング比の整数倍に適応されるか又はそのようにセットされる。

【0 2 3 2】

図 9 で、入力画像は、データを次々に処理するであろう全てのダウンサンプリングレイヤを考慮するようパディングされる（これは画像リサイジングの一種である）。図 9 で、ダウンサンプリング比は、例として、デモンストレーションの目的で、2 に等しくなるよう選択される。この場合に、比率 2 でダウンサンプリングを実行する N 個のレイヤが存在するので、入力画像サイズは、 2^N の整数倍であるよう（ゼロによる）パディングによって調整される。ここで、整数“倍”は依然として 1 に等しくてもよく、つまり、倍数は、複数の意味よりむしろ、（例えば、1 以上による）定数の意味を有することに留意されたい。

【0 2 3 3】

実施形態は図 1 2 で示されている。図 1 2 で、入力リサイジングは各ダウンサンプリングレイヤの前に適用される。入力は、各レイヤのダウンサンプリング比の整数倍であるようリサイジングされる。例えば、レイヤのダウンサンプリング比が 3 : 1（入力サイズ：出力サイズ）、つまり比率 3 である場合に、レイヤの入力は、3 の倍数になるようリサイジングされる。

【0 2 3 4】

いくつかの実施形態は図 6 にも適用することができる。図 6 では、ダウンサンプリングを有する 6 つのレイヤ、つまり、レイヤ 8 0 1、8 0 2、8 0 3、8 0 4、8 0 5、及び 8 0 6 が存在する。全てのダウンサンプリングレイヤは係数 2 を有する。一実施形態に従って、入力リサイジングは、全ての 6 つのレイヤの前に適用される。図 6 で、リサイジングは、アップサンプリングレイヤ（8 0 7、8 0 8、8 0 9、8 1 0、8 1 1、及び 8 1 2）の各レイヤ出力の後にも、対応する方法（上の段落で説明される）で適用される。これは、エンコーダのニューラルネットワークにおいて具体的な順序又は位置でダウンサンプリングレイヤの前に適用されたりリサイジングが、デコーダ内の対応する位置で適用されることを意味する。

【0 2 3 5】

いくつかの実施形態において、入力をリスケーリングする 2 つのオプションが存在し、それらのうちの 1 つは、例えば、更に以下で説明される条件又は状況に応じて、選択され得る。これらの実施形態は、図 1 3 から 1 5 を参照して記載される。

【0 2 3 6】

第 1 のオプション 1 5 0 1 は、入力のサイズを、ダウンサンプリング比の整数倍に一致するサイズに増大させるために、入力を、例えば、ゼロ又は入力それ自体からの冗長な情報でパディングすることを有してよい。デコーダ側では、リスケーリングするために、クロッピングが、このオプションでは、入力のサイズを、例えば、後続のアップサンプリングレイヤの目標入力サイズに一致するサイズに低減させるために、使用されてよい。

【0 2 3 7】

このオプションは、計算効率良く実施することができるが、エンコーダ側ではサイズを増大させることしか可能でない。

【0238】

第2のオプション1502は、入力をリスケーリング／リサイジングするために、エンコーダでは補間を、デコーダでは補間を利用してよい。これは、補間が、入力のサイズを、後続のダウンサンプリングレイヤのダウンサンプリング比の整数倍、又は後続のアップサンプリングレイヤの目標入力サイズなどの意図されたサイズに増大させるために使用され得ること、あるいは、補間が、後続のダウンサンプリングレイヤのダウンサンプリング比の整数倍、又は後続のアップサンプリングレイヤの目標入力サイズなどの意図されたサイズに低減させるために使用され得ることを意味する。それによって、入力のサイズを増大又は低減させることによってエンコーダでリサイジングを適用することが可能である。更に、このオプション1502では、異なる補間フィルタが使用されてもよく、それによってスペクトル特性制御がもたらされる。

【0239】

異なるオプション1501及び1502は、例えば、サイド情報としてビットストリームにおいて、通知され得る。第1のオプション（オプション1）1501と第2のオプション（オプション2）1502との間の違いは、2つの値のうちの1つをとることができるシンタックス要素 `methodIdx` などの指示で通知され得る。例えば、第1の値（例えば、0）は、パディング／クロッピングを示すためのものであり、第2の値（例えば、1）は、リサイジングのために補間が使用されることを示すためのものである。例えば、デコーダは、ピクチャを符号化しており、場合により、要素 `methodIdx` を含むサイド情報を有するビットストリームを受信し得る。このビットストリームをパースすると、サイド情報が取得され、`methodIdx` の値が導出され得る。`methodIdx` の値に基づいて、デコーダは、次いで、`methodIdx` が第1の値を有する場合にはパディング／クロッピングを用いて、あるいは、`methodIdx` が第2の値を有する場合に対応する場合には補間を用いて、リサイジング又はリスケーリング方法が続けることができる。

【0240】

これは図13に示されている。`methodIdx` の値が0又は1であることに依存して、クリッピング（パディング若しくはクロッピングのどちらかを含む）又は補間のいずれかが選択される。

【0241】

たとえ図13の実施形態が、リサイジングを実現するために使用される方法として、`methodIdx` に基づいた、クリッピング（パディング／クロッピングの一方を含む）と補間との間の選択又は決定に言及しているとして、本発明はこれに関して制限されないことに留意されたい。図13に関して説明されている方法は、第1のオプション1501がリサイジング操作中にサイズを増大させる補間であり、第2のオプション1502がリサイジング操作中にサイズを低減させる補間である場合にも実現され得る。上又は下で説明されているいずれか2つ又はそれ以上の（`methodIdx` の二値サイズに依存する）異なるリサイジング方法を選択して、`methodIdx` で通知することができる。一般に、`methodIdx` は、別のシンタックス要素である必要はない。それは、他の1つ以上のパラメータと一緒に指示又はコーディングされてもよい。

【0242】

更なる指示又はフラグが図14に示されるように供給されてもよい。`methodIdx` に加えて、サイズ変更フラグ（1ビット）`SCIdx` が、第2のオプション1502の場合についてのみ条件付きで通知され得る。図14の実施形態では、第2のオプション1502は、リサイジングを実現するための補間の使用を含む。図14で、第2のオプション1502は、`methodIdx` = 1の場合に選択される。サイズ変更フラグ `SCIdx` は、第3又は第4の値を有する可能性があり、これは、0（例えば、第3の値に対する）又は1（例えば、第4の値に対する）のどちらか一方の値であってよい。この実施形態

では、“0”はダウンサイジングを示してよく、“1”はアップサイジングを示してよい。よって、SCIdxが0である場合には、入力のサイズが低減されるように、リサイジングを実現する補間が行われる。SCIdxが1である場合には、入力サイズを増大させるように、リサイジングを実現する補間が行われ得る。SCIdxの条件付きコーディングは、より簡潔かつ効率的なシンタックスを提供し得る。しかし、本開示は、そのような条件付きシンタックスによって制限されず、SCIdxは、methodIdxとは無関係に指示されても、又はmethodIdxと一緒に（例えば、SCIdx及びmethodIdxの全ての組み合わせを示す値の中から値の一部のみをとることができる共通のシンタックス要素内で）指示（コーディング）されてもよい。

【0243】

指示methodIdxと同様に、SCIdxも、再構成されるべきピクチャも潜在的に復号するビットストリームを解析することによってデコーダによって取得され得る。SCIdxの値を取得すると、ダウンサイジング又はアップサイジングが選択され得る。

【0244】

上記の指示に加えて、又はその代替案として、図15に示されるように、リサイジングフィルタインデックスRFIdxの追加（サイド）指示が通知されてもよい（ビットストリーム内で指示される）。

【0245】

いくつかの例示的な実施形態において、RFIdxは、第2のオプション1502の場合に条件付きで指示されてよく、RFIdxが、methodIdx=1の場合に通知され、methodIdx=0の場合に通知されないことを有してよい。RFIdxは、1ビットよりも大きいサイズを有してよく、例えば、その値に応じて、どの補間フィルタがリサイジングを実現する補間で使用されるかを通知し得る。代替的に、又は追加的に、RFIdxは、複数の補間フィルタからフィルタ係数を指定してもよい。これは、例えば、とりわけ、双線形、バイキュービック、Lanczos3、Lanczos5、Lanczos8であってよい。

【0246】

上述されたように、methodIdx、SCIdx、及びRFIdxのうちの少なくとも1つ又はそれらの全て又はそれらのうちの少なくとも2つは、ビットストリームで供給されてよく、ビットストリームは、再構成されるべきピクチャも符号化するビットストリームであってよく、あるいは、追加のビットストリームである。デコーダは、次いで、各々のビットストリームをパースし、methodIdx及び／又はSCIdx及び／又はRFIdxの値を取得し得る。値に応じて、上述された動作が行われてよい。

【0247】

リサイジングを実現する補間に使用されるフィルタは、例えば、スケーリング比によって決定することができる。

【0248】

項目171により図15の右下に示されるように、RFIdxの値は明示的に通知されてよい。代替的に、又は追加的に、RFIdxは、RFIdx=LUT(SCIdx)であるように、ルックアップテーブルから取得されてもよい。

【0249】

他の例では、2つのルックアップテーブルが存在し、アップサイジングの場合に1つ、ダウンサイジングの場合に1つであってもよい。この場合に、LUT1(SCIdx)は、ダウンサイジングが選択される場合のリサイジングフィルタを示してよく、LUT2(SCIdx)は、アップサイジングの場合のリサイジングフィルタを示してよい。一般に、本開示は、任意の特定のRFIdxシグナリング方法に制限されない。それは、個別的であり、他の要素から独立していても、あるいは、一緒に通知されてもよい。

【0250】

図16及び図17は、リサイジング方法のいくつかの例を示す。図16及び図17には、3つの異なるタイプのパディング操作及びそれらの性能が示されている。示されている

10

20

30

40

50

図中の横軸は、サンプル位置を示す、縦軸は、各々のサンプルの値を示す。

【0251】

続く説明は、単なる例示であって、本発明を特定の種類のパディング操作に制限する意図はないことに留意されたい。直線の縦線は、入力（実施形態に従って、ピクチャ）の境界を示し、境界の右手側は、新しいサンプルを生成するようパディング操作が適用されるサンプル位置である。これらの部分は、以下で「利用不可能部分」とも呼ばれ、それらが元の入力に存在せず、更なる処理のためにリスケーリング操作中にパディングによって加えられていることを意味する。入力境界線の左側は、利用可能であり、入力の部分であるサンプルを表す。図に表されている3つのパディング方法は、複製パディング、反射パディング、及びゼロ充てんである。いくつかの実施形態に従って実行されるべきダウンサンプリング操作の場合に、NNのダウンサンプリングレイヤへの入力は、パディングされた情報、つまり、適用されたパディングによって拡張された元の入力になる。

【0252】

図16において、利用不可能であり、パディングによって充てんされ得る位置（つまり、サンプル位置）は、位置4及び5である。ゼロ充てんの場合に、利用不可能な位置は、値0を有するサンプルで満たされる。反射パディングの場合に、位置4でのサンプル値は、位置2でのサンプル値に等しくセットされ、位置5での値は、位置1での値に等しくセットされる。換言すれば、反射パディングは、入力境界にある最後の利用可能なサンプルである位置3での利用可能なサンプルをミラーリングすることと同等である。複製パディングの場合に、位置3でのサンプル値は、位置4及び5にコピーされる。異なるパディングタイプは、異なるアプリケーションごとに好まれ得る。具体的に、適用されるパディングタイプは、実行されるべきタスクに依存し得る。例えば：

ゼロによるパディング又は充てんは、認識又は検出タスクなどのコンピュータビジョン（CV）タスクに使用されるのが適当であり得る。それによって、元の入力に既に存在している情報の量／値／重要度を変更しないために、情報は追加されない。

【0253】

反射パディングは、追加される値が、定義された“反射ライン”（つまり、元の入力の境界）に沿って既存の値からコピーされさえすればよいので、計算上容易なアプローチであり得る。

【0254】

複製パディング（繰り返しパディングとも呼ばれる）は、ほとんどのサンプル値及び導関数の連続性が留保されるので、畳み込みレイヤによる圧縮タスクのために好ましい場合がある。サンプル（利用可能なサンプル及びパディングされたサンプルを含む）の導関数は、図16及び図17の右手側に示されている。例えば、反射パディングの場合に、信号の導関数は、位置4での急激な変化を示す（図示されている例示的な値については、-9の値が得られる）。滑らかな信号（導関数が小さい信号）は圧縮が容易であるから、ビデオ圧縮タスク場合に反射パディングを使用することは望ましくない場合がある。

【0255】

示されている例では、複製パディングは、導関数の変化が最も小さい。これは、ビデオ圧縮タスクの観点から有利であるが、より多くの冗長な情報が境界で加えられることになる。これにより、境界での情報は、他のタスクで意図されたよりも重みになる可能性がある。いくつかの実施では、ゼロによるパディングの全体的な性能は、反射パディングよりも優先される場合がある。

【0256】

図18は更なる実施形態を示す。ここで、エンコーダ2010及びデコーダ2020は横並びに示されている。示されている実施形態において、エンコーダは複数のダウンサンプリングレイヤ1からNを有する。ダウンサンプリングレイヤは、グループにまとめられ、あるいは、エンコーダ2010内のニューラルネットワークのサブネットワーク2011及び2012の部分を形成することができる。これらのサブネットワークは、例えば、デコーダ2020へ供給され得る特定のビットストリーム1及び2を供給することに関与

10

20

30

40

50

することができる。この意味で、エンコーダのダウンサンプリングレイヤのサブネットワークは、合理的に分離することができない論理的なユニットを形成し得る。図18に示されるように、エンコーダ2020の第1サブネットワーク2011はダウンサンプリングレイヤ1から3を含み、夫々がその各々のダウンサンプリング比を有する。第2サブネットワーク2012は、各々のダウンサンプリング比を持ったダウンサンプリングレイヤMからNを含む。

【0257】

デコーダ2020は、アップサンプリングレイヤ1からNの対応する構造を備えている。デコーダ2020の1つのサブネットワーク2022は、アップサンプリングレイヤNからMを含み、他のサブネットワーク2021は、アップサンプリングレイヤ3から1（ここでは、各々の入力の実処理順序で見たときに、番号付けがデコーダと一致するように降順にしている）を含む。

10

【0258】

上述されたように、エンコーダのダウンサンプリングレイヤ2の前に入力に適用されたリスケーリングは、相応して、アップサンプリングレイヤ2の出力に適用される。これは、上述されたように、ダウンサンプリングレイヤ2への入力のサイズがアップサンプリングレイヤ2の出力のサイズと同じであることを意味する。

【0259】

より一般的には、エンコーダのダウンサンプリングレイヤnの入力に適用されたリスケーリングは、アップサンプリングレイヤnの出力に適用されるリスケーリングに対応し、それにより、リスケーリングされた入力のサイズは、リスケーリングされた出力サイズと同じになる。

20

【0260】

図19から23に関連して続く記載では、いくつかの更なる実施形態が更に詳細に説明される。後述される実施形態は上記の実施形態のいずれかと組み合わせることが意図され、特に、符号化及び復号化におけるニューラルネットワークの具体的な使用については、図12から15及び18を参照されたい。更に、図13から15で説明されているような、どのリスケーリングモードが行われるべきかを通知する具体的な方法は、必要なリスケーリングに関する情報や、ピクチャに加えて、ピクチャをビットストリームに符号化するとき適用されるべきリスケーリングに関する情報を供給する情報を、例えば、ピクチャを符号化しているビットストリームで、デコーダへ供給するために、以下の教示と組み合わせられてよい。また、後述される実施形態は、図1から3に関連して記載されているシステム及び方法のいずれかを用いて実施されてよく、あるいは、それらは、これらの図で記載されているシステム及び方法の部分として実施されてもよい。

30

【0261】

図19及び20は、いくつかの実施形態に使用され得るエンコーダ（図19）及びデコーダ（図20）の例示的な実施形態を示す。エンコーダ及びデコーダの具体的な実施形態は、図1から3及び図4及び6から8を参照して上で既に説明されている。各々の実施形態は、続く記載に含まれることが意図される。これは、図1から3及び図4及び6から8並びに図10及び11に関連して記載されている全てのコンポーネント及び機能が、たとえ以下で明示的に言及されていないとしても、図20のデコーダ及び図19のエンコーダにも提供されてよいことを意味する。

40

【0262】

エンコーダ及びデコーダの構造に関して詳細に入る前に、以下の記載は、エンコーダ又はデコーダの具体的な実現に本発明を制限する意図はない。更に、それらの夫々は、エンコーダ／デコーダの機能を実現するよう汎用ハードウェアと相互作用するソフトウェアを使用すること、又はその構造に基づいて、エンコーダ／デコーダの機能を実現する具体的に適応されたハードウェアを使用することのいずれかによって、実施され得る。更に、それらの組み合わせも可能である。

【0263】

50

更に、続く記載は、エンコーダ及びデコーダで提供されるニューラルネットワークに焦点を当てている。しかし、エンコーダ／デコーダは、パッシングユニット、インタープリタ、暗号化ユニット、暗号解読ユニット、ピクチャ又は他のデータを受信する受信ユニット、データ及び／又はビットストリームを出力する出力ユニットなどの更なるコンポーネントも有してよい。

【0264】

パッシングユニットは、当業者に知られているパーサとして実現されてもよい。受信ユニットは、アンテナ又は受信器として実施されてよく、あるいは、当業者に知られているように、それらを有してもよい。また、出力ユニットは、送信器又はアンテナとして実施されてよく、あるいは、それらを有してもよい。更に、受信ユニット及び出力ユニットは、単一のコンポーネントとして、例えばトランシーバとして、実施されてもよい。

【0265】

図19から始めると、エンコーダ2100のコンポーネントが模式的に示されている。

【0266】

この実施形態において、エンコーダは、複数のレイヤ2110、2120、2130及び2140を含むニューラルネットワークを有する。

【0267】

それらのレイヤの夫々は、望ましくは、特定の 방법으로ニューラルネットワークの各々のレイヤへの入力2101、2102、2103、2104を処理するよう構成される。図19において、入力2101はニューラルネットワークへの入力であり、つまり、ニューラルネットワークの先行レイヤは入力を処理していない。いくつかの実施形態に従って、ニューラルネットワークへの入力はピクチャである。ピクチャは、静止画若しくは動画又はそれらの部分であってよい。入力は必ずしもピクチャそのものである必要はなく、ピクチャを表す適切なデータ構造の形で供給されてもよい。例えば、ピクチャは、高さ及び幅を有するマトリクス並びに対応するサンプルの値を表すマトリクスの各エントリのように扱われ得る複数のサンプルの形で供給され得る。

【0268】

出力2105はここでは、ニューラルネットワークの最終的な出力として示され得る。これに関して、本発明は、前処理、例えば、パース、暗号解読、などされていない入力に限られない。また、出力は、暗号化などの何らかの更なる処理を受け得る。一般に、出力2105は、ビットストリーム又は符号化されたビットストリームと呼ばれ得る。それは、入力ピクチャの表現であるか、又はそれを含む。

【0269】

更に、図19に関連して、エンコーダ2100は、ニューラルネットワーク4つのレイヤしか有さないことに制限されず、任意の数のレイヤが、各々の入力を処理するためにニューラルネットワークに設けられてもよい。

【0270】

ピクチャは、例えば、単一のピクチャ若しくはピクチャファイル又は動画若しくはビデオファイルのような動画ファイル、あるいは、例えば高さH及び幅Wとして表され得る2つの次元を有するマトリクスによって表現することができる任意の他の情報を表し得る。

【0271】

一実施形態に従って、エンコーダは、受信された入力にダウンサンプリングを適用するよう構成されている、ニューラルネットワークの複数のレイヤ内の少なくとも1つのレイヤを有する。入力は必ずしも元のピクチャである必要はなく、何らかの変更をピクチャに既に適用したニューラルネットワークの先行レイヤの出力であってもよく、あるいは、それは、ピクチャを前処理したプリプロセッサの出力であってもよい。

【0272】

かようなダウンサンプリングは、ピクチャを符号化するために使用され得る。このダウンサンプリングによって、ピクチャのサイズは、例えば、ビットストリームが実際に表すマトリクスのサイズを低減することによって、低減される。

【0273】

例えば、 1024×512 ピクセルの解像度を有するピクチャが例として考えられ得る。説明を簡単にするために、このピクチャは、グレースケールピクチャであるように1つ（カラーチャネル）しか提供しなくてもよく、従って、2次元マトリクス内で各ピクセルが 1024×512 個の点を形成する単一色（強度）値を有する、ことが更に仮定される。本発明の一実施形態に従ってダウンサンプリングを適用することによって、このマトリクスのサイズは、2つの次元のうちの少なくとも1つで係数2によって低減され、例えば、 512×256 ピクセルのサイズを有するダウンサンプリングされた画像をもたらす。これは、ニューラルネットワークの複数のレイヤのうちの少なくとも1つのダウンサンプリングレイヤを、受信した入力に畳み込みを適用する畳み込みとして設けることによって、達成され得る。他の例では、ダウンサンプリングは、1つの次元で入力の n 番目ごとのサンプルを保持し、 n 個ごとのサンプルのうちの $n-1$ 個のサンプルを破棄することによって、実施され得る。

10

【0274】

ダウンサンプリング操作は1回以上適用されてよく、つまり、サイズが 512×256 であるダウンサンプリングされた中間画像は、両方の次元で係数2によって更にダウンサンプリングすることができ、 256×128 の出力サイズが得られる。ダウンサンプリングの目的は、互いに相関性が高いサンプルを捨てることであるから、画像の圧縮性は高まる。自然画像では、隣接するサンプル値は通常非常に類似している。例えば、青空の画像の1つのピクセルサンプルは、隣のピクセルサンプルに非常に類似している（どちらも青いが、色合いがわずかに変わる可能性がある）これは、画像の隣接するサンプルの値が非常に相関しており、1つのサンプルの値が分かれば、隣接するサンプルの値について非常に正確に推測できることを意味する。ダウンサンプリング操作は、そのような冗長性を排除することを目的としているので、冗長な情報を捨てることによって、画像をより圧縮しやすくする。結果として生じるコーディングされたビットストリームサイズは、冗長な情報がどれくらい含まれているかに依存するので、冗長な情報が少なければ少ないほど、ビットストリームのサイズはますます小さくなる。

20

【0275】

上記の例では、NNが、（ダウンサンプリングを実行しない他の処理レイヤとともに）2のダウンサンプリング比を夫々有している全部で2つのダウンサンプリングレイヤを有する場合に、かつ、入力画像のサイズが 1024×512 である場合に、問題は観測されない。2つのダウンサンプリング操作の後以降、結果として得られるダウンサンプリングされた出力は 256×128 である。しかし、入力が 1024×511 のサイズを有する場合には、最初のダウンサンプリングの後で、中間出力の期待されるサイズが 512×255 、2であって整数ではなく、部分的なピクセルサンプルを有することが不可能であるために、NNにより入力を処理することは不可能となる。これは、当該例でのNNが、4×4の倍数ではない入力画像を処理することができないことを意味する。

30

【0276】

問題は、上では、少数（例えば、2つ）のダウンサンプリングレイヤの場合について説明された。しかし、画像圧縮は複雑なタスクであり（画像は、膨大な量の次元を有する信号であるから）、通常、ディープニューラルネットワークがこのタスクを実行するために必要である。これは、通常、NNに含まれているダウンサンプリングレイヤの数が2つよりも多いか又はそれよりずっと多いことを意味する。これは、例えば、ダウンサンプリングレイヤの数が6である（2のダウンサンプリング比を夫々有する）場合に、NNが、 $2^6 \times 2^6 = 64 \times 64$ の倍数である入力サイズしか処理することができないために、問題を大きくする。異なるエンドユーザデバイスによって得られた画像のほとんどは、この要件を満足しない。

40

【0277】

かような畳み込みは、入力の元のマトリクス（この場合、 1024×512 のエントリを含むマトリクスであり、エントリは M_{ij} で表される）のエントリと、このマトリクス

50

に対して実行（シフト）され、通常は入力のサイズよりも小さいカーネル K との要素ごとの乗算を有する。2つの離散変数の畳み込み演算は次のように記載され得る：

【数 1 5 6】

$$(f * g)[n] = \sum_{m \in K} f[m]g[n - m]$$

従って、 n の全てのとり得る値に対する関数 $(f * g)[n]$ の計算は、入力アレイ $g[]$ に対してカーネル又はフィルタ $f[]$ を実行（シフト）し、各シフトされた位置の要素ごとの乗算を実行することと同等である。

【0 2 7 8】

上記の例では、カーネル K は、2のステップング範囲によって入力に対して実行される 2×2 マトリクスであるから、ダウンサンプリングされたビットストリーム D の最初のエントリ D_{11} は、カーネル K にエントリ M_{11} 、 M_{12} 、 M_{21} 、 M_{22} を乗じることによって取得される。水平方向における次のエントリ D_{12} は、次いで、カーネルとエントリ及びエントリ M_{12} 、 M_{14} 、 M_{23} 、 M_{24} を含む縮退されたマトリクスとの内積を計算することによって取得され得る。垂直方向で、これは相応して実行され、それにより、 M と K との各々の内積を計算することで取得されたエントリ D_{ij} を有し、方向又は次元ごとに半分のエントリしか有さないマトリクス D が最終的に取得される。

【0 2 7 9】

換言すれば、畳み込み出力を取得するために使用されるシフト量は、ダウンサンプリング比を決定する。カーネルが各計算ステップの間で2サンプルをシフトされる場合に、出力は係数2によってダウンサンプリングされる。2のダウンサンプリング比は、次のように上記の式で表現することができる：

【数 1 5 7】

$$(f * g)[n] = \sum_{m \in K} f[m]g[(2n) - m]$$

転置畳み込み演算は、畳み込み演算と同じように数学的に表現することができる。「転置」という用語は、上記の転置畳み込み演算が特定の畳み込み演算の反転に対応するという事実に対応する。しかし、実施ごとに、転置畳み込み演算は、上記の式を使用することによって同様に実施され得る。転置畳み込みを使用することによるアップサンプリング演算は、関数：

【数 1 5 8】

$$(f * g)[n] = \sum_{m \in K} f[m]g[\text{int}(n/u) - m]$$

を使用することによって実施することができる。上記の式において、 u はアップサンプリング比に対応し、 $\text{int}()$ は整数への変換に対応する。 $\text{int}()$ は、例えば、丸め演算として実施することができる。

【0 2 8 0】

上記の式において、値 m 及び n は、畳み込みカーネル又はフィルタ $f()$ 及び入力変数アレイ $g()$ が1次元アレイである場合にスカラーインデックスであることができる。そ

れらはまた、カーネル及び入力アレイが多次元である場合に多次元インデックスとして理解することもできる。

【0281】

本発明は、畳み込み又は逆畳み込みによるダウンサンプリング又はアップサンプリングに制限されない。如何なる可能なダウンサンプリング又はアップサンプリング方法も、ニューラルネットワークNNのレイヤにおいて実施され得る。

【0282】

このプロセス（ダウンサンプリング）は、1つよりも多いダウンサンプリングレイヤがより一層サイズを低減するためにニューラルネットワーク内に設けられる場合に繰り返され得る。それによって、符号化されたビットストリーム2105は、図19に従ってニューラルネットワークから出力として供給され得る。この繰り返しダウンサンプリングは、図6、図10及び図11で議論されているエンコーダにおいて実装され得る。

【0283】

エンコーダ及び具体的にニューラルネットワークのレイヤは、畳み込みを適用するダウンサンプリングレイヤを単に有するものに制限されず、入力サイズの低減を取得する畳み込みを必ずしも適用する必要がない他のダウンサンプリングレイヤも考えることができる。

【0284】

更に、ニューラルネットワークのレイヤは、ニューラルネットワークのそれらの対応するレイヤの各々の入力及び／又は出力に対して他の演算を実行する更なるユニットを有することができる。例えば、ニューラルネットワークのレイヤ2120はダウンサンプリングレイヤを有してもよく、ダウンサンプリング前のこのレイヤへの入力の処理順序において、正規化線形ユニット（ReLU）及び／又はバッチ正規化が設けられてもよい。

【0285】

正規化線形ユニットは、変更されたエントリ P'_{ij} を形式

【数159】

$$P'_{ij} = \begin{cases} 0 & \forall P_{ij} \leq 0 \\ P_{ij} & \forall P_{ij} > 0 \end{cases}$$

で取得するために、マトリクスPのエントリ P_{ij} に調整を適用することが知られている。それによって、変更されたマトリクス内の値は全て0に等しいかそれよりも大きいことが保証される。これは、いくつかのアプリケーションの場合に必要又は有利であり得る。

【0286】

バッチ正規化は、サイズ $M \times N$ を有するマトリクスPのエントリ P_{ij} から形式

【数160】

$$V = \frac{\sum_{ij} P_{ij}}{N \cdot M}$$

で平均値を最初に計算することによって、マトリクスの値を正規化することが知られている。この平均値Vを用いて、次に、エントリ P'_{ij} を有するバッチ正規化されたマトリクス P' が、 $P'_{ij} = P_{ij} - V$ によって取得される。

【0287】

バッチ正規化によって得られる計算も、正規化線形ユニットによって得られた計算も、どちらもエントリの数（つまりサイズ）を変更せずに、マトリクス内の値のみを変更する。

【0288】

かようなユニットは、状況に応じて、各々のダウンサンプリングレイヤの前、又は各々のダウンサンプリングレイヤの後に配置され得る。具体的に、ダウンサンプリングレイヤはマトリクス内のエントリ数を減らすということで、ビットストリームの処理順序において各々のダウンサンプリングレイヤの後にバッチ正規化を配置する方が適切である。それによって、 V 及び P_{ij} を取得するのに必要な計算の数は大幅に減る。正規化線形ユニットは、一部のエントリが0になる可能性があるため、畳み込みがダウンサンプリングレイヤに使用される場合に、サイズを低減されたマトリクスを取得するための乗算を簡単にすることができるということで、畳み込みの適用の前に正規化線形ユニットを配置することが有利であり得る。

10

【0289】

しかし、本発明はこれに関して制限されず、バッチ正規化又は正規化線形ユニットは、ダウンサンプリングレイヤに対して他の順序で配置されてもよい。

【0290】

更に、各レイヤが必ずしもそれらの更なるユニットのうちの1つを有する必要はなく、あるいは、他の変更又は計算を実行する他の更なるユニットが使用されてもよい。

【0291】

ニューラルネットワークがニューラルネットワークを通じて2101の入力を確実に処理し、最終的に出力2105を得ることができるようにするために、エンコーダのニューラルネットワークを通じて入力ビットストリームを処理するときに適用されるダウンサンプリング操作は、定義された数学的演算であることが必要である。

20

【0292】

畳み込みがダウンサンプリングレイヤの一実施として提供される上記の例を考慮して、実施形態は、例えば、各次元で入力（ピクチャ）によって表されるマトリクスのエントリ数がこの次元でのカーネルのサイズの整数倍に対応することを保証することができる。よって、それらは、畳み込みが数学的に実行され得ず、エンコーダ2100のニューラルネットワークを通じて入力2101を処理するときに障害を発生させたり、あるいは一貫性のない結果をもたらしたりする可能性がある状況を回避することができる。同時に、実施形態は、エンコーダ2100のニューラルネットワークを通じた入力ピクチャ2101の処理の終わりに最終的に出力されるビットストリームのサイズを可能な限り小さいままにすることができる。

30

【0293】

これは、図21に関連して更に詳細に説明される方法を適用することによって達成され、ニューラルネットワークのレイヤへの入力のサイズをリスケーリングするリスケーリング操作を含む。これは、以下で更に詳細に説明される。

【0294】

図20は、一実施形態に係るデコーダ2200を示す。

【0295】

このデコーダは、ここで示されるように連続した順序で複数のレイヤ2210、2220、2230、及び2240を含むニューラルネットワークを少なくとも有し、それにより、入力2201は、ニューラルネットワークの後続レイヤによって処理され、それがニューラルネットワークの全てのレイヤによって処理された（その後の出力2201、2203、2204を得た）後、出力2205が供給される。

40

【0296】

本発明と関連して、デコーダ2200のニューラルネットワークは少なくとも1つのレイヤ、例えば、レイヤ2220を有し、このレイヤ2220への入力2202にアップサンプリング操作を適用する。このアップサンプリング操作は、入力がマトリクスの形で表現され得る場合に、少なくとも1つの次元での入力のサイズの増大を含む。これに関連して図19で既に与えられた説明は、ここでも当てはまる。

【0297】

50

図19の上記の実施形態を考慮すると、デコーダへの入力2201は、画像を表すビットストリームである。具体的に、ビットストリームは、図19のエンコーダによって取得されたビットストリームであってよい。よって、ビットストリームは、静止画若しくは動画又はそれらの部分のようなピクチャを表してよく、前にダウンサンプリングされた（それによって符号化された）ビットストリームのアップサンプリングを適用することによって、具体的に図20に従うデコーダ2200を使用することによって復号され得る。

【0298】

入力ビットストリーム2201のサイズは、それが少なくとも2つの次元を有するマトリクスを表す限りは、このようにして、デコーダ2200の出力2205と比較してこれらの次元のうちの少なくとも1つで小さくなる。具体的に、出力は、デコーダ2200へ入力されたビットストリーム2201から再構成された復号化されたピクチャ又は再構成されたピクチャであってよい。

10

【0299】

しかし、ニューラルネットワークのレイヤは、アップサンプリングレイヤを含むものに限られず、少なくとも1つのアップサンプリングレイヤに加えて、少なくとも1つの更なるレイヤを有することができる。更に、レイヤの数も本発明に従って制限されず、デコーダは任意の数のレイヤを有してよい。

【0300】

アップサンプリングレイヤは、例えば、入力に対して補間又は逆畳み込み（つまり、転置畳み込み演算）を実行することによって、受信した入力を少なくとも1つの次元でアップサンプリングする。これに関連して、アップサンプリングは、アップサンプリング比

20

【数161】

$$u > 1, u \in \mathbb{N}$$

によって入力のサイズを増大させることを含む。

【0301】

デコーダ2200のニューラルネットワーク及び具体的にこのニューラルネットワークのアップサンプリングレイヤは、上で説明された正規化線形ユニット及び／又はやはり上で説明されたバッチ正規化のような追加のユニットを更に有してもよい。

30

【0302】

図19及び図21の実施形態を考えると、デコーダはエンコーダの逆の操作であると仮定することができ、図19のエンコーダ2101によって符号化された入力2201に対してデコーダがその操作を適用する場合に、最初にエンコーダへ供給された入力2101をもたらす。これは理想的なケースであるが、図19に従うエンコーダを用いてビットストリームを符号化する際の情報の損失により、デコーダ2200によるビットストリーム2201の処理の終了時の復号されたビットストリーム2205は、元の入力2101と同じでない可能性がある。

40

【0303】

いくつかの実施形態において、エンコーダは多数の連続した畳み込みレイヤ又は（より一般的には）0からNまで番号付けされたダウンサンプリングレイヤを有すると仮定され得る。デコーダは、そのような場合に、対応する順序で対応する数Nのアップサンプリングレイヤを有してよく、デコーダのアップサンプリングレイヤnのアップサンプリング比uは、エンコーダの対応するダウンサンプリングレイヤのダウンサンプリング比rに等しく、つまり、 $u_n = r_{N-n}$ であり、インデックスn及びN-nは夫々、入力を処理する順序におけるアップサンプリングレイヤ又はダウンサンプリングレイヤの位置を表す。これは、デコーダのニューラルネットワークを通る入力（つまり、ビットストリーム）の処理順序において最初のアップサンプリングレイヤのアップサンプリング比は、エンコーダの

50

ニューラルネットワークを通る入力（つまり、ピクチャ）の処理順序における最後のダウンサンプリングレイヤのダウンサンプリング比に等しいことを意味する。

【0304】

これにより、各々のニューラルネットワークを通る入力の処理において同じポイントで実行されるダウンサンプリング操作と比較して逆のアップサンプリング操作が得られる。

【0305】

上記の実施形態は、例えば、シェーディングなしの白黒ピクチャの2次元マトリクスで表される例示的な入力ピクチャを参照して、エンコーダ及びデコーダに関して記載された。しかし、技術は、夫々の利用可能なカラーチャネルで複数の値を有するピクチャ又は（動画としての）カラー映像のような、より複雑な情報を含むピクチャにも適用されることが、本開示によって企図されている。

【0306】

かような場合について、上述されるような単一のニューラルネットワークでそのようなより複雑な入力も処理することは原理上可能であるが、各カラーチャネルについて、図19及び／又は図20に従って提供される構造を備えた“サブニューラルネットワーク” sNN が各々のエンコーダ及び／又はデコーダで設けられることが好ましい。更に、エンコーダ及び／又はデコーダのニューラルネットワークのレイヤを特徴付ける値は、各 sNN について同一であってよい。しかし、 sNN の1つでニューラルネットワーク及びニューラルネットワークレイヤを規定するパラメータは、他の sNN の各々の値と相違することも企図されてよい。例えば、緑のカラーチャネルを処理するサブニューラルネットワークは、赤の色チャネルと比較して異なるように構造されてよい（カーネルの異なる値又はダウンサンプリング比の異なる順序によるより多い又はより少ないレイヤ又は畳み込みを含む）。これに関連して、如何なる技術的に適当な組み合わせも想到可能である。しかし、エンコーダ内の sNN に関してピクチャの部分を処理することによって取得されたサブビットストリームがエンコーダの各々のニューラルネットワークを通る処理の終わりに同じサイズを有する場合に、それは好ましい場合がある。デコーダにも同じことが好ましい。これは、異なる sNN を通してサブビットストリームを処理することによって処理されたビットストリームが、サブビットストリームを復号することで得られた再構成された画像をもたらすことを意味し、夫々の復号されたサブビットストリームは、復号される場合に、望ましくは、他の復号されたサブビットストリームと同じサイズを有する。

【0307】

図21は、例えば図19のニューラルネットワークの部分として提供されるダウンサンプリングレイヤにより少なくとも1つの次元で入力ピクチャ又は一般的にサイズ S を有する入力を処理する方法を特定する。

【0308】

方法2300はステップ2301から始まり、少なくとも1つの次元でサイズ S を有する入力が受け取られる。説明を容易にするために、ダウンサンプリングは、ただ1つの次元に関して以下で記載される。しかし、本発明は2つの次元で入力をダウンサンプリングするために使用されることは上記から明らかであり、意図される。

【0309】

より容易に説明されるように、入力は、静止画又は動画などのようなピクチャであってよい。更に、ステップ2301は、図19に表されているニューラルネットワーク内の任意のポイントで行われてもよい。よって、入力は、ニューラルネットワークを通して入力2101を処理する順序において最初のレイヤであるレイヤ2110で受け取られる必要はなく、ステップ2301は、その後のダウンサンプリングレイヤ、例えば、インデックス i を有するダウンサンプリングレイヤで行われてもよい。

【0310】

サイズ S を有する入力は、次のステップで、望ましくは、そのサイズが各々のダウンサンプリングレイヤ i のダウンサンプリング比 r_i の倍数の整数に等しいかどうかについてチェックされる。

【0311】

このチェックは、例えば、サイズ S と $\text{ceil}(s/r)r$ 又は $\text{floor}(s/r)r$ との間の差を計算することを含むことができる。この差がゼロである場合に、 S は既に、各々のダウンサンプリングレイヤのダウンサンプリング比 r の整数倍である。差がゼロではない場合には、 S は、ダウンサンプリングレイヤのダウンサンプリング比 r の整数倍ではない。

【0312】

例えば、サイズ $S = 512$ 及びダウンサンプリング比 $r = 4$ を考える。その場合に、 $S = 128r_i$ であり、従って、ダウンサンプリング比の整数倍である。その場合に、上記の計算は、 $\text{ceil}(s/r)r - S = \text{floor}(s/r)r - S = 0$ に至る。サイズ $S = 513$ ならば、それは、このダウンサンプリングレイヤのダウンサンプリング比 r_i の整数倍にならない。その場合に、例えば、 $\text{ceil}(s/r)r - S = 3$ 及び $\text{floor}(s/r)r - S = 1$ である。従って、差のいずれの計算も、ゼロではない差をもたらす。

【0313】

S がダウンサンプリング比 r の整数倍であるかどうかのこの決定は、 S 及び r 又は r_i に依存する関数を使用して、 S がダウンサンプリング比 r の整数倍であることを示す結果を前提として、 S がダウンサンプリング比 r の整数倍ではないときには他の結果を与える他の適切な方法でも行うことができる。

【0314】

この決定に応じて、2つの異なるアプローチがとられ得る。

【0315】

ステップ 2302 で、サイズ S がダウンサンプリング比の整数倍に等しいことが決定される場合に (2310 で示される)、方法はステップ 2311 へ進んでよい。これは、サイズ S を有する入力にリスケーリングが適用されないことを意味する。入力のサイズ S が各々のダウンサンプリングレイヤのダウンサンプリング比 r_i の整数倍であるということで、ステップ 2311 で、入力のダウンサンプリング操作は、入力のサイズがサイズ $S_i < S$ に低減されるように実行され得る。

【0316】

このダウンサンプリングを実行した後、出力はこのレイヤによってニューラルネットワークの処理順序において他のレイヤへ供給され得るか、あるいは、これは既に、図 19f で項目 2105 により示されたニューラルネットワークの最終出力であることができる。

【0317】

更に、前に示されたように、ニューラルネットワークの各々のレイヤが更なるユニットに関連するか又はそれを有する場合に、更なる操作は、ステップ 2311 の前に (例えば、正規化線形ユニットを使用することによって) 及び r 又はステップ 2311 の後に、実際に出力 2312 を供給する前に、適用されてよい。具体的に、ダウンサンプリングを実行した後、バッチ正規化は、低減されたサイズ S_i を有する取得されたビットストリームの値を、これがその後のエンティティへの出力として供給される前に正規化するために使用され得る。

【0318】

他方で、ステップ 2302 で、サイズ S がニューラルネットワークの各々のレイヤのダウンサンプリング比 r_i の整数倍ではないと決定される場合に (ステップ 2320)、リスケーリングが入力に適用されてよく、ダウンサンプリングレイヤが入力を実際に処理することを可能にするために入力のサイズ S を変更する。このリスケーリング操作は、入力のサイズ S を増大又は低減することを有することができ、それにより、変更されたサイズ

【数 162】

10

20

30

40

50

$$\bar{S} = nr_i; n, r_i \in \mathbb{N}^+$$

が得られる。

【 0 3 1 9】

望ましくは、サイズ

【数 1 6 3】

$$\bar{S}$$

10

は、元のサイズ S に行われる変更が可能な限り少なくなるように決定される。望ましくは、サイズ S は、ダウンサンプリング比 r の最も近い小さい方の整数倍に又はダウンサンプリング比 r の最も近い大きい方の整数倍に変更される。ダウンサンプリング比 r の最も近い小さい方の整数倍又は最も近い大きい方の整数倍は、関数 $f(n) = |S - nr|$ を最小化することによって取得され得る。正の整数値に制限されるように、この最小化は、ダウンサンプリング比 r の最も近い小さい方の整数倍又は最も近い大きい方の整数倍のどちらかをもたらす。ダウンサンプリング比の最も近い小さい方の整数倍又は最も近い大きい方の整数倍は、 S が既にダウンサンプリング比 r の整数倍である場合に、 S と同じであることもできる。

20

【 0 3 2 0】

代替的に、

【数 1 6 4】

$$\bar{S}$$

は、 $\text{ceil}(s/r)r$ 又は $\text{floor}(s/r)r$ のどちらか一方を計算することによっても取得され得る。前者は、ダウンサンプリング比の最も近い大きい方の整数倍をもたらす、一方、後者は、ダウンサンプリング比の最も近い小さい方の整数倍をもたらす。

30

【 0 3 2 1】

これら 2 つの選択肢には、 n の値を明示的に取得することが必ずしも含まれるわけではない。 n の各々の値は、これらが n の値を提供するため、別々のステップで $\text{ceil}(s/r)$ 又は $\text{floor}(s/r)$ を計算することによって明示的に取得され得る。

【 0 3 2 2】

サイズ

【数 1 6 5】

$$\bar{S}$$

40

を取得すると、入力のサイズ S は、そのサイズが

【数 1 6 6】

$$\bar{S}$$

に変更されるようにリスケーリングを適用することにおいて増大又は低減される。これは

50

、如何なる適切な方法でも行われ得る。サイズ S からサイズ
【数 1 6 7】

$$\bar{S}$$

へのサイズの低減をもたらすリスケーリングの場合について、これは、図 1 3 から 1 5 に
関して上で説明されたように、入力をクロッピングするか、又は補間を使用することによ
って、行われてもよい。リスケーリングがサイズ S からサイズ
【数 1 6 8】

10

$$\bar{S}$$

へのサイズの増大をもたらす場合に、これは、入力にゼロをパディングするか、又は補間
を使用するか、又は冗長な情報によるパディングを使用することによって、行われてもよ
い。ここで、具体的に、反射パディング又は繰り返しパディングは、図 1 3 から 1 5 に関
してやはり上で説明されたように使用されてよい。

【0 3 2 3】

20

明示的な例を与えるために、上述されたように、入力が 5 1 3 のサイズを有するが、ダ
ウンサンプリング比 r が 4 である場合に、入力サイズは、ダウンサンプリング比の整数倍
に対応するために、1 では大きすぎ、3 では小さすぎる。従って、境界での“0”による元
の入力のパディング（又は反射若しくは繰り返しパディングによって元の入力内の既存の
サンプルを存在しない位置にコピーすること）は、サイズ S を、その場合にダウンサンプ
リング比 r の整数倍に等しいサイズ

【数 1 6 9】

$$\bar{S}$$

30

に増大させるために行われる。代替的に、サイズ S を有する入力は、サイズ S をサイズ
【数 1 7 0】

$$\bar{S}$$

に低減するよう入力の 1 つ以上の境界でエントリを削除することによってクロッピングさ
れてもよい。他の手段、例えば、補間が、各々のリスケーリングを実行するために使用さ
れてもよい。

【0 3 2 4】

40

本発明はこれに関して制限されず、図 2 3 に関して更に説明されたように、入力のサイ
ズを増大すべきか又は入力のサイズを低減すべきかの決定は、また、このためにどのリス
ケーリング操作を実際に使用すべきかの決定も、特定の条件に依存してよい。

【0 3 2 5】

更に、ニューラルネットワークにより入力を処理するとき、次のダウンサンプリングレ
イヤの前にサイズが増大されるか、又は次のダウンサンプリングレイヤの前にサイズが低
減される場合に、リスケーリングは単一の方法でのみ適用されることは不要であることに
留意されたい。例えば、第 1 のダウンサンプリングレイヤの前に、入力のサイズはゼロに
よるパディングを用いて増大され、一方、第 2 のダウンサンプリングレイヤの前に、サイ

50

ズは、冗長な情報によるパディングを用いて、具体的には、反射パディング又は繰り返しパディングを用いて増大されることが提供されてもよい。

【0326】

入力がステップ2322でサイズ

【数171】

$$\bar{S} = nr_i$$

にリスケーリングされると、ダウンサンプリング操作はステップ2311に従って、しかし、サイズSを有する元の入力によらずに、サイズ

【数172】

$$\bar{S}$$

を有する変更された入力により実行され得る。いずれの場合にも、ダウンサンプリングは、元の入力サイズSよりも依然として小さいダウンサンプリングレイヤの出力のサイズ S_i をもたらすことが企図される。これは、ステップ2321で適用されるリスケーリングが、望ましくは、可能な限り少ない追加のエントリを加えるように実行されるか、あるいは、デフォルトに従って、いずれの場合にも入力サイズSを各々のダウンサンプリング比の最も近い小さい方の整数倍に低減することによって、実現されてもよい。

【0327】

例えば、各々のダウンサンプリングレイヤのダウンサンプリング比が2であり、入力のサイズが2の整数倍でない場合に、リスケーリングは、原則として、入力のサイズの2倍を有してもよい。これは、いずれの場合にも、リスケーリングされたサイズ

【数173】

$$\bar{S} = nr_i$$

をもたらすことになる。これは、しかし、サイズSと比較してサイズ

【数174】

$$\bar{S}$$

を大幅に増大させ、ダウンサンプリングを実行した後のサイズ S_i が実際に入力のサイズSよりも実際には小さくなるわけではない。そのような操作は、従って、望ましくは回避され、むしろ、入力がサイズ $lr_i < S < r_i(l+1)$ を有する場合に、

【数175】

$$\bar{S} = lr_i \text{ 又は } \bar{S} = (l+1)r_i$$

になるように、ステップ2321に従うリスケーリングが行われることが保証される。これは、望ましくは、ダウンサンプリング比の最も近い整数倍がリスケーリングのために選

択されることを意味する。

【0328】

l の値は、 $\text{ceil}()$ 若しくは $\text{floor}()$ 、又は各々の値を得ることを可能にする任意の関数を用いて、取得されてもよい。例えば、l を取得するために、 $\text{floor}(s/r)$ は、r を乗じられる場合に、S 以下でありながら S に最も近くなる整数値 l になるように計算され得る。従って、これにより

【数176】

$$\bar{S}$$

10

を取得することは、

【数177】

$$\bar{S}$$

が実際にダウンサンプリング比の最も近い小さい方の整数倍であることを確かにする。l 又は $l+1$ は、 $\text{ceil}(s/r)$ により取得されてもよい。この関数は、ダウンサンプリング比 r を乗じられる場合に、S 以上でありながら S に最も近くなる整数値 ($l+1$) になるように ($l+1$) を提供する。従って、これにより

【数178】

$$\bar{S}$$

20

を取得することは、実際にダウンサンプリング比の最も近い大きい方の整数倍であることを確かにする。

【0329】

例を与えるために、ダウンサンプリング比 $r=4$ であり、入力サイズは 513 であってよい。その場合に、l を計算するために $\text{floor}(s/r)$ を使用することは、最も近い小さい方の整数として $l=128$ をもたらす。 $\text{ceil}(s/r)$ を使用することは、 $(l+1)=129$ をもたらす。513 は ($4 \times 128 = 512$ であるから) r の最小整数倍に近いということで、入力サイズは 512 に低減されて、可能な限りほとんど情報は失われないことになる。しかし、デフォルトとして、サイズ S は、情報が失われないことを確かにするために 516 に増大されることも提供されてよい。

【0330】

他の実施形態では、2302のチェックステップ及び2310のステップを含み左分岐は存在しなくてもよい。かような場合に、リスケーリングステップ2321は、S が既にダウンサンプリング比 r_i の整数倍であるときに、S に等しいサイズ

【数179】

$$\bar{S}$$

30

40

を出力することができる。換言すれば、リスケーリング関数は、S が r_i の整数倍である可能性があるという事実を既に考慮に入れることができる。従って、チェックステップ2302は実施されなくてもよい。更なる実施形態では、2321のリスケーリングステップは、S が既にダウンサンプリング比 r_i の整数倍である場合でも、リスケーリングを実

50

行することができる。これは、例えば、入力のサイズの低減が重要である場合に、かつリサイジング操作が S よりも小さいサイズ

【数 180】

$$\bar{S}$$

を生成する場合に、有利になり得る。

【0331】

更に、この決定は明示的に行われなくてもよい。むしろ、例えばデフォルトによって、サイズ S を有する入力ダウンサンプリング比 r を有するダウンサンプリングレイヤについて取得されるとき、値 $\text{ceil}(s/r)r$ 又は $\text{floor}(s/r)r$ は決定されてよく、これに基づいて、リスケージングは、 $\text{ceil}(s/r)r$ 又は $\text{floor}(s/r)r$ が S に等しいかどうかを決定又は判断せずに実行される。これらの値が S に等しい場合に、リスケージングは、“ゼロ”リスケージング又は同一リスケージングであってよく、サイズ S を変更しない。これらの値が S と相違する場合にのみ、実際のリスケージングは適用される。

【0332】

本発明の実施形態に従って、方法 2300 は、各ダウンサンプリングレイヤの前に別々に実行される。具体的に、ステップ 2302 の決定及びステップ 2321 のリスケージングは、望ましくは図 19 に従うエンコーダのニューラルネットワーク内の他のダウンサンプリングレイヤの任意の他のダウンサンプリング比から独立している。例えば、図 19 のレイヤ 2110 がダウンサンプリングレイヤであり、レイヤ 2130 もダウンサンプリングレイヤである場合に、方法 2300 は、ダウンサンプリングレイヤ 2110 のダウンサンプリング比 r_{2110} のみを用いて、ダウンサンプリングレイヤ 2110 の前に実行される。その後、ニューラルネットワークのレイヤ 2120 によるビットストリームの処理後、それを更なるダウンサンプリングレイヤ 2130 で処理する前に、ビットストリーム 2103 は、その各々のサイズ及びダウンサンプリングレイヤ 2130 のダウンサンプリング比 r_{2130} を使用することによって、方法 2300 により処理される。

【0333】

それによって、リスケージング操作は、現在の入力を処理すべき特定のダウンサンプリングレイヤについてステップ 2302 の条件のみを考慮して実行され、ビットストリームによって表されるマトリクスへの追加のエントリの付加又は削除は、必要な分だけ行われる。それによって、図 19 に従ってエンコーダ 2100 による出力として供給される符号化されたビットストリーム 2105 のサイズは、可能な限り低減される。

【0334】

図 22 は、図 20 に従うデコーダのアップサンプリングレイヤの 1 つにより実行されるリサイジング又はリスケージング操作を参照する更なる実施形態を示す。

【0335】

符号化の場合と同様に、このリスケージングは、各アップサンプリングに対して別々に実行される。しかし、以下の議論から明らかなように、これは、特に、デコーダのニューラルネットワーク（図 20）の最終出力 2205 の目標サイズが既に固定されている場合に、後続のアップサンプリングレイヤから完全に独立している方法で実行され得ない。これに対する 1 つの例外は、リサイジングに関する全ての必要な情報がビットストリームで明示的にデコーダに供給される場合であってよい。

【0336】

それでもなお、1 つよりも多いアップサンプリングレイヤ、例えば、図 20 に示されるレイヤ 2200 及びレイヤ 2240 が存在する場合に、アップサンプリングは、ビットストリームがレイヤ 2210 によって処理された後、それがアップサンプリング 2220 によって処理される前に、適用されてよい。同様に、第 2 のアップサンプリングは、レイヤ

2 2 3 0 による処理の後、アップサンプリングレイヤ 2 2 4 0 による処理の前に、行われてよい。

【0 3 3 7】

図 2 2 及び方法 2 4 0 0 に関連して、入力は、サイズ T を有してステップ 2 4 0 1 で受け取られてよい。この入力の受け取りは、前のレイヤによる出力の供給に対応してよい。いずれの場合にも、ステップ 2 4 0 1 のビットストリームは、例えば、ビットストリームが水平成分及び垂直成分を有する 2 次元マトリクスを表す場合に水平次元などの少なくとも 1 つの次元で、サイズ T を有する。

【0 3 3 8】

アップサンプリングレイヤによりこの入力を処理する前に、ステップ 2 4 0 2 で決定が行われる。一般に、この決定は、アップサンプリングがサイズ T を有する入力に適用される場合に、対応する出力が意図された目標サイズに一致するサイズを有するかどうかを考慮すると見なされてよい。意図された目標サイズは、この特定のレイヤの処理の後に意図されているサイズであってよい。それは、しかし、ニューラルネットワークを通る符号化されたビットストリームの処理の終わりに最終的な復号されたビットストリームが取得されるまで、この特定のアップサンプリング操作後に実行されるべきアップサンプリングなどの追加情報を考慮した評価も含んでよい。かような場合に、ステップ 2 4 0 2 で、アップサンプリングレイヤ（アップサンプリング比 u を有する）による入力のアップサンプリング後に取得されたサイズ $T_i = u T$ が意図されたサイズ

【数 1 8 1】

$$\hat{T}$$

に等しいかどうか決定されてよく、この意図されたサイズは、

【数 1 8 2】

$$\hat{T} = \text{ceil}\left(\frac{T_{\text{target}}}{\prod_i u_i}\right)$$

によって与えられ、 u_i は、後続のアップサンプリングレイヤのアップサンプリング比である。全てのアップサンプリング比が等しい場合に、この積は u^N で置換されてよく、 N は後続のアップサンプリングレイヤの数である。

【0 3 3 9】

出力の意図されたサイズ又は目標サイズを

【数 1 8 3】

$$\hat{T} = 256$$

と見なす。出力のこの目標サイズは、デコーダ又は再構成された画像の目標出力サイズ T_{target} から取得されてよい。これは、例えば、1 0 2 4 であってよく、残り 2 つのアップサンプリングレイヤについては、アップサンプリング比 $u_i = u = 2$ である。

【数 1 8 4】

$$\text{ceil}\left(\frac{T_{\text{target}}}{\prod_i u_i}\right)$$

の値を計算すると、
【数 1 8 5】

$$\hat{T} = 256$$

が得られる。サイズ T_i が、例えば、2 5 8 である場合に、アップサンプリングレイヤへの入力が 1 2 9 のサイズを有しているので、サイズ T_i は
【数 1 8 6】

$$\hat{T}$$

10

に一致しないので、リスケーリングが必要になる。
【0 3 4 0】

この決定ステップ 2 4 0 2 は、上述されたように明示的な決定として提供されてよい。しかし、それは、記載された方法において別個のステップとして又は明示的なステップとして実行されなくてもよい。例えば、いくつかの実施形態において、デフォルトによって、リスケーリングは、サイズ T を
【数 1 8 7】

20

$$\text{ceil}\left(\frac{T_{\text{target}}}{\prod_i u_i}\right)$$

に変更するために適用される。 T が既に
【数 1 8 8】

$$\text{ceil}\left(\frac{T_{\text{target}}}{\prod_i u_i}\right)$$

30

である場合に、これによりサイズ T は変更されない。他方で、 T が既に
【数 1 8 9】

$$\text{ceil}\left(\frac{T_{\text{target}}}{\prod_i u_i}\right)$$

40

に等しくない場合に、サイズのリサイジング（例えば、パディングによる増大又はクロッピングによる低減）が提供されてよい。しかし、この実施形態では、ステップ 2 4 0 2 でのような別個の決定は実行されず、このチェックは、リスケーリングを適用する計算の暗黙的な一部である。

【0 3 4 1】

一般に、意図されたサイズ

【数 1 9 0】

$$\hat{T}$$

50

は、形式

【数 1 9 1】

$$\hat{T} = f(u_i, u_j, u_k, \dots, T_{target}) \text{ 又は } \hat{T} = f(u_i, u_j, u_k, \dots, T_{target}, \bar{T})$$

のリサイジング関数によって取得することができ、 $u_i, u_j, u_k, \dots$ は、入力の処理順序における後続のアップサンプリングレイヤのアップサンプリング比である。

【数 1 9 2】

$$f(u_i, u_j, u_k, \dots, T_{target})$$

10

の他の例は、

【数 1 9 3】

$$\hat{T} = \text{floor}\left(\frac{T_{target}}{\prod_i u_i}\right)$$

20

であってよい。

【0 3 4 2】

従って、後続のアップサンプリングレイヤによる処理のために現在のアップサンプリングレイヤによって供給される出力の実際のサイズは、必要になる後続のアップサンプリングに依存するが、既に実行されたアップサンプリングには依存しない。しかし、実行されたかもしれない前のアップサンプリングプロセスが対応する決定により取得され得ることは明らかであり、その場合、当然に、先行するレイヤの一部も考慮に入れられる。それによって、既に入力のサイズTは、少なくとも何らかの形で完全なニューラルネットワークのアップサンプリング比に依存するようになった。

30

【0 3 4 3】

Ceil関数は、余りのない除算を結果として与える関数である。この演算の、具体的にはステップ2 4 0 2の結果は、

【数 1 9 4】

$$T_i > \hat{T} \text{ 又は } T_i < \hat{T} \text{ 又は } T_i = \hat{T}$$

を示し得る。

40

【0 3 4 4】

場合

【数 1 9 5】

$$T_i = \hat{T}$$

は、サイズTを有する入力のこれ以上のリスケージングがアップサンプリング後に不要であることが決定されるステップ2 4 1 0の決定をもたらし、ステップ2 4 1 1で、入力を

50

サイズ T からサイズ
【数 1 9 6】

$$\bar{T}$$

(アップサンプリングレイヤ i の出力として図 2 2 では T_i とも呼ばれる) へアップサンプリングするアップサンプリング操作が実行され得、サイズ
【数 1 9 7】

10

$$\bar{T}$$

を有する出力は、例えば、更なるアップサンプリングレイヤによる更なる処理のために、ステップ 2 4 1 2 で供給され得る。

【0 3 4 5】

他方で、ステップ 2 4 0 2 において、サイズ

【数 1 9 8】

$$\bar{T}$$

20

が

【数 1 9 9】

$$\hat{T}$$

に等しくないことが決定される場合に (2 4 2 0)、サイズ T_i を有する出力のリスケーリングが、

30

【数 2 0 0】

$$T_i = \hat{T}$$

を取得するためにステップ 2 4 2 1 が必要である。このリスケーリングは、補間又は 0 によるパディング又は繰り返しパディング又は反射パディングなどのような上記の手段の 1 つを適用することによって、サイズ T_i をサイズ

【数 2 0 1】

40

$$\hat{T}$$

に増大させることを有してもよい。代替的に、

【数 2 0 2】

$$T_i > \hat{T}$$

の場合に、リスケーリングは、サイズ T_i を低減するカッティング、クロッピング又は任

50

意の他の操作を有してもよく、それにより、サイズ T_i は意図されたサイズ

【数 2 0 3】

$$\hat{T}$$

と一致する。

【0 3 4 6】

リスケーリングが必要であったならば、リスケーリングされた出力は、次いで、例えば、他のアップサンプリングレイヤによる後続の処理のために、出力として供給され得る。

【0 3 4 7】

ステップ 2 4 0 2 でどのような決定がなされたとしても、ステップ 2 4 1 1 でのアップサンプリングの適用後に供給される出力は、適切なサイズを有する出力になるように後続のレイヤによって更に処理され得るサイズ

【数 2 0 4】

$$\bar{T}$$

を有することになる。

【0 3 4 8】

2 4 0 2 のチェックステップは任意であってよく、リサイジング（リスケーリング）操作／関数に組み込まれ得る。この場合に、リサイジング関数は、 T_i に等しい

【数 2 0 5】

$$\hat{T}$$

を出力し得る。これは、先行するアップサンプリングレイヤの出力を次のアップサンプリングレイヤによって処理する前に、サイズ T が

【数 2 0 6】

$$\hat{T} = \text{ceil}\left(\frac{T_{\text{target}}}{\prod_i u_i}\right)$$

にリスケーリングされる場合に当てはまる。いくつかの場合に、

【数 2 0 7】

$$T = \text{ceil}\left(\frac{T_{\text{target}}}{\prod_i u_i}\right)$$

の場合に、これは、出力のサイズが変更されない“ゼロ”リスケーリング又は同一リスケーリングをもたらす。これは、チェックステップがリサイジング関数 $f(u_i, u_j, u_k, \dots, T_{\text{target}})$ の実行を既に含む場合に特に有用であり得る。かような場合に、チェックステップは、リスケーリングから独立して実行されないが、リスケーリングの部分を既に形成している暗黙的なステップであると思なされてよい。例えば、図 2 2 で、関数 $f()$ は

【数 2 0 8】

$$\text{ceil}\left(\frac{T_{\text{target}}}{\prod u_{j>i}}\right)$$

に等しく、それは、リサイジングが必要であるか否かを決定するために実行される必要がある。従って、いくつかの実施において、このチェックステップは提供されなくてもよく、リサイジング操作（2 4 2 1）の部分として解釈され得る。

【0 3 4 9】

しかし、チェックステップが同じ方法でリサイジング操作の実行を含まない他の実施では、チェックステップは別個に実施されてもよい。一例は、チェックステップ 2 4 0 2 がレイヤ ID（処理順序におけるレイヤ番号）をチェックすることを含み得ることであってよい。このチェック動作はリサイジング動作を含まない。従って、チェック動作はこの例では必要である。

【0 3 5 0】

次のアップサンプリングレイヤでは、対応する決定が行われてよく、リスケーリング（リサイジング）は実行されてもされなくてもよい。

【0 3 5 1】

リサイジング及びリスケーリングという用語は同義的に使用され、どちらも、ニューラルネットワークのダウンサンプリングレイヤ又はアップサンプリングレイヤによって入力を実際に処理することの外側で入力のサイズを変更することを指す。

【0 3 5 2】

上記の実施形態では、決定 2 4 0 2 は、アップサンプリングレイヤによる入力の処理の前に行われている。しかし、この決定は、アップサンプリングが実行されると、又は後続のアップサンプリングレイヤの直前に行われることも提供され得る。いずれの場合にもリスケーリングは、アップサンプリングが実行された後に実行されるべきである。

【0 3 5 3】

決定 2 4 0 2 を行うために、デコーダ及び具体的にはニューラルネットワークは、異なる方法で取得され得る追加情報を知っている必要があり得る。具体的には、リスケーリングが実際に適用されるようにリスケーリングが適用されるべきか否かの決定は、先に与えられた式を考慮して、符号化されたビットストリームを復号するとき実際に取得されるべきである目標サイズに依存してよい。例えば、復号されたビットストリームの最終的な目標サイズが 2 次元マトリクスとして表現されるときに 1024×512 である場合に、異なるリスケーリングが、ニューラルネットワーク内の特定の点での 2048×1024 と比較して必要である場合がある。従って、ニューラルネットワークの各レイヤ又はニューラルネットワークそれ自体又はデコーダは、例えば、目標サイズを知っている場合がある。

【0 3 5 4】

代替的に、又は追加的に、ニューラルネットワークが、各アップサンプリングレイヤ i 後に出力されるべきである出力サイズ

【数 2 0 9】

$$\hat{T}_i$$

を

知っている場合が望ましい可能性がある。これは、目標サイズの知識を潜在的にもはや

不要にする。更に、リスケーリングが必要であるかどうかを決定するたびに、上記の天井関数を実行することは不要になる。

【 0 3 5 5】

代替的に、各々の出力サイズ

【数 2 1 0】

$$\hat{T}_i$$

は、ルックアップテーブル LUT のような所与のフォーマットで既に与えられている場合がある。そのルックアップテーブルで、サイズ

【数 2 1 1】

$$\hat{T}_i$$

は、それらがマトリクスからフェッチされ得るように与えられ得る。マトリクスのエントリは、目標サイズ及びアップサンプリングレイヤ i の数に応じて定義されてよく、それにより、 i 及び目標サイズが知られている場合に、

【数 2 1 2】

$$\hat{T}_i$$

は、LUT 内の各々の位置を指し示すポインタを使用することによって LUT からフェッチされ得る。

【 0 3 5 6】

更に、リスケーリングが必要であるか否かをステップ 2402 で確実に決定するために、この決定に影響を与えることになるアップサンプリング比も、ニューラルネットワークを通る符号化されたビットストリームの処理におけるこの時点で知られている必要がある場合がある。

【 0 3 5 7】

そのような情報は、既に上述されたように、異なる方法で供給され得る。本発明のいくつかの実施形態の利点が符号化されたビットストリームの削減にあるので、符号化されたビットストリームがサイズを増大されないように、符号化されたビットストリームから独立している可能な限り多くの情報を提供することが好ましいが、依然として必須ではない。

【 0 3 5 8】

従って、例えば、各アップサンプリングレイヤの後に取得されるべきである中間目標サイズ

【数 2 1 3】

$$\hat{T}_i$$

の目標サイズに関する情報、又はアップサンプリング比に関する情報は、符号化されたビットストリームの部分でない別個のビットストリームで供給されてよい。代替的に、この必要な情報も、ビットストリームの全体のサイズが小さく保たれ得る限りは、ビットストリームで供給され得る。

【 0 3 5 9】

更なる代替案は、特にアップサンプリングレイヤのアップサンプリング比及び／又はデコーダ上に既に存在する目標サイズに関して、この情報の一部又は全部を供給することである。このように、デコーダは、固定された（又は少なくとも既知の）アップサンプリングを特定の順序で適用することによって、入力から固定目標サイズを持つ復号されたビットストリームを取得するように特に設計されている。従って、ビットストリームのサイズを可能な限り小さく保ちながら、ビットストリームに含める必要がない。しかし、デコーダが異なる目標サイズの出力を供給することができる場合に、ビットストリームには、その目標サイズを示すフラグ又はポインタが含まれてもよい。

【0360】

この情報は、どの種類のリスケージングが適用されるべきかを示す情報（図13から15で説明される）とともに、図13から15に関してやはり上で説明されたように、ビットストリーム自体又は別のビットストリームに含まれ得る。具体的に、ビットストリーム（又は更なるビットストリーム）は、上述された`methodIdx`、`SCIdx`、および`b意RFIdx`のような指示を含んでよい、これらは、エンコーダによって、どのような種類のリスケージングを適用すべきかを決定するために使用されてよい。

【0361】

図23は、図21のステップ2321で入力のサイズを増大させるべきか又はダウンサンプリングを実行する前に入力のサイズを低減させるべきかどうかを決定する更なる方法2500を示す。

【0362】

方法2500はステップ2501から始まり、ステップ2501で、受け取られた入力サイズ $S \neq l_{rm}$ を有することが既に決定されているか又はステップ2320で決定されている。

【0363】

ステップ2501でこの決定がなされると、次のステップ2502は、入力のより詳細な評価を有することができる。具体的に、それは、サイズ S が $l_{rm} \leq S \leq r_m(l+1)$ であるとの決定を有することができる。

【0364】

当業者には明らかなように、ステップ2501は不要であり無くてもよい。

【0365】

これが決定されると、条件がステップ2530使用され得、この結果に基づいて、ステップ2510又は2520で、異なる更なるステップが行われる。これは、条件に対するサイズ S の評価が、サイズが

【数214】

$$\bar{S} = r_m(l+1)$$

に増大されるとの結論を有する第1の結果をもたらすことができ、一方で、条件の評価の第2の結果2520が、ダウンサンプリングがステップ2530で実行される前にステップ2521で、サイズの

【数215】

$$\bar{S} = l_{rm}$$

への低減をもたらすことができる、ことを意味する。

【0366】

例として、条件は、 $\text{Min}(|S - l_{rm}|, |S - r_m(l+1)|)$ を利用することができ、 l 及び $l+1$ の値は、図21に関して上述されたものに従って決定され得る。

具体的に、 l は、 $l = \text{floor}(s / r_m)$ から取得されてよく、 $l + 1$ は、この結果に1を加えること又は $l + 1 = \text{ceil}(s / r_m)$ を計算することによって取得されてよい。 l を取得すると、又は上記のMin関数において直接に $\text{floor}(s / r_m)$ 及び $\text{ceil}(s / r_m)$ を使用することによって、 $S - l r_m$ の差又は次に大きい整数倍数 $S - r_m(l + 1)$ に対する差がより小さいかが決定されるので、入力に提供される追加情報の数が最小限であること、又は元の入力から削除される情報の数が最小限であることを必要とする。

【0367】

例えば、一例として上で既に参照されたように、 $r_m = 4$ 及び $S = 513$ の場合に、 $l = 128$ であり、入力のサイズ S は、 r_m の最も近い大きい方の整数倍よりも r_m の最も近い小さい方の整数倍により近い。 S が $S = 515$ になる場合に、入力のサイズは、 r_m の最も近い小さい方の整数倍よりも r_m の最も近い大きい方の整数倍により近くなる。

【0368】

上記のMin評価に基づいて、実行される必要がある計算ステップの数はこのようにして決定され、最小限に保たれ得る。

【0369】

上記のMin評価を利用する条件は、リスケーリング操作を実行するための計算ステップを小さく保つことであってよい。例えば、1次元の場合に、ダウンサンプリング比 r_m が4であり、サイズ $S = k r_m + 1$ であるならば、サイズ S を

【数216】

$$\bar{S} = k r_m$$

に低減し、それを

【数217】

$$\bar{S} = (k + 1) r_m$$

に増大しない方が計算効率が良い可能性がある。

【0370】

リスケーリングで実行される必要がある操作の数がサイズ S を増大させること及びサイズ S を低減させることに関して等しい場合に、条件は更に、付加される冗長な情報を可能な限り少なくすること、又は入力から削除される情報を可能な限り少なくなることのどちらかであってよい。上記の場合に、サイズ $S = k_m + 2$ に関して、サイズを、ゼロをパディングすることによって

【数218】

$$\bar{S} = (k + 1) r_m$$

に増大させることが決定されてもよい。これは、冗長な情報の量を増大させるが、情報の損失をもたらさない。

【0371】

一般に、ステップ2503の条件に対して入力のサイズ S を評価することによって、サイズ $S - r_m(l + 1)$ の差が $S - l r_m$ よりも小さいと決定される場合に、例えば、ダウンサンプリングを実行する前に、結果2510から、サイズ S をサイズ

【数219】

10

20

30

40

50

$$\bar{S} = r_m(l + 1)$$

に増大させる方が計算効率が良い可能性があるとして決定される。

【0372】

他方で、差 $S - l r_m$ が差 $S - r_m(l + 1)$ よりも小さいと決定され、例えば、サイズ S をサイズ

【数220】

$$\bar{S} = r_m l$$

10

に低減させる方が計算効率が良い可能性があるとして決定される場合に、サイズ S は

【数221】

$$\bar{S} = r_m l$$

20

に低減され得る。

【0373】

しかし、これは、ダウンサンプリングを実行する前に入力サイズを増大又は低減すべきかどうかという評価が行われ得る条件を特に実施するために使用される1つの実施形態にすぎない。この点で、入力サイズ

【数222】

$$\bar{S}$$

30

が実際に小さければ小さいほど、ダウンサンプリングを実行するために通常必要な計算はますます少なくなることも考慮に入れることができる、また、適切と見なされる如何なる他の条件も、この決定を行うために使用されてよい。

【0374】

上記を考慮して、いずれの場合にも、ダウンサンプリングレイヤによって入力を処理する前の入力のリスキューリングは、特定の次元での入力サイズの増大又は低減をもたらすことになる。

【0375】

上記の例では、任意のダウンサンプリング比及びアップサンプリング比が使用された。しかし、いくつかの好適な実施形態は、エンコーダのニューラルネットワークの全てのダウンサンプリングレイヤのダウンサンプリング比が同じであってもよいことを有し得る。同様に、又は代替的に、デコーダのニューラルネットワークの全てのアップサンプリングレイヤのアップサンプリング比は同じであってもよい。本発明は、しかしながら、この点で制限されず、アップサンプリング及び／又はダウンサンプリング比は、想到可能である如何なる整数値もとることができる。しかし、好適な実施形態は、2及び10の範囲内にあるアップサンプリング及び／又はダウンサンプリング比を参照する。

40

【0376】

ここで記載されている符号化方法又はエンコーダの実施形態において、例えば、NNによって出力されるビットストリーム出力は、例えば、NNの最後のサブネットワーク又は

50

ネットワークレイヤの出力又はビットストリームであってよい。

【0377】

ここで記載されている符号化方法又はエンコーダの更なる実施形態においては、例えば、NNによって出力されるビットストリーム出力は、例えば、サブビットストリームとも呼ばれ得る2つのビットストリーム、例えば、ビットストリーム又はサブビットストリーム ビットストリーム1及びビットストリーム2、あるいは、より一般的には、第1ビットストリーム又は第1サブビットストリーム及び第2ビットストリーム又は第2サブビットストリーム（例えば、NNの各々のサブネットワークによって生成及び／又は出力される各ビットストリーム又はサブビットストリーム）によって決定されるか又はそれらを含むビットストリームであってよい。どちらのビットストリーム又はサブビットストリームも別々に伝送若しくは記憶されるか、あるいは、1つのビットストリームとして結合、例えば、多重化されてよい。

10

【0378】

ここで記載されている符号化方法又はエンコーダの更に別の実施形態においては、例えば、NNによって出力されるビットストリーム出力は、例えば、サブビットストリームとも呼ばれ得る2つよりも多いビットストリーム、例えば、第1ビットストリーム／サブビットストリーム、第2ビットストリーム／サブビットストリーム、第3ビットストリーム／サブビットストリーム、及び任意に更なるビットストリーム／サブビットストリーム（例えば、NNの各々のサブネットワークによって生成及び／又は出力される各ビットストリーム／サブビットストリーム）によって決定されるか又はそれらを含むビットストリームであってよい。ビットストリーム／サブビットストリームは、別々に伝送若しくは記憶されるか、あるいは、1つのビットストリーム又は1つよりも多い複合ビットストリームとして結合、例えば、多重化されてよい。

20

【0379】

ここで記載されている復号化方法又はデコーダの実施形態において、例えば、NNによって受け取られる受信ビットストリームは、例えば、NNの最初のサブネットワーク又はネットワークレイヤの入力として使用されてよい。

【0380】

ここで記載されている復号化方法又はデコーダの更なる実施形態においては、受信ビットストリームは、例えば、サブビットストリームとも呼ばれ得る2つのビットストリーム、例えば、ビットストリーム／サブビットストリーム ビットストリーム1及びビットストリーム2、あるいは、より一般的には、第1ビットストリーム／サブビットストリーム及び第2ビットストリーム／サブビットストリーム（例えば、NNの各々のサブネットワークによって受信及び／又は処理される各ビットストリーム／サブビットストリーム）によって決定されるか又はそれらを含むビットストリームであってよい。どちらのビットストリーム又はサブビットストリームも別々に受信若しくは記憶されるか、あるいは、1つのビットストリームとして結合、例えば、多重化され、ビットストリーム／サブビットストリームを取得するよう逆多重化されてよい。

30

【0381】

ここで記載されている復号化方法又はデコーダの更に別の実施形態においては、受信ビットストリームは、例えば、サブビットストリームとも呼ばれ得る2つよりも多いビットストリーム、例えば、第1ビットストリーム／サブビットストリーム、第2ビットストリーム／サブビットストリーム、第3ビットストリーム／サブビットストリーム、及び任意に更なるビットストリーム／サブビットストリーム（例えば、NNの各々のサブネットワークによって受信及び／又は処理される各ビットストリーム／サブビットストリーム）によって決定されるか又はそれらを含むビットストリームであってよい。ビットストリーム／サブビットストリームは、別々に受信若しくは記憶されるか、あるいは、1つのビットストリームとして結合、例えば、多重化され、ビットストリーム／サブビットストリームを取得するよう逆多重化されてよい。

40

【0382】

50

数学演算子

本願で使用される数学演算子は、Cプログラミング言語で使用されているものと類似している。しかし、整数除算及び算術シフト演算の結果は、より厳密に定義され、指数及び実数値除算のような、更なる演算が定義される。番号付け及び計数の規則は、一般的に0から始まり、例えば、「第1」は0番目と等価であり、「第2」は1番目と等価である、など。

【0383】**算術演算子**

以下の算術演算が、次のように定義される：

【表1】

+	加算	10
-	減算（2引数演算子として）又はマイナス（単項前置演算子として）	
*	行列乗算を含む乗算	
x^y	べき乗。 x を y の累乗に指定する。他のコンテキストでは、このような表記は、べき乗としての解釈を目的としない上付き文字に使用される。	20
/	結果をゼロに向かって切り捨てる整数除算。例えば、 $7/4$ 及び $-7/-4$ は、1に切り捨てられ、 $-7/4$ 及び $7/-4$ は-1に切り捨てられる。	
÷	切り捨て又は丸めが意図されない数式の除算を示すために使用される。	
$\frac{x}{y}$	切り捨て又は丸めが意図されない数式の除算を示すために使用される。	
$\sum_{i=x}^y f(i)$	i が x から y までの全ての整数値をとる $f(i)$ の合計。	30
$x \% y$	法。 x を y で割った余りであり、 $x \geq 0$ 及び $y > 0$ である整数 x 及び y に対してのみ定義される。	

【0384】**論理演算子**

以下の算術演算子が、次のように定義される：

$x \&\&y$ x 及び y のブール論理“AND”

$x \mid \mid y$ x 及び y のブール論理“OR”

! ブール論理“NOT”

$x ? y : z$ x がTRUEであるか、又は0に等しくない場合には、 y の値として評価し、そうでない場合には、 z の値として評価する。

【0385】**関係演算子**

以下の関係演算子が、次のように定義される：

> ~よりも大きい

>= ~以上

10

20

30

40

50

< ~よりも小さい
 <= ~以下
 == ~に等しい
 != ~に等しくない。

【0386】

関係演算子が、値“NA”(not applicable)を割り当てられているシンタックス要素又は変数に適用される場合に、値“NA”は、そのシンタックス要素又は変数のディスティンクト値として扱われる。値“NA”は、如何なる他の値にも等しくないと見なされる。

【0387】

ビット演算子

以下のビット演算子が、次のように定義される：

& ビット単位“AND”。整数引数に作用する場合に、整数値の2の補数表現に作用する。他の引数よりも少ないビットを含む2進引数に作用する場合に、より短い方の引数は、0に等しい更なる有効ビットを加えることによって拡張される。

| ビット単位“OR”。整数引数に作用する場合に、整数値の2の補数表現に作用する。他の引数よりも少ないビットを含む2進引数に作用する場合に、より短い方の引数は、0に等しい更なる有効ビットを加えることによって拡張される。

^ ビット単位“XOR”。整数引数に作用する場合に、整数値の2の補数表現に作用する。他の引数よりも少ないビットを含む2進引数に作用する場合に、より短い方の引数は、0に等しい更なる有効ビットを加えることによって拡張される。

$x \gg y$ y 個の2進桁による x の2の補数整数表現の算術右シフト。この関数は、 y の非負整数値についてのみ定義される。右シフトの結果として最上位ビット(MSB)へシフトされたビットは、シフト演算の前の x のMSBに等しい値を有する。

$x \ll y$ y 個の2進桁による x の2の補数整数表現の算術左シフト。この関数は、 y の非負整数値についてのみ定義される。左シフトの結果として最下位ビット(LSB)へシフトされたビットは、0に等しい値を有する。

【0388】

代入演算子

以下の算術演算子が、次のように定義される：

= 代入演算子

++ インクリメント、すなわち、 $x++$ は、 $x = x + 1$ と等価であり、アレイインデックスで使用される場合には、インクリメント演算の前の変数の値として評価する。

-- デクリメント、すなわち、 $x--$ は、 $x = x - 1$ と等価であり、アレイインデックスで使用される場合には、デクリメント演算の前の変数の値として評価する。

+= 指定された量によるインクリメント、すなわち、 $x += 3$ は、 $x = x + 3$ と等価であり、 $x += (-3)$ は、 $x = x + (-3)$ と等価である。

-= 指定された量によるデクリメント、すなわち、 $x -= 3$ は、 $x = x - 3$ と等価であり、 $x -= (-3)$ は、 $x = x - (-3)$ と等価である。

【0389】

範囲表記

以下の表記は、値の範囲を指定するために使用される：

$x = y \dots z$ x 、 y 及び z は整数であり、 z は y よりも大きいとして、 x は、 y 以上 z 以下の整数値を呈する。

【0390】

数学関数

以下の数学関数が定義される：

【数223】

10

20

30

40

50

$$\text{Abs}(x) = \begin{cases} x & ; \quad x \geq 0 \\ -x & ; \quad x < 0 \end{cases}$$

A s i n (x) -1.0 以上 1.0 以下の範囲内にある引数 x に作用して、出力値がラジアン単位で $-\pi \div 2$ 以上 $\pi \div 2$ 以下の範囲内にある三角関数の逆正弦関数

A t a n (x) 引数 x に作用して、出力値がラジアン単位で $-\pi \div 2$ 以上 $\pi \div 2$ 以下の範囲内にある三角関数の逆正接関数

【数 2 2 4】

$$\text{Atan2}(y, x) = \begin{cases} \text{Atan}\left(\frac{y}{x}\right) & ; \quad x > 0 \\ \text{Atan}\left(\frac{y}{x}\right) + \pi & ; \quad x < 0 \ \&\& \ y \geq 0 \\ \text{Atan}\left(\frac{y}{x}\right) - \pi & ; \quad x < 0 \ \&\& \ y < 0 \\ +\frac{\pi}{2} & ; \quad x = 0 \ \&\& \ y \geq 0 \\ -\frac{\pi}{2} & ; \quad \text{上記以外} \end{cases}$$

10

20

C e i l (x) x 以上の最小の整数

C l i p 1_y (x) = **C l i p 3** (0, ($1 << \text{B i t D e p t h}_y$) - 1, x)

C l i p 1_c (x) = **C l i p 3** (0, ($1 << \text{B i t D e p t h}_c$) - 1, x)

【数 2 2 5】

$$\text{Clip3}(x, y, z) = \begin{cases} x & ; \quad z < x \\ y & ; \quad z > y \\ z & ; \quad \text{上記以外} \end{cases}$$

30

C o s (x) ラジアン単位で引数 x に作用する三角関数の余弦関数

F l o o r (x) x 以下の最大の整数

【数 2 2 6】

$$\text{GetCurrMsb}(a, b, c, d) = \begin{cases} c + d & ; \quad b - a \geq d / 2 \\ c - d & ; \quad a - b > d / 2 \\ c & ; \quad \text{上記以外} \end{cases}$$

40

L n (x) x の自然対数 (底 e の対数、ここで、 e は、自然対数の底定数 $2.718281828 \dots$ である)

L o g 2 (x) x の底 2 の対数

L o g 10 (x) x の底 10 の対数

【数 2 2 7】

50

$$\text{Min}(x, y) = \begin{cases} x & ; \quad x \leq y \\ y & ; \quad x > y \end{cases}$$

$$\text{Max}(x, y) = \begin{cases} x & ; \quad x \geq y \\ y & ; \quad x < y \end{cases}$$

10

$$\text{Round}(x) = \text{Sign}(x) * \text{Floor}(\text{Abs}(x) + 0.5)$$

$$\text{Sign}(x) = \begin{cases} 1 & ; \quad x > 0 \\ 0 & ; \quad x == 0 \\ -1 & ; \quad x < 0 \end{cases}$$

S i n (x) ラジアン単位で引数 x に作用する三角関数の正弦関数

20

S q r t (x) = $\sqrt{x}$

S w a p (x , y) = (y , x)

T a n (x) ラジアン単位で引数 x に作用する三角関数の正接関数。

【 0 3 9 1 】

演算の優先順位

式中の優先順位がパラメータの使用によって明示的に示されていない場合に、以下の規則が適用される：

- ・優先度のより高い演算は、優先度のより低い演算の前に評価される。
- ・同じ優先度の演算は、左から右に順番に評価される。

【 0 3 9 2 】

30

以下の表は、最高から最低までの演算の優先度を定め、表中のより高い位置は、より高い優先度を示す。

【 0 3 9 3 】

C プログラミング言語でも使用される演算については、本明細書で使用されている優先順位は、C プログラミング言語で使用されているものと同じである。

40

50

【表 2】

表：最高（表の上部）から最低（表の下部）までの演算の優先度

演算（オペランド x 、 y 及び z を使用）	
" $x++$ ", " $x--$ "	
" $!x$ ", " $-x$ "（単項前置演算子として）	
x^y	10
" $x * y$ ", " x / y ", " $x \div y$ ", " $\frac{x}{y}$ ", " $x \% y$ "	
" $x + y$ ", " $x - y$ "（2 引数演算子として）, " $\sum_{i=x}^y f(i)$ "	
" $x \ll y$ ", " $x \gg y$ "	
" $x < y$ ", " $x \leq y$ ", " $x > y$ ", " $x \geq y$ "	20
" $x == y$ ", " $x != y$ "	
" $x \& y$ "	
" $x y$ "	
" $x \&\& y$ "	
" $x y$ "	30
" $x ? y : z$ "	
" $x..y$ "	
" $x = y$ ", " $x += y$ ", " $x -= y$ "	

【0394】

論理演算のテキスト記述

テキスト中で、次の形式で数学的に記述されることになる論理演算のステートメント：

$i \text{ f (条件 0)}$

ステートメント 0

$e \text{ l s e } i \text{ f (条件 1)}$

ステートメント 1

...

$e \text{ l s e } / * \text{残りの条件に関する説明的所見} * /$

ステートメント n

は、次の様態で記述されてもよい：

... 次のように / ... 次が適用される：

10

20

30

40

50

- － 条件 0 の場合に、ステートメント 0
- － そうではなく、条件 1 の場合に、ステートメント 1
- － . . .
- － 上記以外の場合に（残りの条件に関する説明的所見）、ステートメント n

【0395】

テキスト中の夫々の「. . . の場合に、. . . そうではなく、. . . の場合に、. . . 上記以外の場合に、. . .」のステートメントは、「. . . の場合に」の直ぐ後に続く「. . . 次のように」又は「. . . 次が適用される」により導入される。「. . . の場合に、. . . そうではなく、. . . の場合に、. . . 上記以外の場合に、. . .」の最後の条件は常に「上記以外の場合に、. . .」である。交互配置された「. . . の場合に、. . . そうではなく、. . . の場合に、. . . 上記以外の場合に、. . .」のステートメントは、「. . . 次のように」又は「. . . 次が適用される」を終わりの「上記以外の場合に、. . .」と適合させることによって識別され得る。

10

【0396】

テキスト中で、次の形式で数学的に記述されることになる論理演算のステートメント：

if (条件 0 a & & 条件 0 b)

ステートメント 0

else if (条件 1 a || 条件 1 b)

ステートメント 1

. . .

else

ステートメント n

20

は、次の様態で記述されてもよい：

. . . 次のように / . . . 次が適用される：

- － 次の全ての条件が真である場合に、ステートメント 0：
 - － 条件 0 a
 - － 条件 0 b
- － そうではない場合に、次の条件の 1 つ以上が真である場合に、ステートメント 1：
 - － 条件 1 a
 - － 条件 1 b
- － . . .
- － 上記以外の場合に、ステートメント n

30

【0397】

テキスト中で、次の形式で数学的に記述されることになる論理演算のステートメント：

if (条件 0)

ステートメント 0

if (条件 1)

ステートメント 1

は、次の様態で記述されてもよい：

条件 0 の場合に、ステートメント 0

40

条件 1 の場合に、ステートメント 1

50

【図面】

【図 1 A】

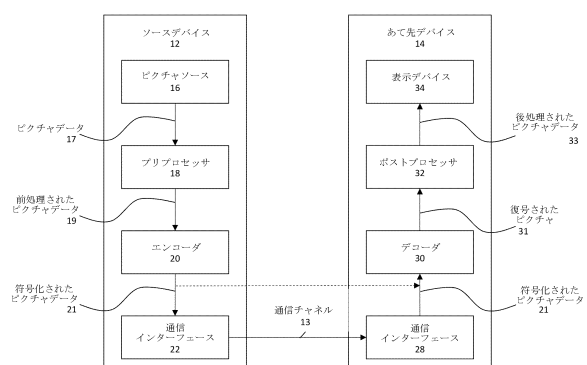


Fig. 1A

【図 1 B】

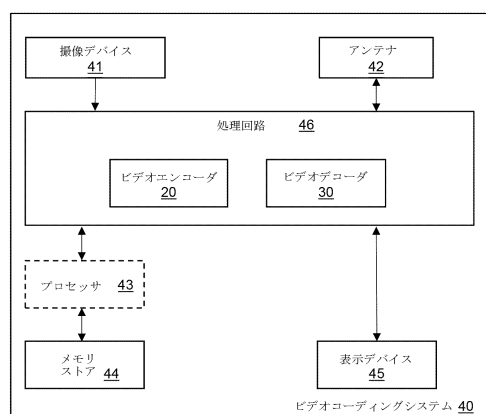
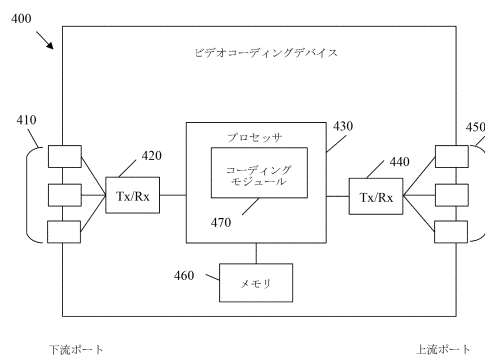


Fig. 1B

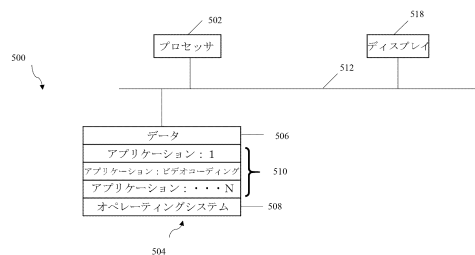
【圖 2】

Fig. 2



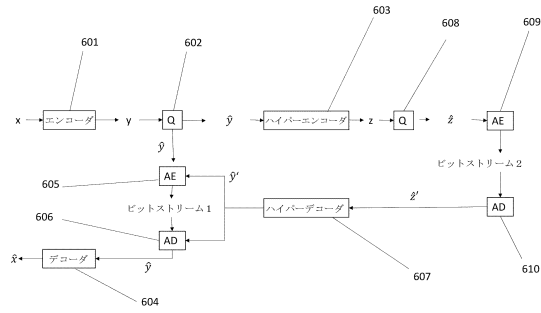
【図 3】

Fig. 3



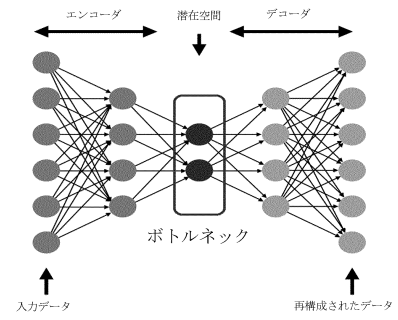
【図 4】

Fig. 4



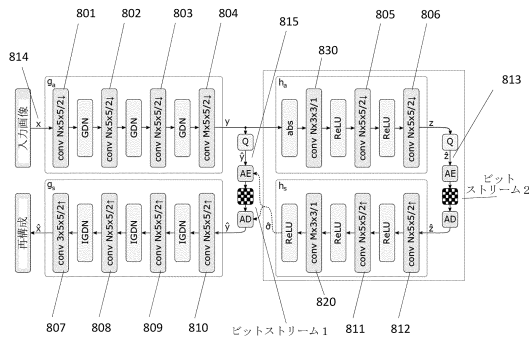
【図 5】

Fig. 5



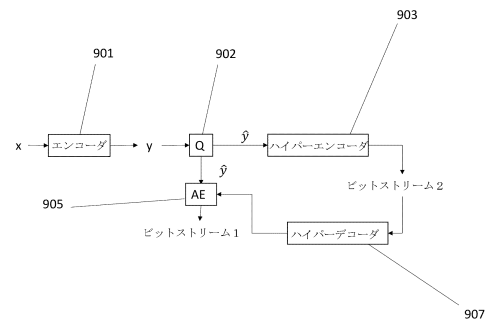
【図 6】

Fig. 6



【図 7】

Fig. 7



10

20

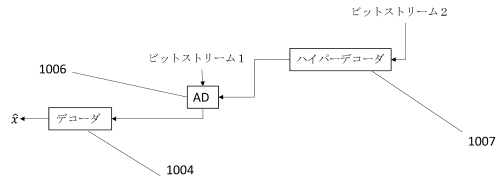
30

40

50

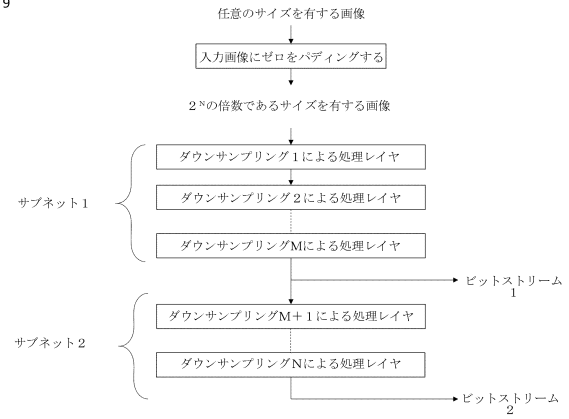
【図 8】

Fig. 8



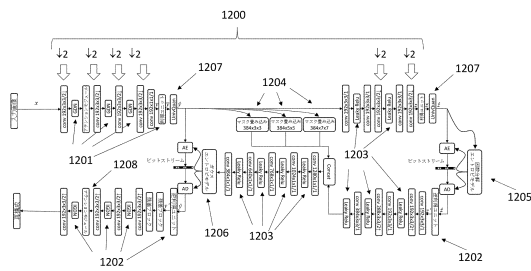
【図 9】

Fig. 9



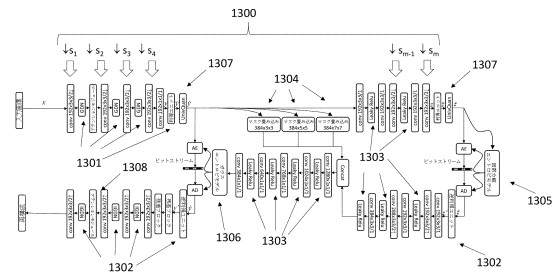
【図 10】

Fig. 10



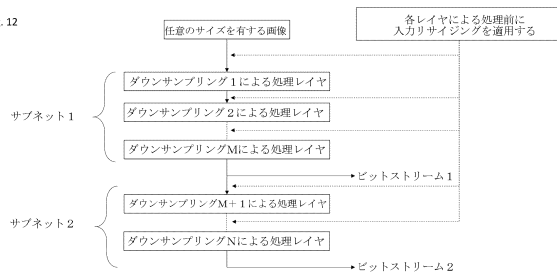
【図 11】

Fig. 11



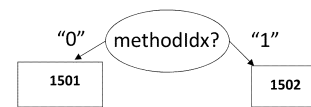
【図 12】

Fig. 12



【図 13】

Fig. 13



10

20

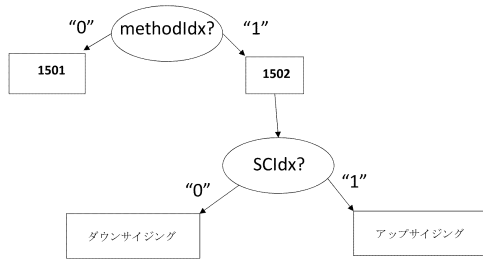
30

40

50

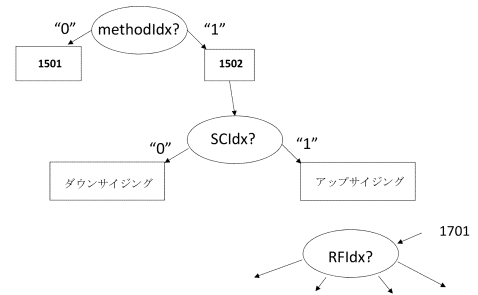
【図 14】

Fig. 14



【図 15】

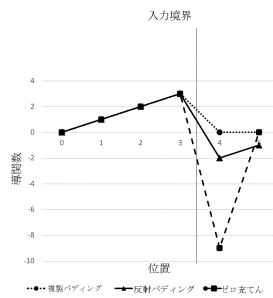
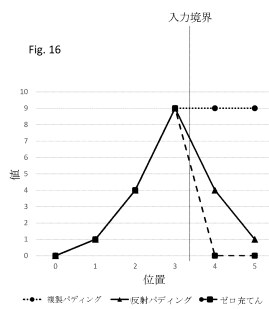
Fig. 15



10

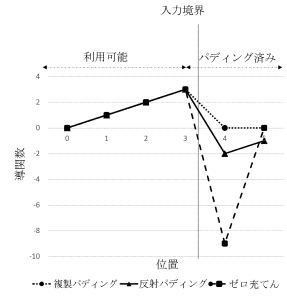
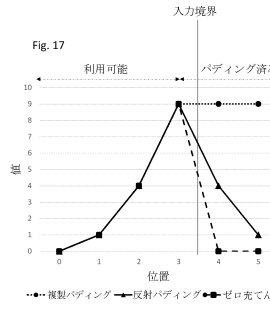
【図 16】

Fig. 16



【図 17】

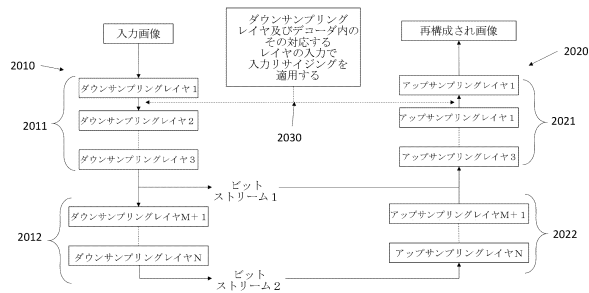
Fig. 17



20

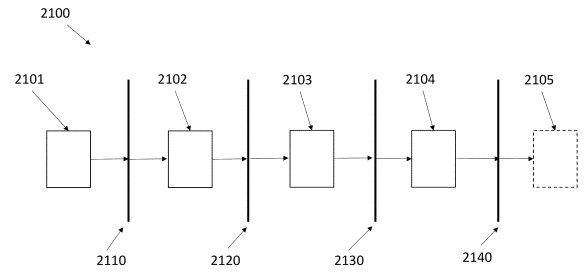
【図 18】

Fig. 18



【図 19】

Fig. 19



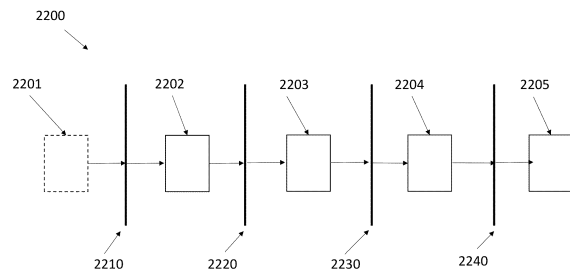
30

40

50

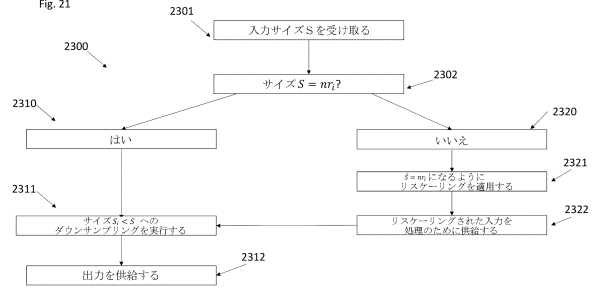
【図 20】

Fig. 20



【図 21】

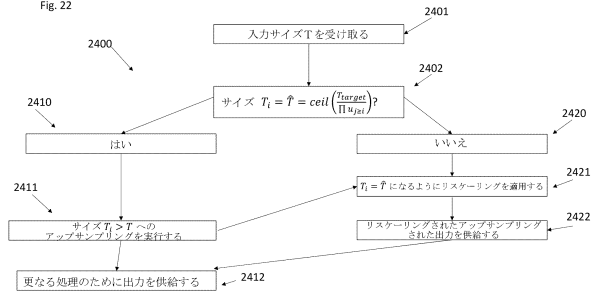
Fig. 21



10

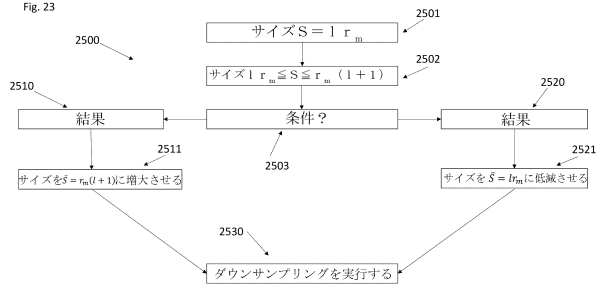
【図 22】

Fig. 22



【図 23】

Fig. 23



20

30

40

50

フロントページの続き

- (74)代理人 100070150
弁理士 伊東 忠彦
- (74)代理人 100135079
弁理士 宮崎 修
- (72)発明者 アルシナ, エレナ アレキサンドロヴナ
ドイツ 80992 ミュンヘン リースシュトラッセ 25 ホアウェイ・テクノロジーズ・デュッセルドルフ・ゲーエムベーハー内
- (72)発明者 ガオ, ハン
ドイツ 80992 ミュンヘン リースシュトラッセ 25 ホアウェイ・テクノロジーズ・デュッセルドルフ・ゲーエムベーハー内
- (72)発明者 エセンリク, セミ
ドイツ 80992 ミュンヘン リースシュトラッセ 25 ホアウェイ・テクノロジーズ・デュッセルドルフ・ゲーエムベーハー内
- 審査官 田中 純一
- (56)参考文献 米国特許出願公開第2020/0162751 (US, A1)
国際公開第2020/059581 (WO, A1)
米国特許出願公開第2021/0344967 (US, A1)
特表2019-531036 (JP, A)
米国特許出願公開第2019/0230349 (US, A1)
- (58)調査した分野 (Int.Cl., DB名)
H04N 7/12
H04N 19/00 - 19/98
G06F 18/00 - 18/40
G06N 3/00 - 99/00
IEEE Xplore