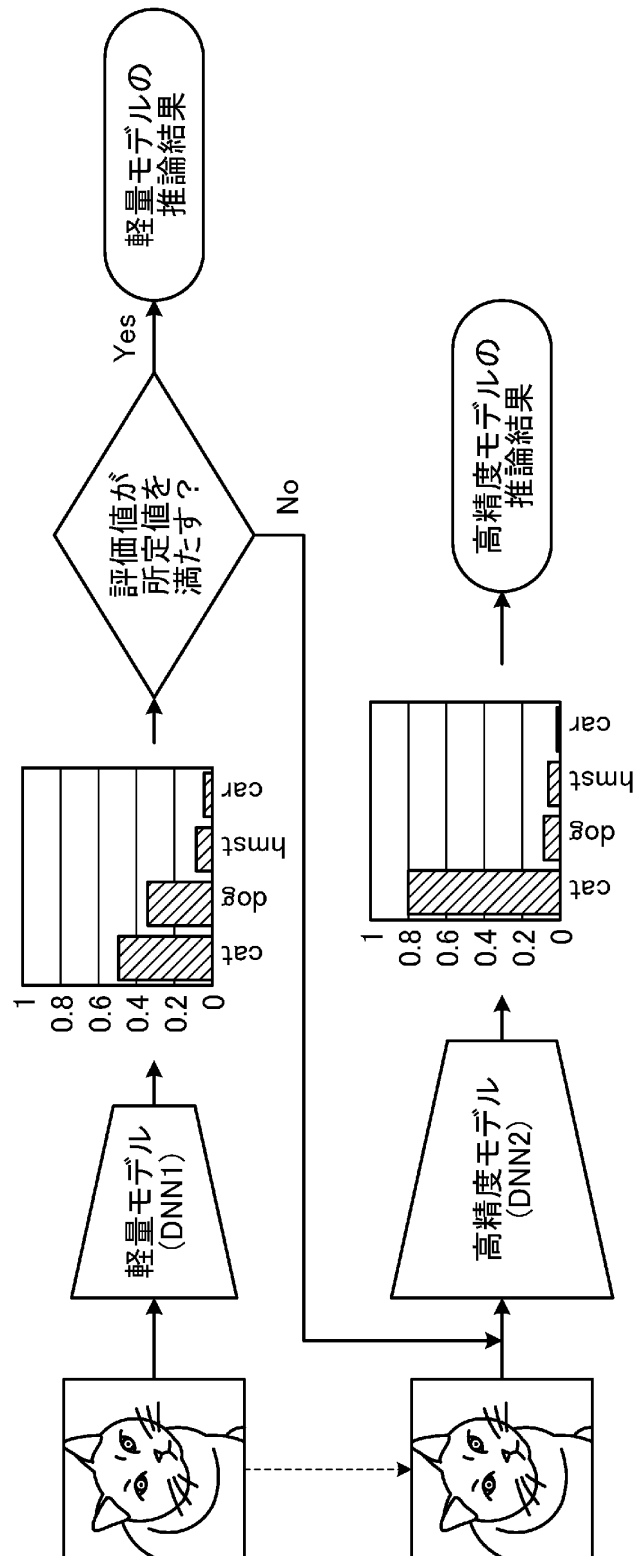
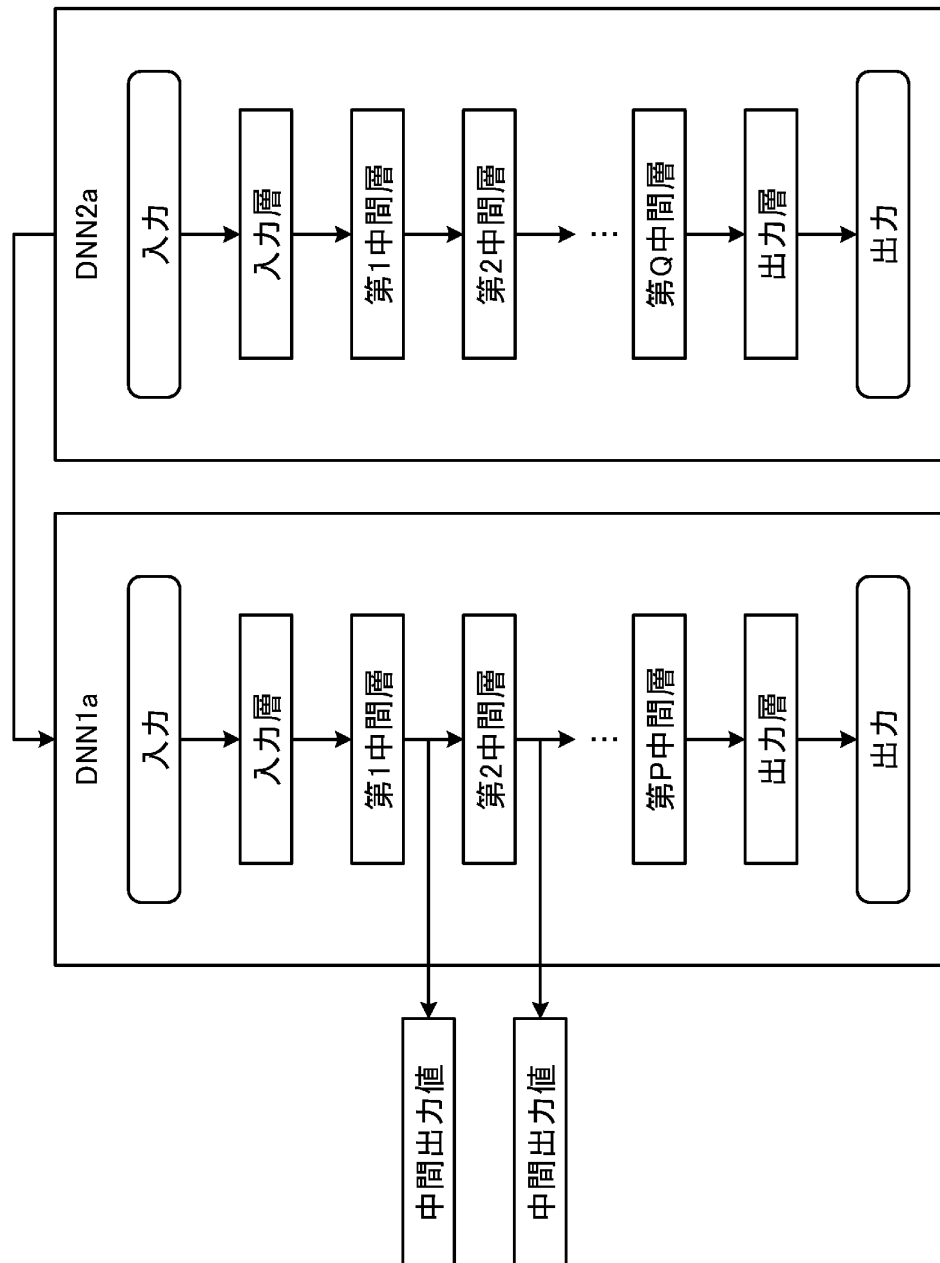


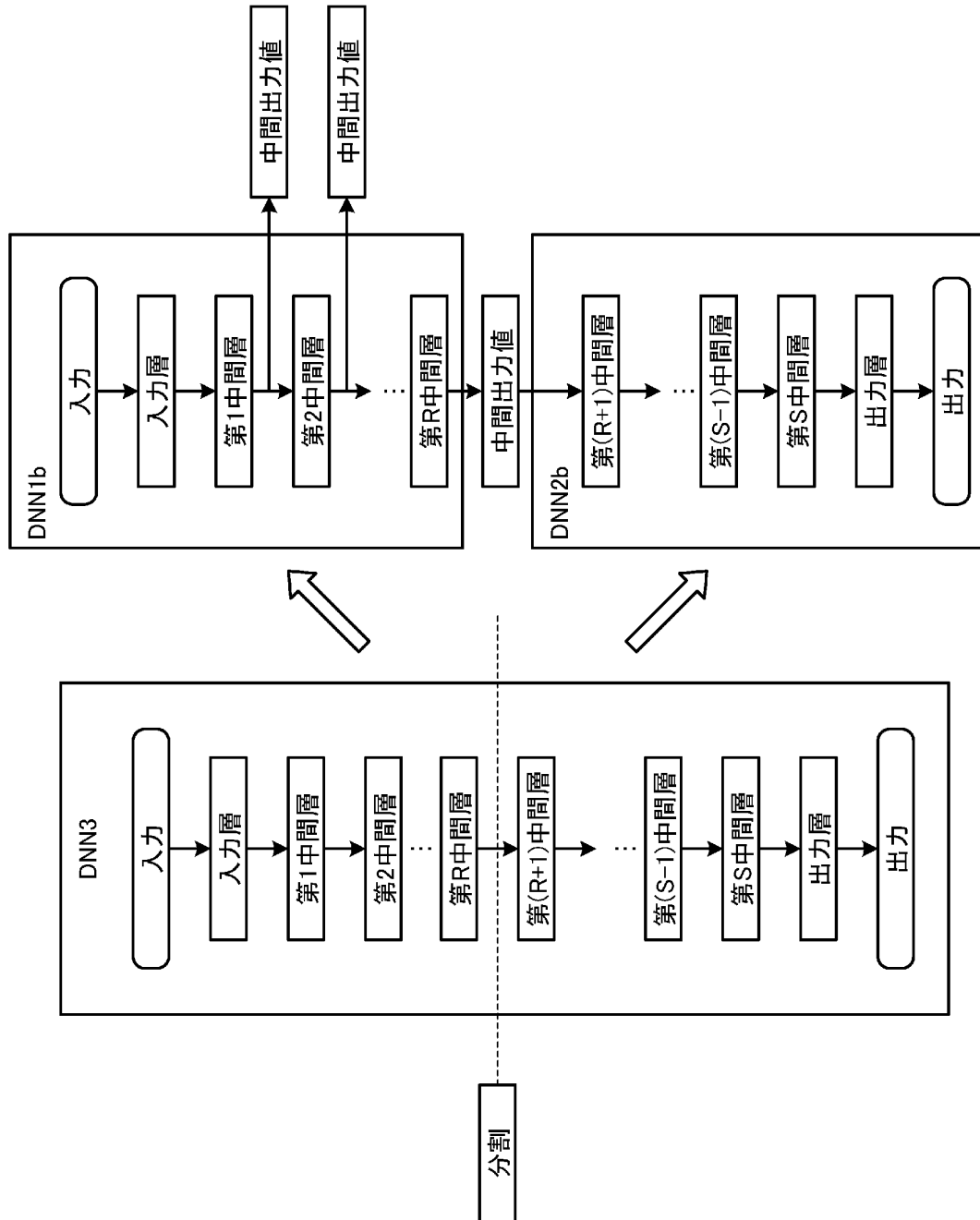
[図1]



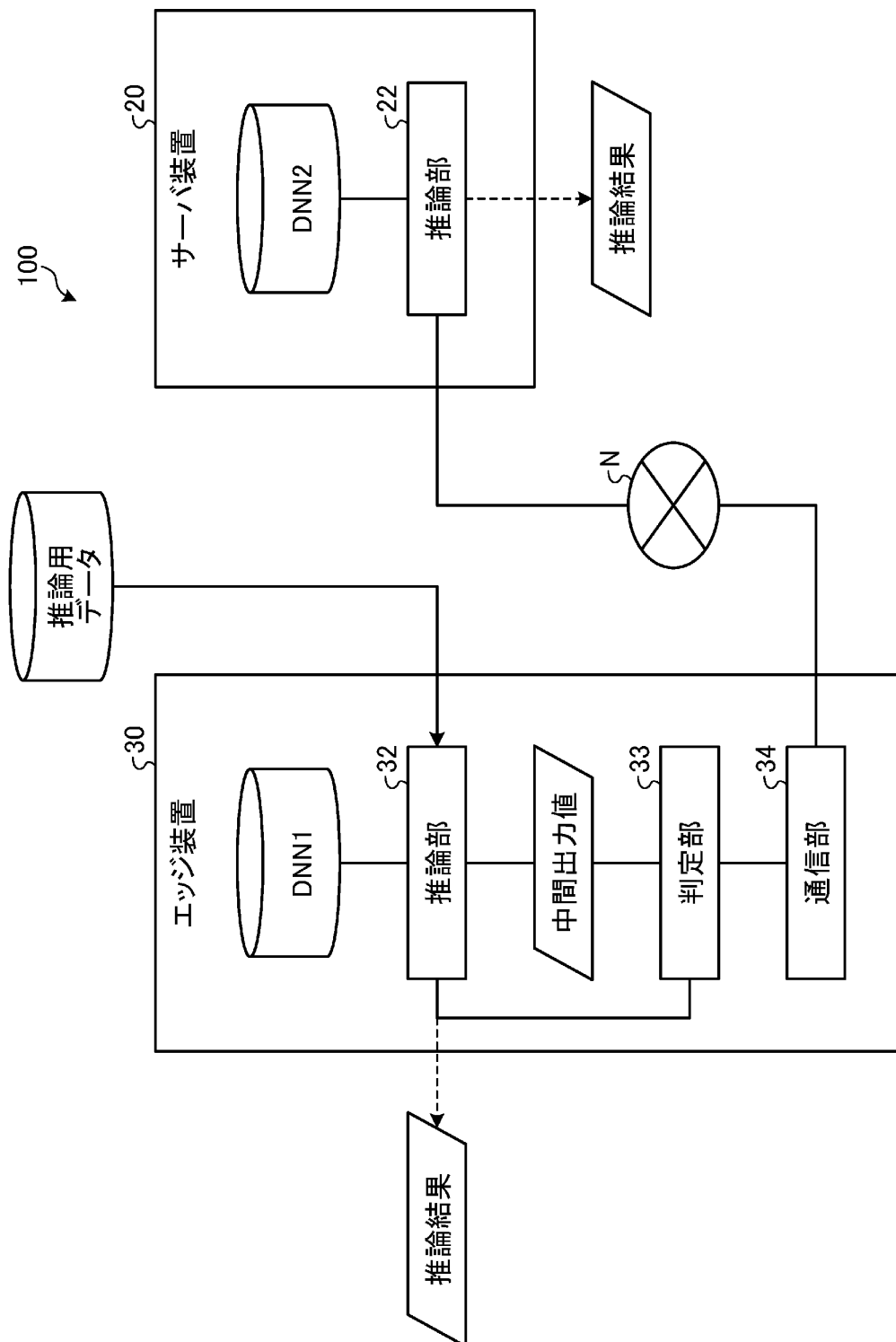
[図2]



[図3]



[図4]

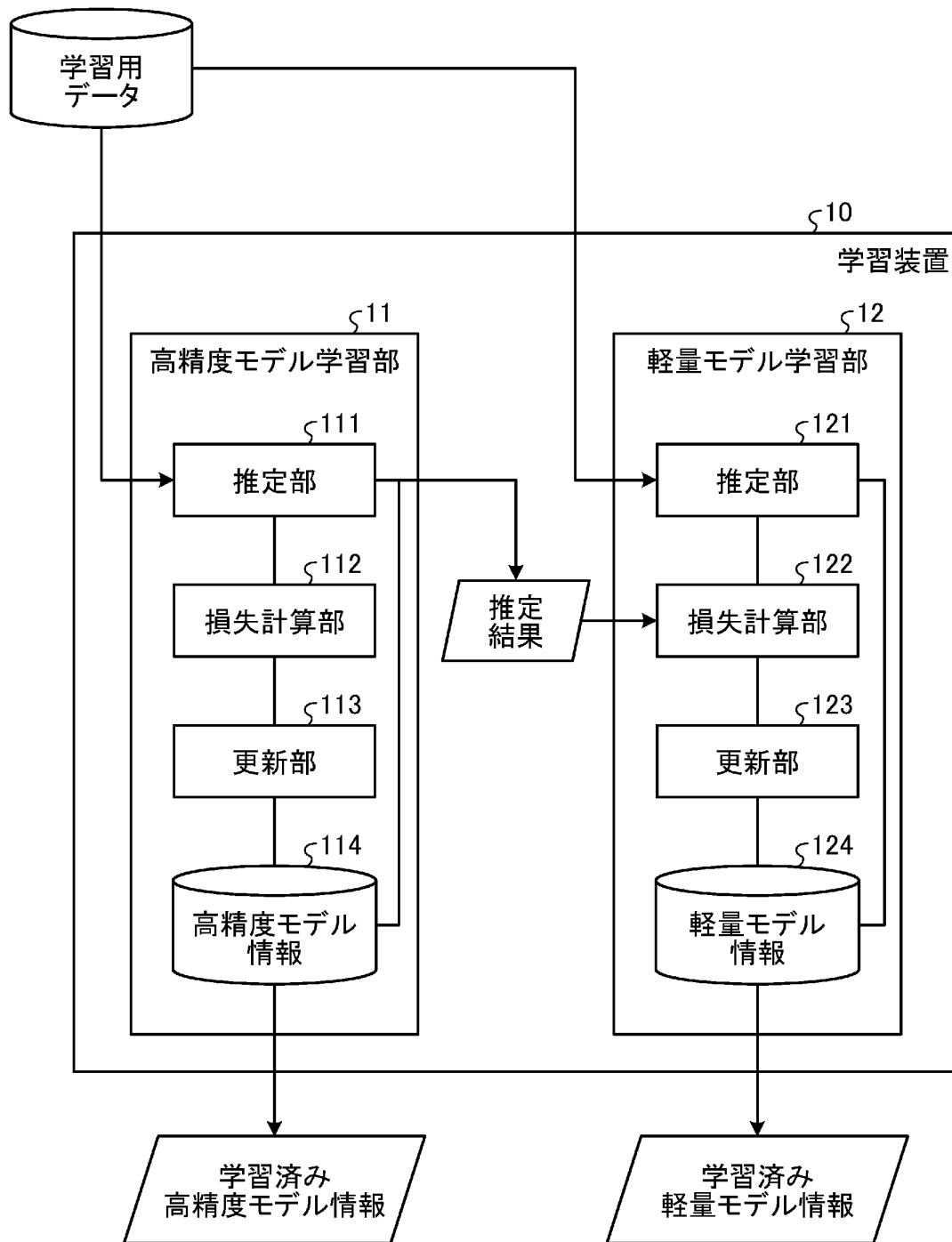


```

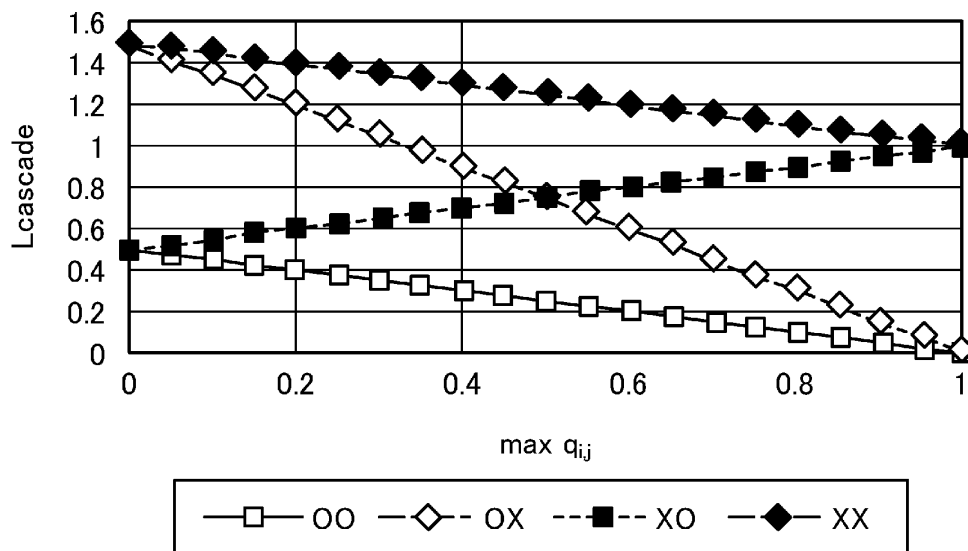
graph TD
    subgraph Edge_Device_30 [エッジ装置30]
        direction TB
        subgraph Recommendation_Unit_32 [推薦部 32]
            direction TB
            S1((推薦用データ S1)) --> S2((推薦用データ S2))
            S2 --> DNN1_32((DNN1))
            DNN1_32 --> S3((中間出力値 S3))
            S3 --> S4[中間出力値取得 S4]
            S4 --> S5[評価値取得 S5]
            S5 --> S6{評価値が  
所定値を満たす? S6}
            S6 -- Yes --> S8((推薦結果 S8))
            S8 --> S9[推薦結果を出力 S9]
            S6 -- No --> S7((中間出力値 S7))
            S7 --> DNN1_32
            DNN1_32 --> S10((推薦結果 S10))
            S10 --> S11[推薦結果を出力 S11]
        end
        subgraph Judgment_Unit_33 [判定部 33]
            direction TB
            S3 --> S4
            S4 --> S5
            S5 --> S6
        end
        subgraph Communication_Unit_34 [通信部 34]
            direction TB
            S10 --> S11
            S11 --> S12((推薦結果 S12))
            S12 --> S13[推薦結果を取得 S13]
            S13 --> S14[推薦結果 S14]
            S14 --> S15[推薦結果 S15]
            S15 --> S16[推薦結果を出力 S16]
        end
    end
    S1 --> S2
    S2 --> DNN1_32
    DNN1_32 --> S3
    S3 --> S4
    S4 --> S5
    S5 --> S6
    S6 -- Yes --> S8
    S8 --> S9
    S6 -- No --> S7
    S7 --> DNN1_32
    DNN1_32 --> S10
    S10 --> S11
    S11 --> S12
    S12 --> S13
    S13 --> S14
    S14 --> S15
    S15 --> S16

```

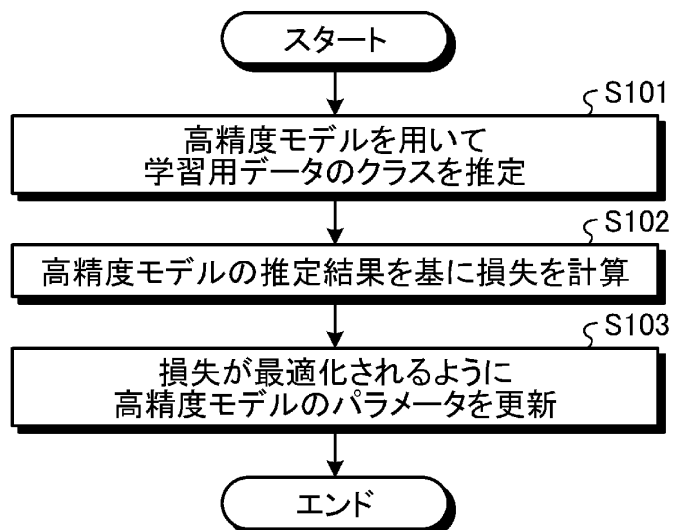
[図6]



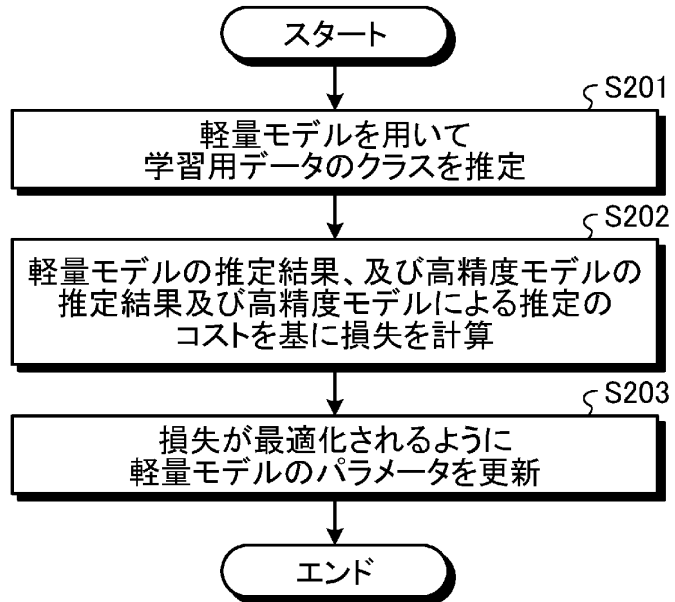
[図7]



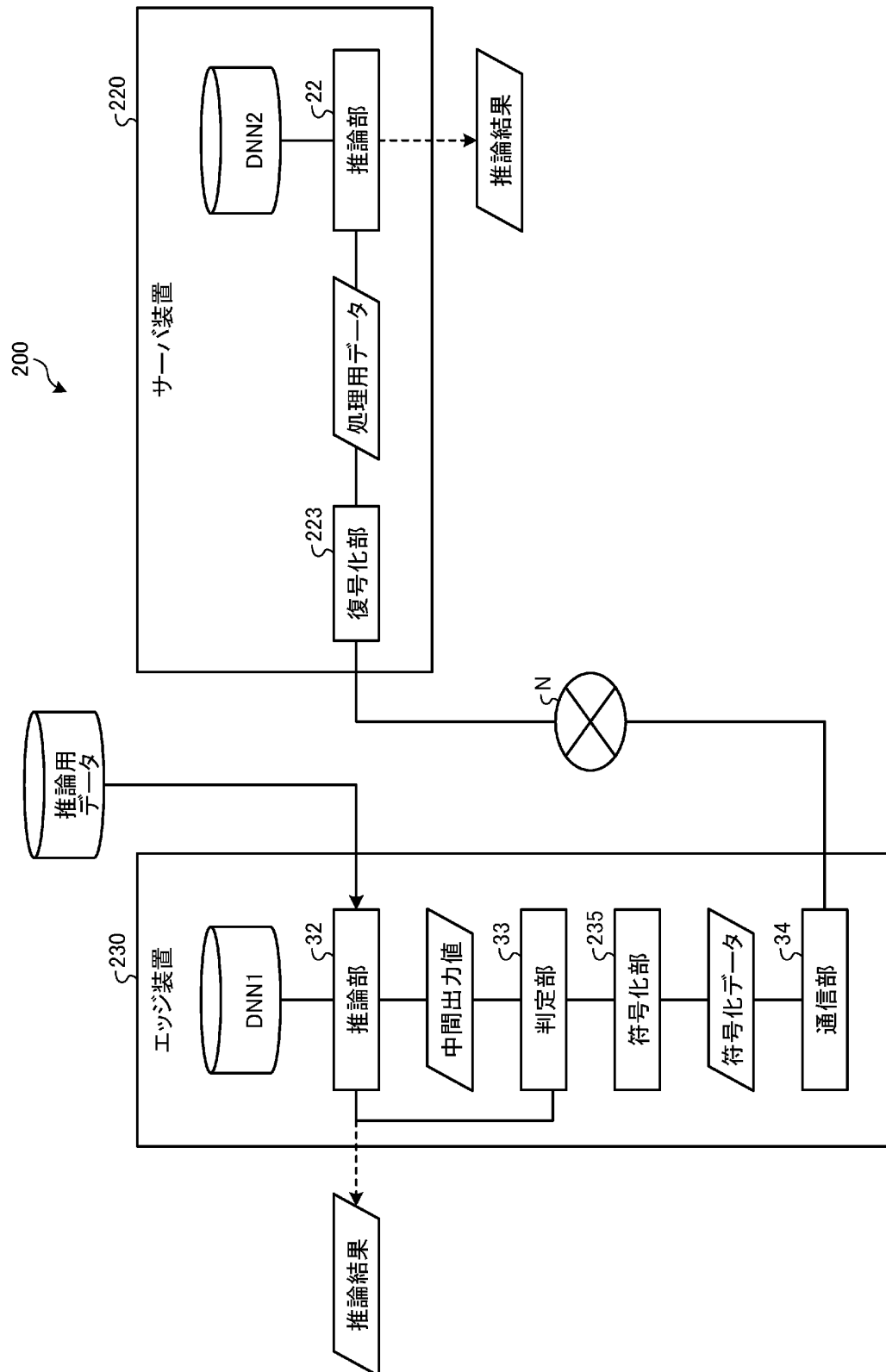
[図8]



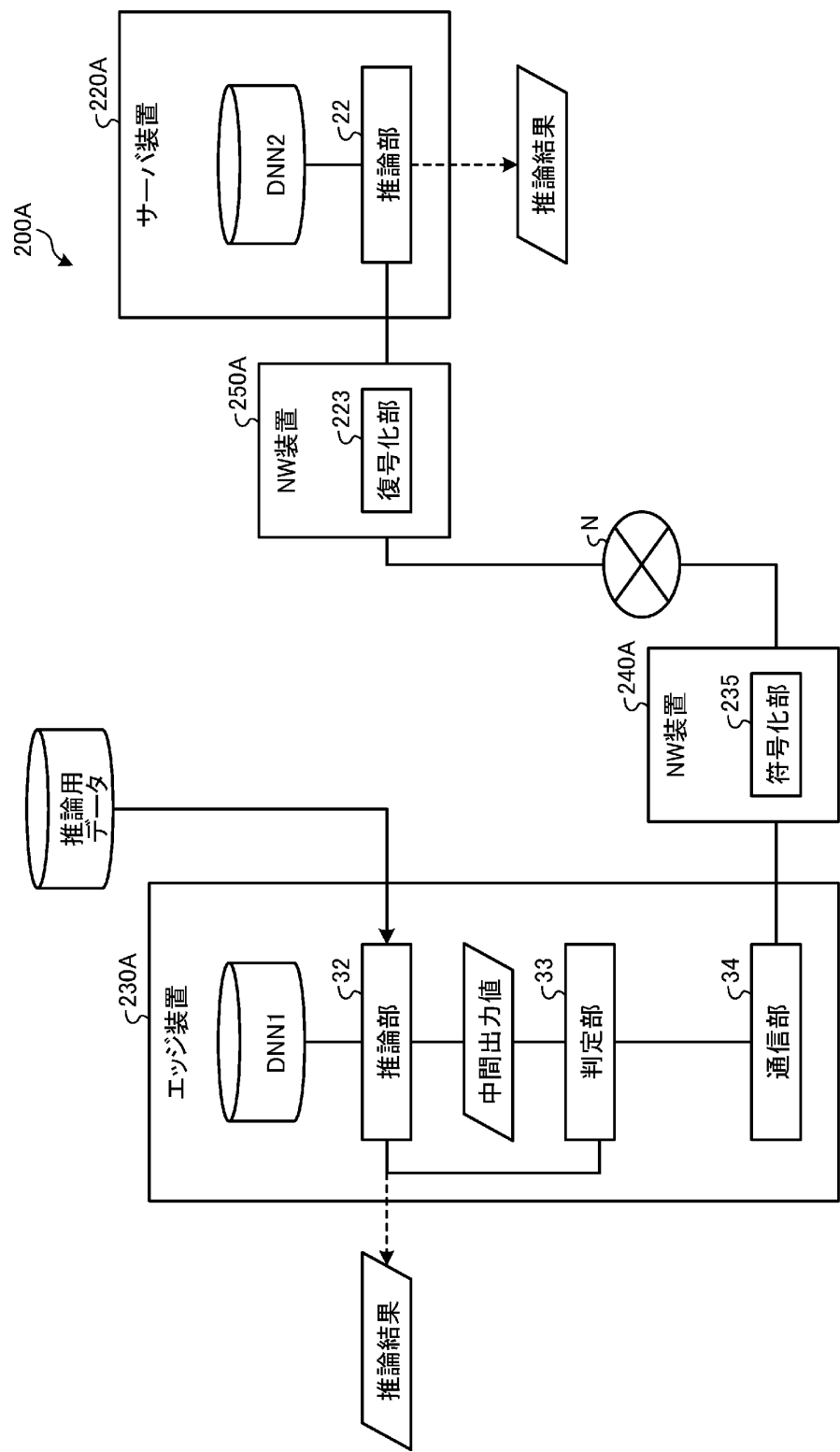
[図9]



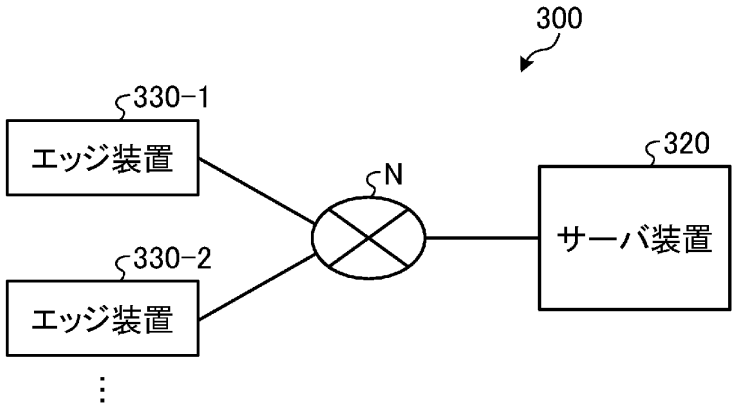
[図10]



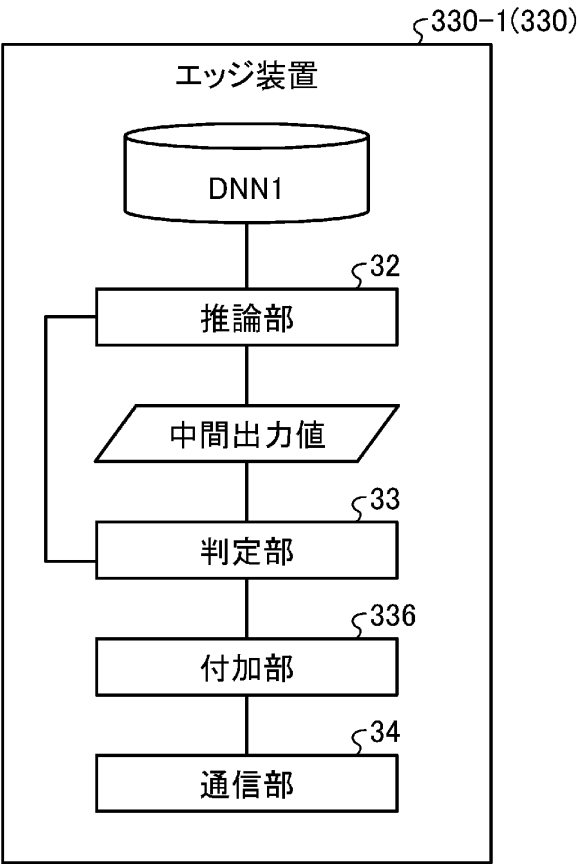
[図12]



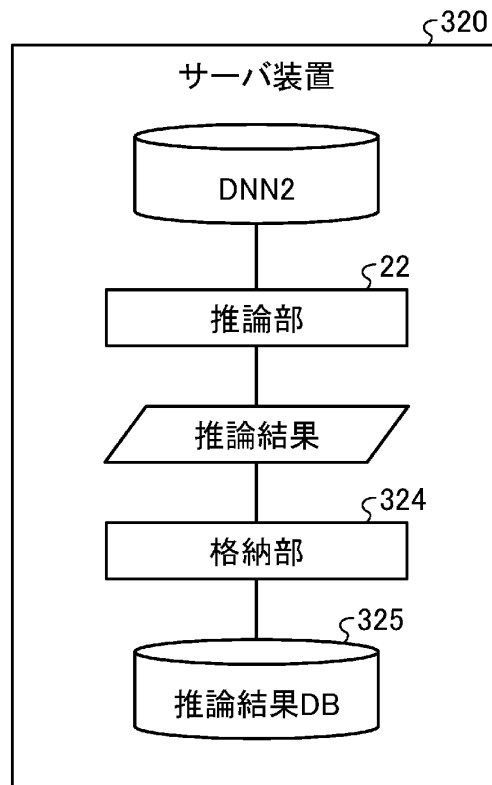
[図13]



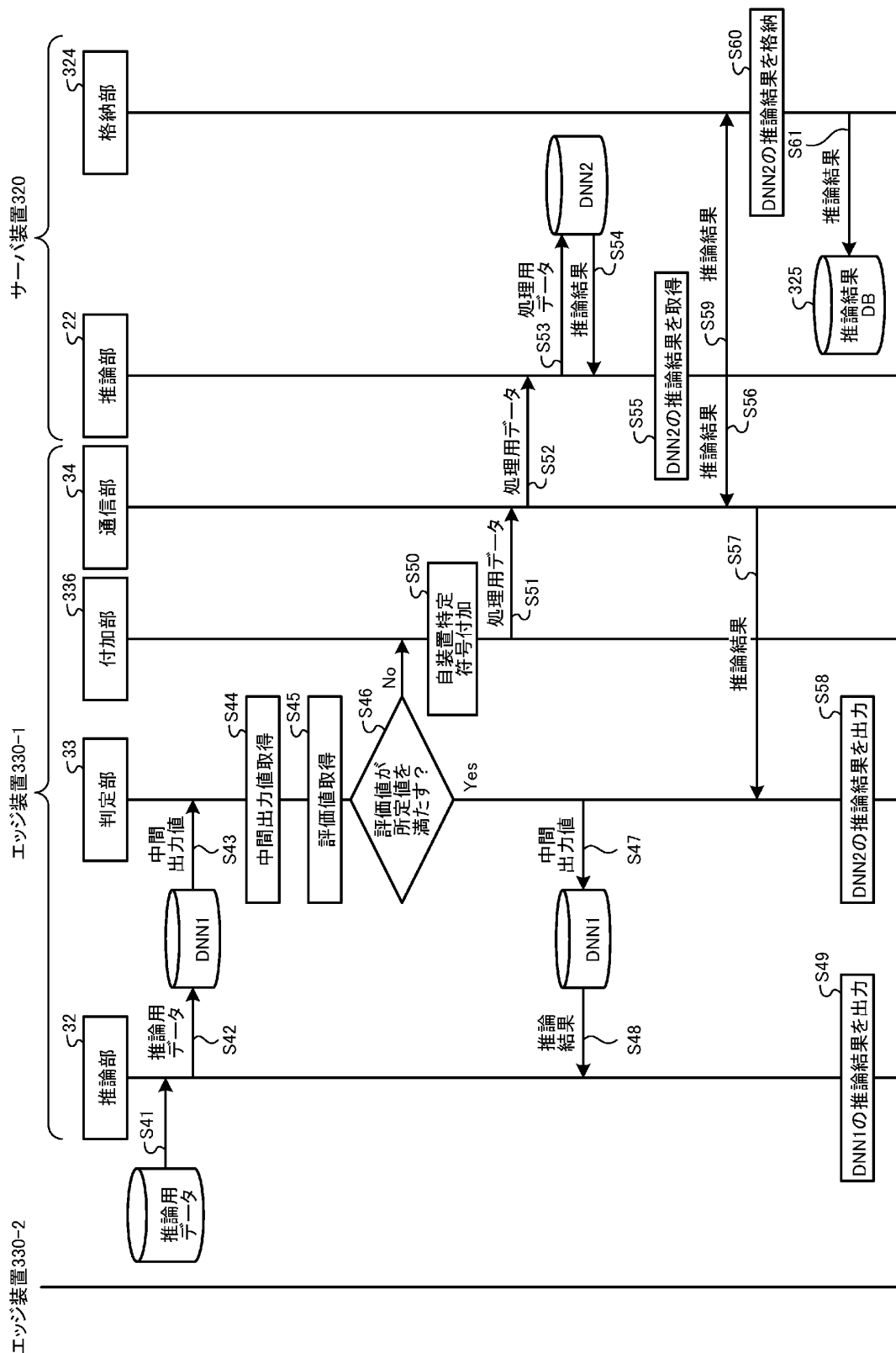
[図14]



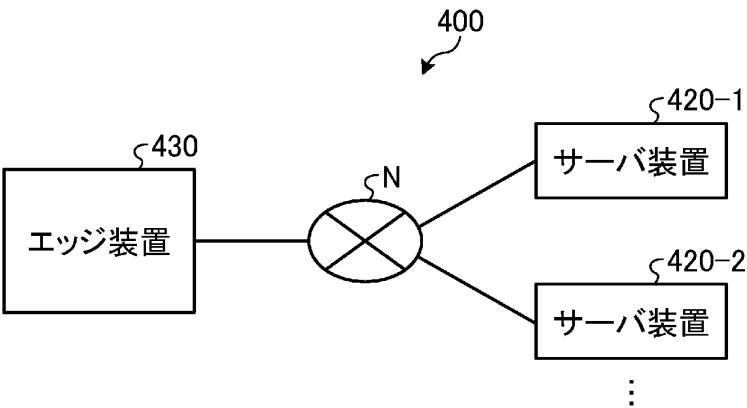
[図15]



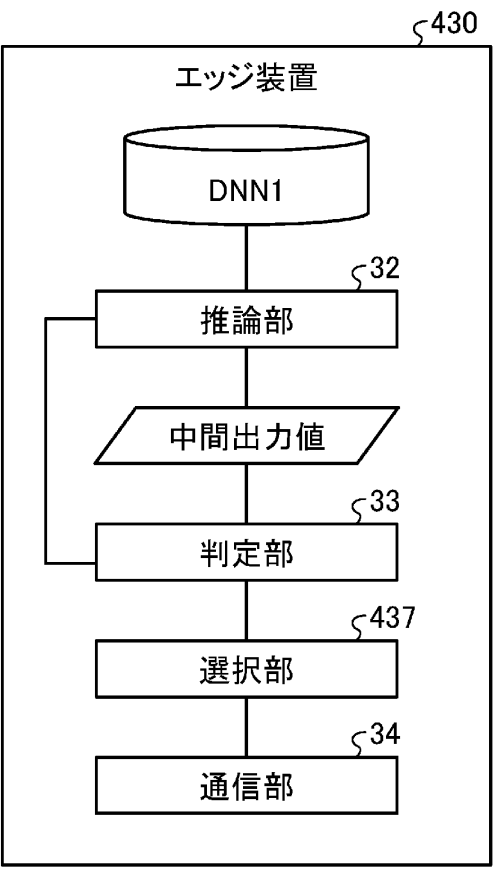
[図16]



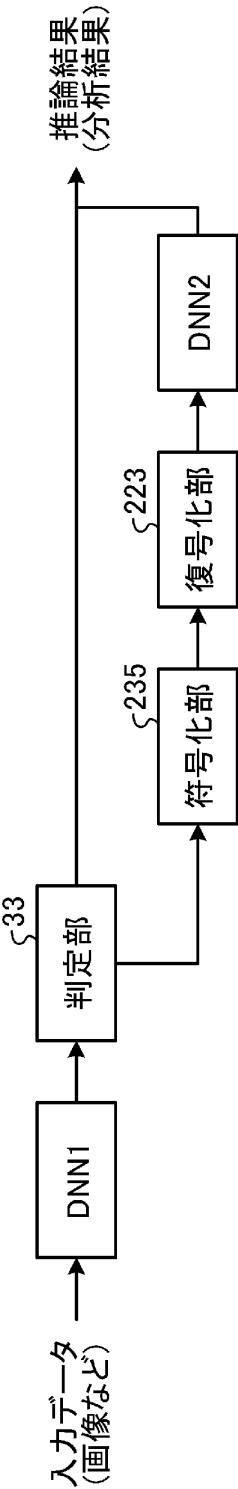
[図17]



[図18]



[図20]



[図21]

DNN1、DNN2	判定部	符号化部、復号化部、通信内容
(1-A) DNN1の負荷が低ければ可能な限りDNN1で最後まで推論をする (HWリソースが足りない時はDNN2に送り、残りをDNN2で実行する)	(2-A) DNN1の実行と並行してDNN2にデータを送り、負荷に余裕があればDNN2でも並行処理する	(3-A) 元画像または、元画像に加えてDNN1の処理結果をそのまま送信
(1-B) 一つのDNNをHWリソースや処理量に応じた特定の層で分割(DNN1とDNN2)しDNN1では途中で処理し、DNN2で残りを実行する 実施例: 負荷ピーク時5層、平時12層等負荷に応じて分割点を随時変更	(2-B) DNN1の実行と並行してDNN2へデータを送り、DNN2は優先度の低いリクエストとして優先制御する	(3-B) DNN1の結果に応じて、必要な情報だけをDNN2に送信する
(1-C) 一つのDNNをエントロピーが少ない層で分割(DNN1とDNN2)しDNN1では途中で処理し、DNN2で残りを実行する	(2-C) DNN1全体または中間出口の処理結果でDNN2処理の要否を判断 実施例: 処理結果として”信頼度”を用いる	(3-C) 一つのDNNを特定の層で分割(DNN1とDNN2)し、分割層の特徵量だけを送る
(1-D) DNN1では低負荷な推論のみを実施し、高負荷・高精度なDNN1はDNN2で行う	(2-D) DNN1の途中状態をDNN2に送る	(3-D-1) 分割層に対して量子化してDNN2に送り、DNN2が受け取った後に元の精度に戻して処理を継続する
(1-E) DNN1の結果により次に必要なDNNを決定し、選択されたDNN2を実行する	(2-E) DNN1の結果で送り先のDNN2を選択する 実施例: DNN1で物体検知、人検知: DNN2で男女判別、車検知: DNN2でナンバー検知	(3-D-2) 分割層に対して量子化してDNN2に送り、元の精度に戻さず、処理を継続する
(1-F) 要求されるレイテンシにより、高レスポンスが必要な場合DNN1で実施、そうでない場合他のエッジとbatchしてDNN2で処理する 実施例: エッジデバイス+serving	(2-F) DNN2までの空間的距離や稼働/システム状態(DNN2の空き具合、待ちタスクの量等)により、利用するDNN2を選定 実施例: Serving	(3-D-3) 分割層に対して量子化してDNN2に送り、DNN2の途中のある層で精度を戻して処理を継続する
(1-G) DNN1、DNN2に様々なタスク難易度に対応可能な中間出口を挿入	(2-G) 局所性の高い(機密・プライバシー等)情報はDNN1で処理、DNN2はそれ以外の加工済みの情報を引き継ぐ	(3-E) DNN2へ引き継ぐ特徴マップの各構成要素(ノード)の出力傾向(DNN1の学習中に取得)に最適化した符号化手段を設計する
(1-H) DNN1は通常の学習に加えて適応的選択を高度に行えるような学習を行う		(3-F) DNN2へ引き継ぐ特徴マップ全体を1空間とし、各マップをこの空間内でクラスタ化した結果を用いて符号化手段を設計する

[図22]

