特別研究報告

題目

デジタルツイン構築のための脳の認知機構を用いたオブジェクト認識手法の実装および評価

指導教員
村田 正幸 教授

報告者
久保 快斗

令和 4 年 2 月 8 日

大阪大学 基礎工学部 情報科学科

令和 3 年度 特別研究報告

デジタルツイン構築のための脳の認知機構を用いたオブジェクト認識手法の実装および評価

久保 快斗

内容梗概

近年，ロボットの自動制御や車の自動運転といった機械が人間に操作されずに自律的に動作する技術が注目を浴び研究が進められている．機械の自律的な動作には情報を取得・処理し，判断行動をとる段階が存在する．情報の取得・処理として実世界を仮想世界上に表現すること，また，判断行動として構築した仮想世界の情報に基づいて実世界上での行動を決定することが必要である．すなわち実世界上の人や物体などのオブジェクトをセンシングしてリアルタイムに 3 次元のデジタルデータとして仮想世界上に表現するデジタルツインの構築が望まれている．デジタルツインの実現における課題としてリアルタイムでのオブジェクト認識，高精度なオブジェクト認識があげられる．しかし現実的には 100% の精度でオブジェクトを認識することは困難である．そこで我々の研究グループでは，誤差を含んだ形で仮想世界上の存在を表現する，確率的デジタルツインの実現を目指している．

誤差の含まれる不確実な観測情報に基づき意思決定を行うシステムとして人の脳がある．人は情報を処理するとき，観測した情報に対して，自身の知識や過去の経験などからそれが何であるかを判断する．近年このような人の脳の情報処理機構を数理モデル化する研究が進められている．その一つに，観測した情報が記憶している選択肢の中のいずれに該当するのかを判断するという意思決定を，ベイズ推定に基づきモデル化した Bayesian Attractor Model (BAM) がある．我々は，人の脳が行う情報処理に倣うことで，確率的なデジタルツインの構築を目指しており，これまでに，BAM に基づくオブジェクト認識技術を開発してきた．本報告では，BAM によるオブジェクト認識を行うために与える特徴量を，動画像から BAM の処理に適した形式で抽出する手法を提案する．動画像から Siamese RPN を用いてオブジェクトを囲むバウンディングボックス位置の推定を行い，バウンディングボックスで囲まれた画像から特徴量を抽出する．この特徴量 BAM に入力してオブジェクト認識を行う一連の手法を実装し，評価を行う．公開データセットを用いて認識精度と速度を評価した結果，BAM の最終的な認識精度は，バウンディングボックス位置の推定精度および特徴量の抽出方法に大きく依存することが確認できた．バウンディングボックス位置の推定精度指標である Intersection over Union (IoU) が 0.444 であった際に，BAM によるオブジェクト認識精度は 54.36%となることを示した．

主な用語

デジタルツイン
ゆらぎ学習
ベイズ推定
機械学習
Siamese Network
AlexNet
Siamese RPN

目 次

図 目 次

表 目 次

1　はじめに

近年ロボットの自動化や車の自動運転技術が望まれ，研究が進められている [1]．現在のロボットは pepper や aibo といったコミュニケーションを重視した家庭用製品が主流であり，自動運転技術も高速道路での車線維持，渋滞時の前車の追従といったものにとどまっている．将来のロボットは建設現場での自動敷設に使われるといった需要が現れることが考えられ，自動運転も最終的には乗った場所から目的地まで，交差点の右左折など含めてすべて自動で行う技術が現れると考えられる．このとき，一つの行動選択の誤りが重大な事故を引き起こすことが想像できる．自動制御の正確性は極めて重要であり，適切な制御が行われるためには正確かつ素早い情報取得，処理，行動判断が不可欠である．このような自動制御の実現方法として，実世界を仮想世界上に実現し，この仮想世界の情報に基づいて実世界上の制御を行う技術が検討されている．すなわち実世界上の人や物体などのオブジェクトをセンシングし，リアルタイムに 3 次元のデジタルデータとして仮想世界上に実現する，デジタルツインと呼ばれる技術である [2].

デジタルツイン構築の課題として，リアルタイムかつ正確なオブジェクト認識があげられる．カメラなどで実世界をセンシングする際，カメラのピントボケ，ブレや転送時のノイズなどによってセンシングデータをそのまま観測結果として用いる際には誤差が含まれることとなる．またセンシングデータの処理は IoT (Internet of Things) 機器などの計算処理能力が乏しいコンピュータで行われることが見込まれ，計算処理にはある程度の時間がかかり，精度も低い計算能力に由来して低精度であると考えられる．以上からセンシングしたデータからオブジェクトを確実に抽出し，誤ることなくリアルタイムで認識することは非常に困難である．我々の研究グループでは，ノイズが含まれる不確実な入力から，認識誤差を許容し誤差を含んだ形でオブジェクトを推定し，推定結果を仮想世界上に確率を含んだ形式で表現する確率的デジタルツインの実現を目指している．

オブジェクトの認識方法について，不確実な入力から意思決定を行うよく知られたものとして人間の脳がある．人の脳の行う情報処理では，聴覚や視覚といった感覚器から情報を得たのち，その情報と自身の知識や経験を照合し，オブジェクトが何であるのかを判定する．BAM (Bayesian Attractor Model) [3] は，このような意思決定をベイズ推定に基づいて数理モデル化したものである．我々の先行研究では，この BAM を用いることで，人の脳が行うユニモーダル情報処理を模したオブジェクト認識手法を提案している [4]．文献 [4] は，動画像におけるオブジェクト認識を BAM によって行う際の，BAM 自体の認識精度，計算速度を評価したものであり，BAM に与える入力には理想的な状況を想定して取得した連続データを用いている．すなわち，動画像中の認識したいオブジェクトを囲むバウンディングボックスが正確に推定できていることを仮定し，その上でバウンディングボックスに囲まれ

た画像に対する特徴量を抽出している．また，特徴量抽出は 4 層 CNN を用いて 128 次元の値として抽出している．

本報告では，BAM に与える入力特徴量を取り出すために，上記の特徴量抽出方法を拡張する．オブジェクトの存在する領域推定には Siamese RPN [5] を採用し，同時に，Siamese RPN において用いている Siamese Network によって特徴量を抽出する．以上のオブジェクト検出，特徴量抽出を含めた形で BAM によるオブジェクト認識手法の実装を行い認識精度を評価する．文献 [4] では 4 層の CNN を用いて比較的小さな次元の特徴量を抽出していたが，これは前段のオブジェクトの検出が確実にできていることを前提としていたものであり，オブジェクトの検出および特徴量抽出，BAM によるオブジェクト認識までの一連の処理の実行に要する計算時間についても評価を行う．

2　関連研究

文献 [4] におけるオブジェクト認識手法に関して，映像モーダル処理の一連の流れを図 1 に示す．

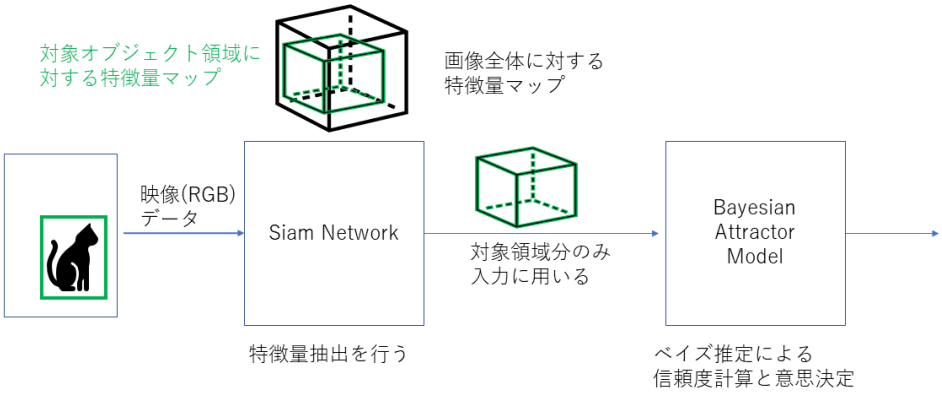


図 1: 映像モーダル処理（文献 [4]）

文献 [4] では BAM に入力する特徴量次元を小さくするために，Siamese Network に 4 層の軽量な CNN を用いている．Siamese Network で取り出した特徴量マップは画像全体に対する特徴を取り出したものであるため，入力した情報が記憶した情報にマッチするのかを識別する BAM には適さない（BAM の詳細は 2.2 節で説明する）．そこで，オブジェクトを囲んだバウンディングボックス領域に対応する特徴量マップを，画像全体に対する特徴量マップから取り出して，BAM の入力としている．

文献 [4] ではバウンディングボックスの推定位置として正しい位置が与えられた上で特徴量抽出を行っている．さらに BAM では認識を行うオブジェクトに関する特徴量がアトラクターに紐づけて記憶されるが，文献 [4] ではこの記憶する特徴量を動画の最初の 1 フレーム目に写ったオブジェクトについて取り出したものとしている．動画像の連続するフレームについて，探索対象であるオブジェクトを正確に囲むバウンディングボックスの内側の画像に関する特徴量が取り出されて BAM に入力され，入力されている特徴量が記憶したどのオブジェクトの特徴量と近いのかを判断する．

2.1　Siamese Network

Siamese Network [6] は図 2 に示す構成となっている．

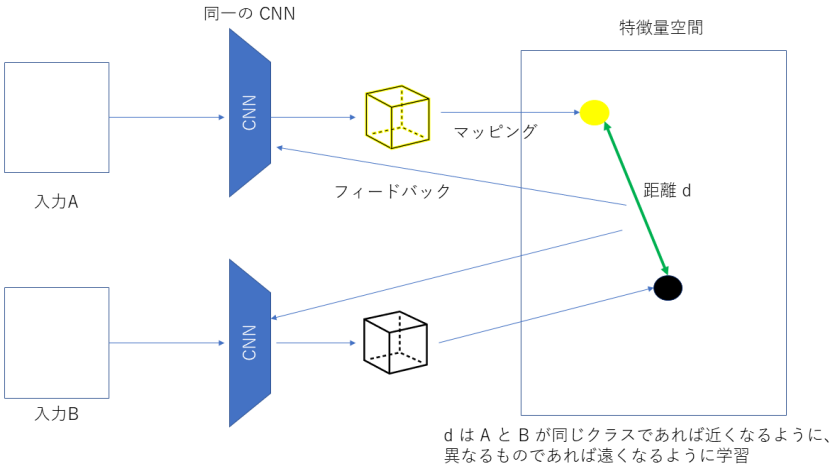


図 2: Siamese Network

Siamese Network はオブジェクトトラッキングの問題を，認識したいクラスの画像（テンプレート画像）から抽出される特徴表現と，探索対象画像（オリジナル画像）から抽出される特徴表現間の，相互相関により得られる汎用的な類似性マップが正解となるように学習することで解決を図る．図 2 の入力 A をテンプレート画像，入力 B をオリジナル画像としたときに A と B を同一の CNN で畳み込み，特徴量マップを取り出す．この特徴量マップの特徴量空間状の位置が，A と B が同一クラスのものであれば近くなるように，違うクラスのものであるならば遠くなるように学習を進める．距離はユークリッド距離で計算される．

2.2 Bayesian Attractor Model (BAM)

文献 [3] では，人が知覚情報に対してどのように意思決定を行っているのかを，ベイズ理論を用いてモデル化している．これを BAM (Bayesian Attractor Model) と呼んでいる．BAM は 四つの主要なステージで構成されている．それぞれのステージがどのような処理を行うかを説明する．

実験刺激の抽象モデル

BAM では，提示される刺激が抽象化され，特徴量空間での変数表現 x_t に変換される．この変換に関して，刺激が同一であれば同じ確率分布に従って値が発生するようなモデル化を行っており，時刻 t における判断選択肢 A_t（すなわち提示される刺激の選択肢）は，$N(\mu_i, s^2 I)$ に従う特徴量ベクトル x_t とマッピングされる．ここで μ_i は選択肢 i が誤差なく提示された場合の特徴量ベクトル，s はノイズの標準偏差，I は単位行列である．

刺激の生成モデル

ある時刻 $(t-\Delta t)$ から次の時刻 (t) となった際，意思決定状態 z_t は次式で更新される．

$$z_t - z_{t-\Delta t} = \Delta t f(z_{t-\Delta t}) + \sqrt{\Delta t} w_t \tag{1}$$

ここで $f(z)$ はアトラクターモデルの 1 種であるホップフィールドダイナミクスであり，勝者総取りのメカニズムである．このダイナミクスは意思決定の選択肢の数に応じた複数のアトラクター $\phi_1 \ldots \phi_n$ を持つように設計されており，それぞれのアトラクター ϕ_i が選択肢 A_i と対応する．w_t はノイズ変数で $N(0, \frac{q^2}{\Delta t} I)$ に従う．q はダイナミクスの不確実性を表し，これが大きいほど z_t がアトラクターの間で切り替わる可能性が高くなる，すなわち，選択肢の間で意思決定状態が切り替わる可能性が高くなる．また，式 (2) により観測値 x の確率分布を予測する．

$$x_t = M\sigma(z) + v_t \tag{2}$$

ここで $M = [\mu_1, \ldots \mu_n]$ は各選択肢（$i = 1 \ldots n$）がノイズなしで提示されたときの特徴量ベクトルを並べた行列であり，$\sigma(z)$ は z の各要素を $0 \sim 1$ に変換するシグモイド関数である．v_t はノイズ変数で $N(0, r^2 I)$ に従う．r は観測の不確実性を表し，r が大きいほどノイズが大きくなる．ここで，$z_t = \phi_i$ であるときには，$M\sigma(z)$ の i 番目の要素は $\mu_i + v_i$ であり，それ以外の要素は $N(0, r^2)$ に従う乱数値である．

ベイズ推定

式 1 と 2 で表される生成モデルを逆方向に推定することで時刻 t における観測値 x_t から意思決定状態 z_t の事後確率分布 $P(z_t|x_t)$ が得られる．生成モデルで z_t が非線形なホップフィールドダイナミクスに従うことを仮定しているため，事後確率を求める推定ではそのことを考慮に入れる必要がある．BAM では UKF (Unscented Kalman Filter) を用いて近似計算を行っている．

UKF

UKF は予測された分布と実際の観測値を比較して，予測値の不確実性を考慮しながら，観測値との不一致に比例して決定状態の推定値 $\bar{z}_t$ を更新する，カルマンフィルタの一種である．実際には予測平均値 $\hat{x}_t$ と実際の観測値 x_t の予測誤差を計算し，カルマンゲイン K_t を介して決定状態の予測値 $\hat{z}_t$ を繰り返し更新する．

$$\bar{z}_t = \hat{z}_t + K_t(x_t - \hat{x}_t) \tag{3}$$

カルマンゲイン K_t は予測された観測値の推定共分散行列 $\hat{\Sigma}_t$ と，予測された観測値 $\hat{x}_t$ と予測された意思決定状態 $\hat{z}_t$ の相互共分散 $\hat{C}_t$ から計算される．$\hat{\Sigma}_t$ は観測の不確実性 r の影響を強く受け，$\hat{C}_t$ は q の影響を強く受ける．

$$K_t = \hat{C}_t {\hat{\Sigma}_t}^{-1} \tag{4}$$

カルマンゲイン K_t は意思決定状態の平均値 $\bar{z}_t$ の更新以外にも状態変数 $z_{i,t}$ の事後共分散 $\bar{P}_t$ の推定にも使用される．これは，確信度の計算に用いられる．

意思決定

本モデルでは時刻 t において以下の条件を満たすときに意思決定状態が i であると決定する．

$$p(z_t = \phi_i | X_{\Delta t:t}) \geq \lambda \tag{5}$$

ただし $X_{\Delta t:t} = x_{\Delta t} \dots x_t$ はその時点までに行われたすべての観測値である．$p(z_t = \phi_i | X_{\Delta t:t})$ は i に対応するアトラクター ϕ_i で評価された事後確率密度であり，λ は信頼度の基準となるしきい値である．λ の設定により，状態変数がアトラクター ϕ_i によって与えられる値の付近にあることが必要となる．シンプルなアトラクターモデルでは，状態変数の各要素一つ一つについてしきい値が設定される点が異なっている．ダイナミクスの不確実性 q が大きいほど事後分布は広くなり，証拠の蓄積が速くなり，確率密度の値は小さくなる．言い換えると，q が大きいほど，アトラクターに落ち着く（意思決定）までは速いが，確信度は小さくなる．

3　脳の認知機構を用いたオブジェクト認識手法の実装

3.1　想定サービス

物販会社の倉庫における，運搬ロボットの自動運転を想定している．カメラを搭載し倉庫に積まれている物品の中から取り出したいものを見つけ，ある場所まで自動で運ぶサービスを考える．倉庫内には人やほかのロボットなどいろいろな障害物が存在することが考えられるため，これらを避けながら当該物品の場所まで行く必要がある．物品の置いてある場所までの移動は，ロボットや人など常に移動が考えられる障害物の動作を考えなければならない．センシングから認識まである長さ以上の時間がかかると，そのセンシングデータは認識時点での時間における環境と異なり意味をなさない．よってセンシングから認識までリアルタイムで行うことが必要となってくる．

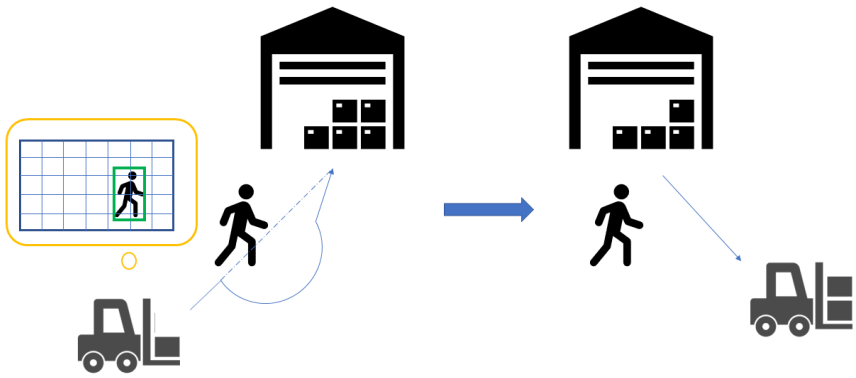


図 3: 想定するサービス

3.2　構成するアーキテクチャ

文献 [4] では，バウンディングボックスが正しく推定できているという理想的な状況下で特徴量を取り出していた．その結果，BAM による認識精度は比較的高い値が得られていた．しかしながら，バウンディングボックスの推定自体に一般に誤りが含まれる．仮に誤った場合には，そのバウンディングボックスに対応する特徴量は BAM によって認識したいオブジェクトの持つ特徴量と異なるものとなってしまう．すなわち特徴量抽出の結果はバウンディングボックスの推定精度に大きく依存する．本報告では，バウンディングボックスの推定とバウンディングボックスに対応する領域の特徴量抽出を同時に行うために，2.1 節の手法を拡

張した Siamese RPN [5,7] を採用する．Siamese RPN 全体の構成アーキテクチャは図 4 になる．

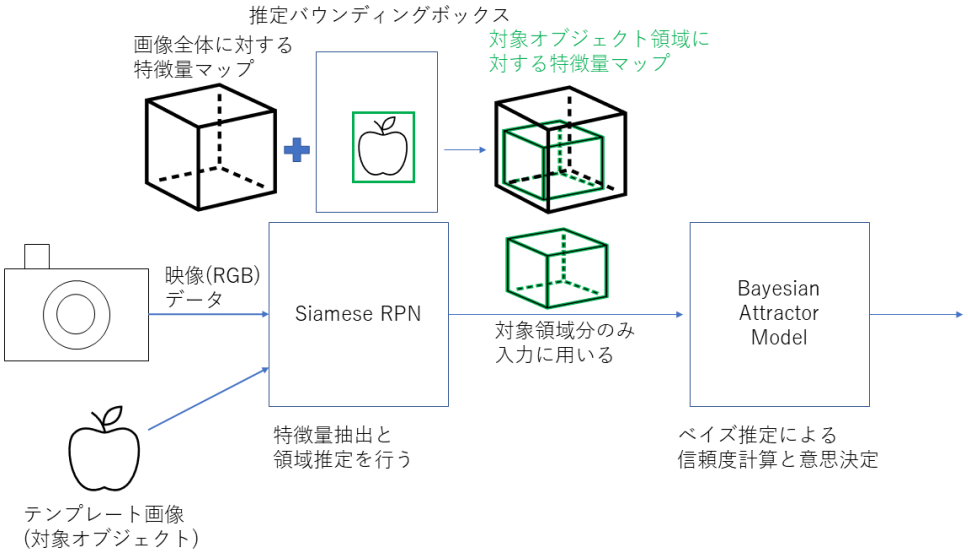


図 4: Siamese RPN を特徴量抽出に用いたユニモーダルオブジェクト認識アーキテクチャ

3.3 Siamese RPN

Siamese RPN のネットワーク構造を図 5 に示す．

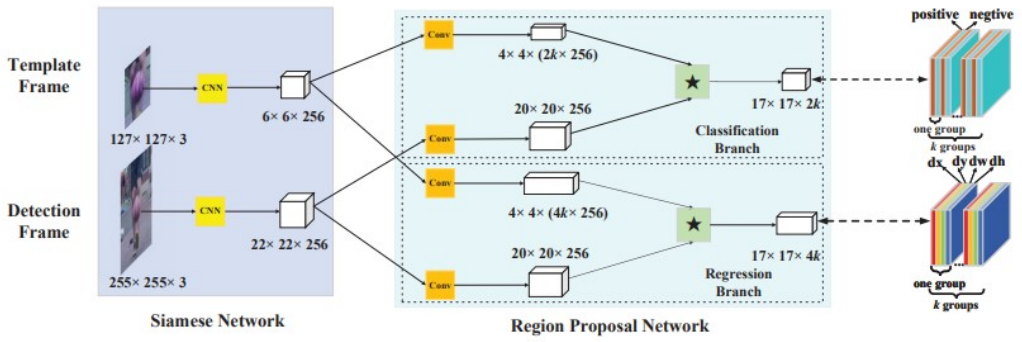


図 5: Siamese RPN のメインフレームワーク（文献 [5] より引用）

Siamese RPN 全体のアーキテクチャは二つのネットワークアーキテクチャにより構成される．一つ目は Siamese Network（2.1 節）である．Siamse Network では探索対象のオブジェクトの画像と，そのオブジェクトが写っているのかを探索する全体画像から，特徴量抽

出器を使って特徴量マップを取り出す．前者をテンプレート画像，後者をオリジナル画像と呼ぶ．二つ目は Region Proposal Network である．ここでは Siamese Network の出力であるテンプレート画像に対する特徴量マップ $\phi(Z)$ と，オリジナル画像に対する特徴量マップ $\phi(X)$ を受け取り，それぞれに対して畳み込み層による処理を行ったのち，それぞれの結果を合わせて畳み込み演算を行う．結果として検出対象オブジェクトが画像のどの位置に写っているのかを推定し，出力する．

3.3.1 Region Proposal Network

Faster R-CNN アーキテクチャ [8] で用いられている Region Proposal Network を活用する．Faster R-CNN は画像内のオブジェクトを探す対象領域の選定においても CNN を用いており，領域選定についての時間が短い特徴を持つ．この領域選定を行うネットワークが Region Proposal Network である．その手順として，まず，領域選定における基準点としてアンカーとアンカーボックスを設定する．設定項目は表 1 のとおりである．

表 1: アンカーボックスの設定

変数名	意味
x	オリジナル画像特徴マップの x-2 次元数
y	オリジナル画像特徴マップの y-2 次元数
RATIOS	縦横比率
SCALES	縦
ANCHOR_NUM	各基準点で考えるアンカーボックス数 (RATIOS と SCALES の組み合わせ数)

特徴量マップにおいて，3×3 のカーネルによって畳み込み演算を行う．このカーネルの各中心点がアンカーとしてオリジナル画像の該当箇所に戻され 縦に $x-2$，横に $y-2$ のアンカー（図 6 の赤い点）が均等に設定される．そのアンカーが領域予測の中心であるとみなして，すべてのアンカーにおいて SCALES と RATIOS のすべての組み合わせからなるアンカーボックス（図 6 の赤い枠）が設定される．

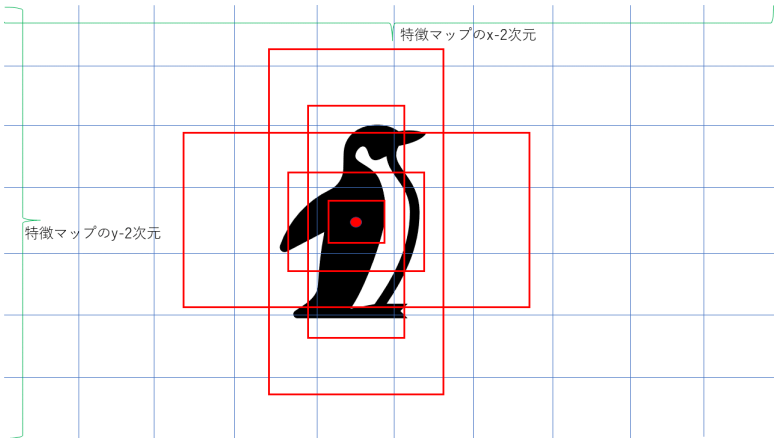


図 6: Anchor と Anchorbox

アンカーボックスを設定したのち，Classification ブランチと Regression ブランチの二つに分かれて処理が行われる．

- Classification ブランチ
  このブランチでは一つ一つのアンカーそれぞれに対するアンカーボックスが囲む領域に対してその領域が対象オブジェクトを含んでいるのかいないのかのニクラス識別を行うために，$\phi(Z)$ と $\phi(X)$ は畳み込みを行った後に拡張され，$[\phi(Z)]_{cls}$ と $[\phi(X)]_{cls}$ になる．このふたつの相関が計算され，各アンカーにおけるそれぞれのアンカーボックスに対して，オブジェクト対象を含むか含まないかを得る．図 5 の Classification ブランチの計算結果である $A^{cls}_{w\times\ h\times\ 2k}$ の各点は 2k 個のチャンネルベクトルが含まれ，これはオリジナルの画像上の対応する位置の各アンカーにおいて，それぞれのアンカーボックスが囲んでいるのがオブジェクトであるか（正）そうでなく背景か（負）の活性化を表している．これは Softmax 関数で制御されている．正解バウンディングボックスが囲む領域を S_{ans}，アンカーボックスが囲む領域を S_{anchor} としたとき，検出精度を表す IoU (Intersection over Union) は次のように定義される．

$$IoU = \frac{S_{ans} \cap S_{anchor}}{S_{ans} \cup S_{anchor}} \tag{6}$$

  IoU が 0.7 以上でアンカーボックスはオブジェクトを囲んでいるものとし，0.3 以下でアンカーボックスは背景を囲んでいるものとする．それ以外はオブジェクトでも背景でもないものと認識する．

- Regression ブランチ
  このブランチでは一つ一つのアンカーそれぞれに対するアンカーボックスが囲む領域

に対して正解バウンディングボックスとの相対位置を予測するために，$\phi(Z)$ と $\phi(X)$ は畳み込みを行った後に拡張され，$[\phi(Z)]_{reg}$ と $[\phi(X)]_{reg}$ になる．この二つの相関が計算され，各アンカーに対して，正解バウンディングボックスとのずれを得る．IoU が 0.7 以上あるいは 0.3 以下のもののみを学習に用いる．計算結果の $A^{reg}_{w\times\ h\times\ 4k}$ の各点は $4k$ 個のチャンネルベクトルが含まれていて，バウンディングボックスとアンカーのずれの距離を示す $dx,\ dy,\ dw,\ dh$ を表す．．$A_x,\ A_y,\ A_w,\ A_h$ をそれぞれアンカーボックスの中心座標と幅と高さ，$T_x,\ T_y,\ T_w,\ T_h$ を正解バウンディングボックスの中心座標と幅と高さとしたとき，正規化した距離は，

$$\begin{aligned} \delta[0] = \frac{T_x - A_x}{A_w}, \quad \delta[1] = \frac{T_y - A_y}{A_h} \\ \delta[2] = ln\frac{T_w}{A_w}, \quad \delta[3] = ln\frac{T_h}{A_h} \end{aligned} \tag{7}$$

と書ける．

以上の結果を用いて損失関数を計算し学習する．

3.3.2 Siamese RPN における損失関数

Faster R-CNN [8] で用いられていた損失関数を利用することとする．オブジェクトか否かの分類のための損失はクロスエントロピー損失であり，次のように表される．

$$L_{cls}(p_i, p_i^*) = -\sum_i p_i^* log(p_i) \tag{8}$$

ただしここで，i はバッチ内でのアンカーボックスのインデックス，p_i はアンカー i にオブジェクトが含まれる予想確率，p_i^* はアンカーが正例なら 1，負例なら 0 をとるものである．p_i と p_i^* の確率分布が似ていると交差エントロピー誤差は小さくなる．領域の損失については，正規化座標を用いた $smooth_{L_1}$ 損失が利用される．$smooth_{L_1}$ 損失は次のように表される．

$$smooth_{L_1}(x) = \begin{cases} 0.5x^2, & |x| < 1 \\ |x| - 0.5, & |x| \geq 1 \end{cases} \tag{9}$$

以上より領域予想における損失は以下のように書ける．

$$\begin{aligned} L_{reg}(t_i, t_i^*) &= smooth_{L_1}(t_i - t_i^*) \\ &= smooth_{L_1}(\delta) \end{aligned} \tag{10}$$

ただし，t_i は予測したアンカーボックス，t_i^* は正解バウンディングボックスである．最後に損失関数を最適化する．

$$loss = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i * L_{reg}(t_i, t_i^*) \tag{11}$$

ここで λ は二つのバランスをとるハイパーパラメータであり，N_{cls} はバッチのサイズ，N_reg はアンカーの数である．領域損失において，重み p_i* は今見ているアンカーボックス i が正例であるときのみ 1 になるので，領域囲みの精度が良いときのみ学習に用いる．

3.4 Siamese RPN におけるモデル学習

図 5 の Siamese Network において，CNN の構成，用意する学習用データセットによって特徴量抽出の学習結果は異なる．今回は学習用データセットに Youtube-BB (YTBB) データセット [9] を用いた．このデータセットはアノテーションされ，正解バウンディングボックスを保持している．Siamese RPN が，取り出した特徴量に準じて予測するバウンディングボックスに対応する場所の特徴量マップを取り出す．よって，本手法の CNN の学習は，特徴量の取り出し方のみではなく，間接的に次の段階である RPN に影響する．CNN について，二通りの構造（三層，四層）を用意し，層の出力も複数種類のものを用意した．

モデル名	1 層目次元数	2 層目次元数	3 層目次元数	4 層目次元数
model A	24	32	16	
model B	24	24	8	
model C	32	16	16	4
model D	16	16	16	16
model E	24	32	32	16

表 2: CNN 構成

三層及び四層の CNN における特徴量抽出部分は，オートエンコーダを構成し，入力画像と出力画像が同じになるような学習を行う（図 7）．この特徴量抽出器において，classification ブランチと regression ブランチをチューニングする．

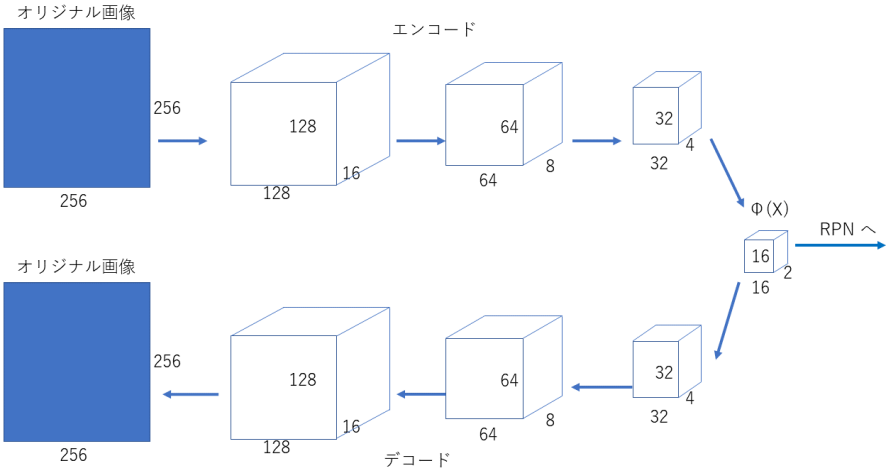


図 7: オートエンコーダによる学習

また，文献 [5] の元来の手法である，AlexNet [10] [11] を特徴量抽出として用いる学習（model F）も試みた．この時 AlexNet はすでに学習済みモデルであるので，特徴量抽出部分は変更せず，classification ブランチと regression ブランチをチューニングする．なお AlexNet によって抽出された特徴量マップの構成は，図 5 の Siamese Network におけるテンソルそれぞれである．

3.5 BAM における学習

BAM への入力は，オブジェクト k についての対象映像の 1 から最終フレーム F までの予測バウンディングボックスから取り出される N 次元の特徴量を標準化したものを各オブジェクトごとに連続して並べたものである．標準化する前の特徴量集合を M とすると，M は，以下のように表せる．

$$\begin{aligned} M = [&[x_1^{1,1}, x_1^{1,2}, \ldots, x_1^{1,N}], \\ &[x_1^{2,1}, x_1^{2,1}, \ldots, x_1^{2,N}], \\ &\cdots \\ &[x_1^{F,1}, x_1^{F,1}, \ldots, x_1^{F,N}], \\ &\cdots \\ &[x_K^{F,1}, x_K^{F,2}, \ldots, x_K^{F,N}]] \end{aligned} \tag{12}$$

ただし，$k\ (1 \leq k \leq K)$（K は全オブジェクト数）はどのオブジェクトか，$n\ (1 \leq n \leq N)$ はどの特徴量か，$f\ (1 \leq f \leq F)$ は何フレーム目かを表し，$x_k^{f,n}$ は オブジェクト k の f フレーム目の n 番目の特徴量を表す．

BAM に記憶させる特徴量は以降の正規化を行う．それぞれのオブジェクトの任意の 1 フレーム分を抜き出す．この集合を I とする．

$$\begin{aligned} I = [&[x_1^{f_1,1}, x_1^{f1,2}, \ldots, x_1^{f1,N}], \\ &[x_2^{f_2,1}, x_2^{f2,2}, \ldots, x_2^{f2,N}], \\ &\cdots \\ &[x_K^{f_K,1}, x_K^{f_K,2}, \ldots, x_K^{f_K,N}]] \end{aligned} \tag{13}$$

ここで，I の平均 $\bar{\mu}$ と分散 ${S_\mu}^2$ を求める．

$$\bar{\mu} = \frac{1}{K}\sum_{k=1}^{K} I[k] \tag{14}$$

$${S_\mu}^2 = \frac{1}{K}\sum_{k=1}^{K} (I[k] - \bar{\mu})^2 \tag{15}$$

その後，正規化を行う．ここで，$l(1 \leq l \leq K * F)$ とする．

$$M'[l] = \frac{M[l] - \bar{\mu}}{S_\mu} \tag{16}$$

$$I'[k] = \frac{I[k] - \bar{\mu}}{S_\mu} \tag{17}$$

すべての M の成分に対して式（16）による標準化を行った M' が BAM への入力となり，すべての I の成分に対して式（17）による標準化を行った I' を BAM のアトラクターに記憶する．

4　実験による性能評価

特徴量抽出のための CNN の構成変更や，BAM　へ記憶させるアトラクターの変更により認識バウンディングボックスがどのように変化するのか，またそれらから取り出す特徴量の変化により，BAM における推定ががどのように変化するのかを観測し，BAM への入力が有効な特徴量抽出を行えているアーキテクチャを調査した．

4.1　実験環境

実験環境を表 3 に示す．

OS	Ubuntu 18.04.06 LTS
CPU	11th Gen Intel Core i7-11850 @ 2.50GHz ×　16
メモリ	14.8GB
GPU	NVIDIA RTX A4000

表 3: 実験環境

Siamese Network の CNN には表 2 で定義したモデル A，B，C，D，E，のうち，三層から一つ（モデル A），四層から一つ（モデル D）に加えて，CNN が AlexNet であるモデル F を使用した．テストには形状，大きさ，質感，重さ，剛性などの異なる日常的なオブジェクトの画像データの一連のセットである YCB データセット [12] を用いた．今回は YCB データセットの中から，固定点にある 4 オブジェクトが移動するカメラにより撮影される全 1111 フレームの映像データセットを用いた．それぞれにおいてアノテーションが行われており，各オブジェクトに関する正解バウンディングボックスもあらかじめ得られている．この情報はバウンディングボックスの推定精度の評価に用いる．

Siamese Network のテンプレート画像として，612 フレーム目の画像において正解バウンディングボックス情報から切り出した各オブジェクトを設定する（図 8）．612 フレーム目を選定した理由は以下の通りである．切り出したオブジェクト画像から取り出す特徴量は 1111 フレームのオブジェクト推定の基準特徴量となるため，他のオブジェクトの特徴量が入り込む影響を少なくする必要やそのオブジェクトから汎用的な特徴量を取り出す必要がある．後者について，仮にオブジェクトを真上からみたものをテンプレートとして与えた場合，そこから取り出す特徴量は真上の情報のみとなってしまい，上と側面で色が全く違うオブジェクトなどを別のものとして判別してしまう可能性が大きい．こうしたことから，主観による評価で，切り出した時に比較的オブジェクト全体を捉えていた 612 フレーム目から切り出した各オブジェクトをテンプレートとして与えた．

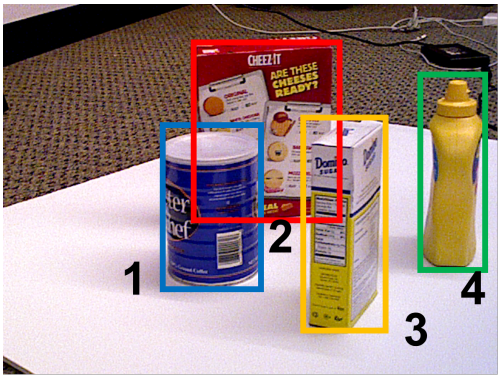
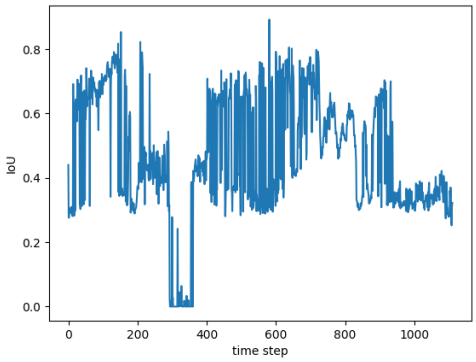


図 8: Siamese Network に入力するフレーム 612 から取り出すテンプレート画像

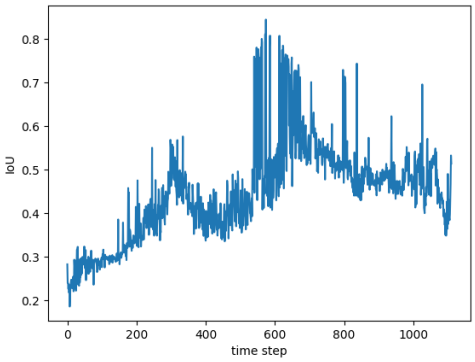
IoU が高いところが実際に見たいオブジェクトを切り出せているところで，その領域から得られる特徴量は実際にそのオブジェクトを表す特徴量に近くなるので，Siamese RPN による領域推定結果を確認したのち，IoU が高いフレームの予測バウンディングボックスから取り出した特徴量を BAM のアトラクターとして設定した．

4.2 三層の CNN [24, 32, 16] からなる特徴量抽出器

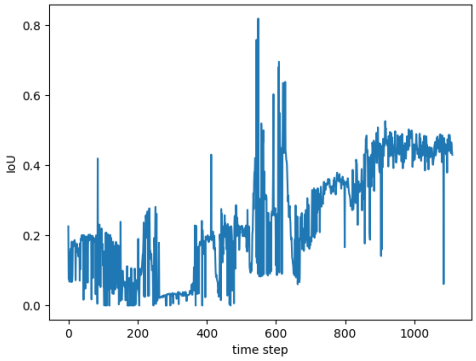
Siamese Network の CNN 構成を 三層で層の出力をそれぞれ 24，32，16 とした特徴量抽出器モデル A の各タイムステップにおける IoU を図 9 に示す．



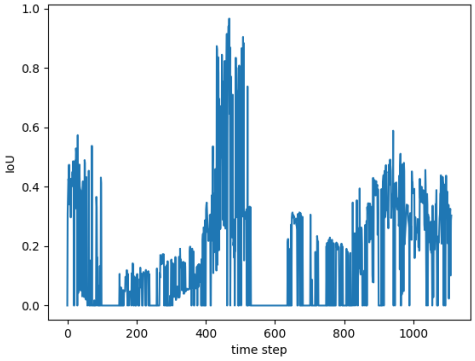
(a) オブジェクト 1 の IoU



(b) オブジェクト 2 の IoU



(c) オブジェクト 3 の IoU



(d) オブジェクト 4 の IoU

図 9: モデル A における各オブジェクトの IoU のタイムステップ変化

次に Siamese RPN の結果と計算時間を表 4 に示す.

	オブジェクト 1	オブジェクト 2	オブジェクト 3	オブジェクト 4
平均 IoU	0.484	0.452	0.241	0.178
平均計算時間 [s]	0.0293	0.0297	0.0304	0.0294

表 4: モデル A の Siamese RPN の計算結果

この CNN はオートエンコーダで学習しているので，テンプレート画像と同等である 612 フレーム付近の IoU は他のタイムステップより比較的高くなる傾向にある．このように取り出した特徴量を標準化したのち BAM に入力として与えた．BAM のアトラクターに記憶させる特徴量は，表 5 に設定した．

オブジェクト 1 のアトラクター	63 フレーム目
オブジェクト 2 のアトラクター	574 フレーム目
オブジェクト 3 のアトラクター	915 フレーム目
オブジェクト 4 のアトラクター	447 フレーム目

表 5: モデル A シミュレーション時の各オブジェクトのアトラクターの選び方

このときの BAM での認識結果は図 10 のようになった.

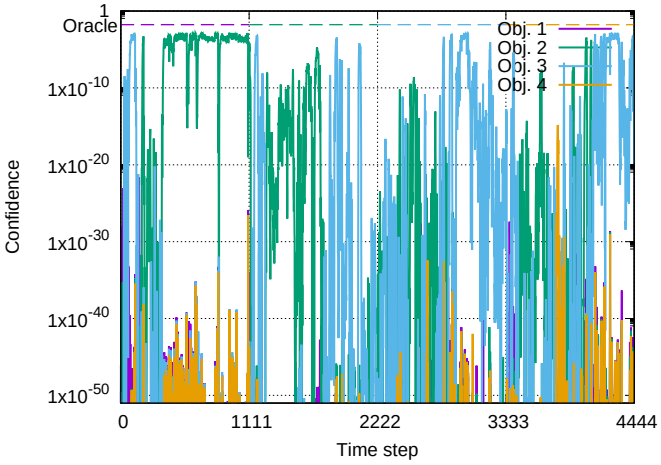


図 10: モデル A で取り出した特徴量に対する BAM のオブジェクト信頼度

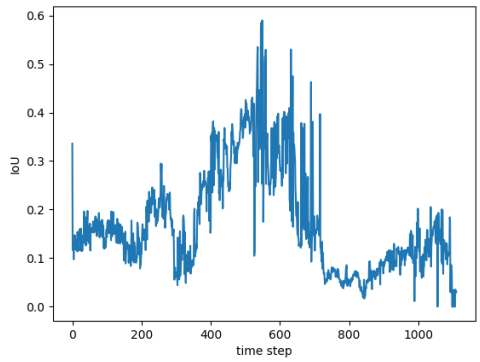
各ステップにおける信頼度が一番高い選択肢を意思決定結果とするとき，認識精度を表 6 に示す.

オブジェクト 1 の認識精度 [%]	5.94
オブジェクト 2 の認識精度 [%]	55.4
オブジェクト 3 の認識精度 [%]	62.1
オブジェクト 4 の認識精度 [%]	6.93
全体 [%]	32.6
計算時間 [s/f]	0.0539

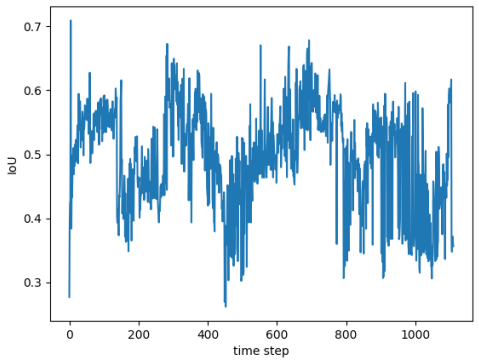
表 6: モデル A で抽出した特徴量における BAM の認識精度

4.3 四層の CNN [16, 16, 16, 16] からなる特徴量抽出器

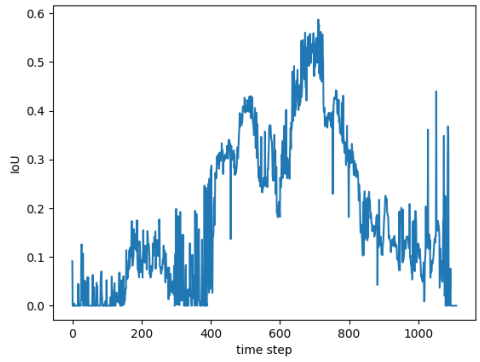
Siamese Network の CNN 構成を 四層で層の出力をそれぞれ 16，16，16，16 とした特徴量抽出器モデル D の各タイムステップにおける IoU を図 11 に示す．



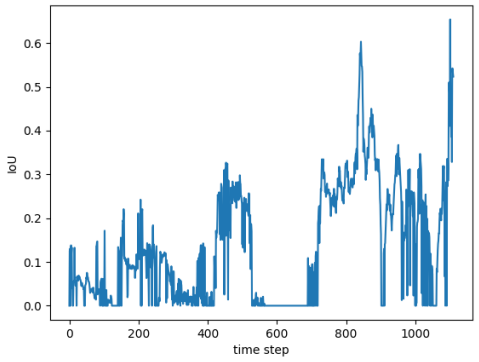
(a) オブジェクト 1 の IoU



(b) オブジェクト 2 の IoU



(c) オブジェクト 3 の IoU



(d) オブジェクト 4 の IoU

図 11: モデル D における各オブジェクトの IoU のタイムステップ変化

次に Siamese RPN の結果と計算時間を表 7 示す．

	オブジェクト 1	オブジェクト 2	オブジェクト 3	オブジェクト 4
平均 IoU	0.279	0.502	0.196	0.131
平均計算時間 [s]	0.0243	0.0238	0.0233	0.0236

表 7: モデル D の Siamese RPN の計算結果

前章同様，テンプレート画像と同等である 612 フレーム付近の IoU は他のタイムステップより比較的高くなる傾向にある．このように取り出した特徴量を標準化したのち BAM に

入力として与えた．BAM のアトラクターに記憶させる特徴量は，表 8 のとおり設定した．

オブジェクト 1 のアトラクター	548 フレーム目
オブジェクト 2 のアトラクター	666 フレーム目
オブジェクト 3 のアトラクター	697 フレーム目
オブジェクト 4 のアトラクター	842 フレーム目

表 8: モデル D シミュレーション時の各オブジェクトのアトラクターの選び方

このときの BAM での推定結果は図 12 のようになった．

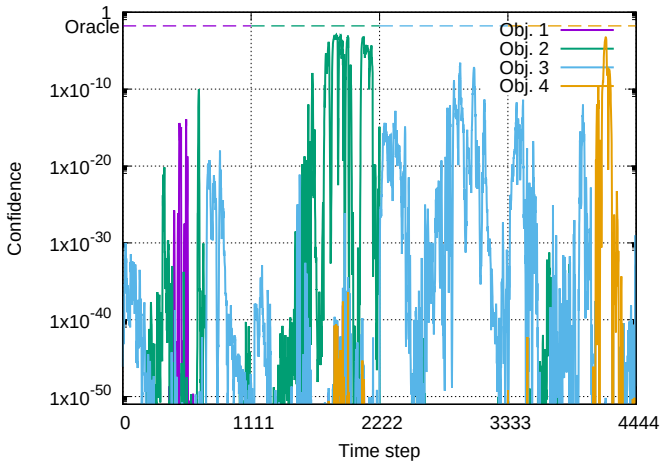


図 12: モデル D で取り出した特徴量に対する BAM のオブジェクト信頼度

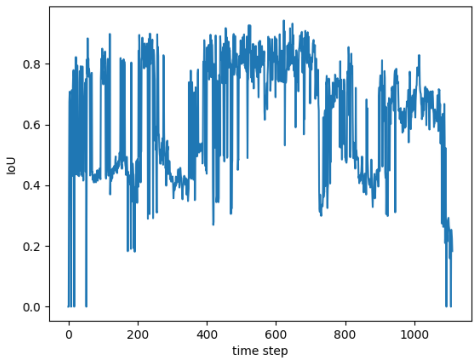
各ステップにおける信頼度が一番高い選択肢を意思決定結果とするとき，認識精度を表 6 に示す．

オブジェクト 1 の認識精度 [%]	13.2
オブジェクト 2 の認識精度 [%]	75.5
オブジェクト 3 の認識精度 [%]	94.5
オブジェクト 4 の認識精度 [%]	22.2
全体 [%]	51.4
計算時間 [s/f]	0.0393

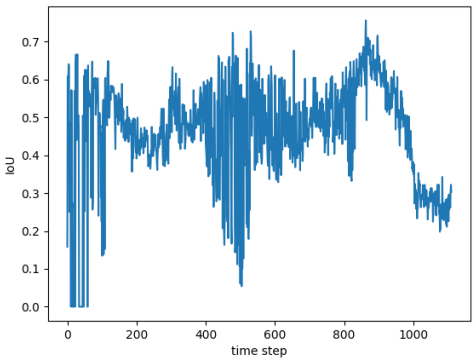
表 9: モデル D で抽出した特徴量における BAM の認識精度

4.4 AlexNet からなる特徴量抽出器

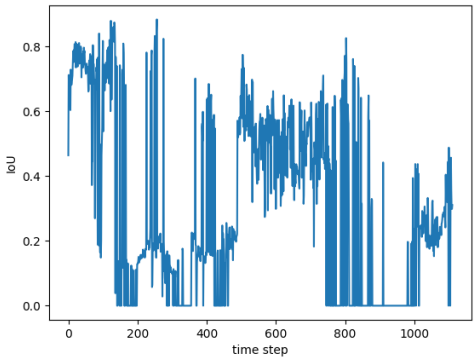
Siamese Network の CNN 構成を AlexNet とした特徴量抽出器モデル F の各タイムステップにおける IoU を図 13 に示す.



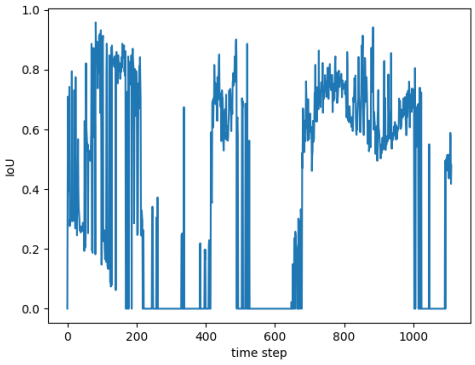
(a) オブジェクト 1 の IoU



(b) オブジェクト 2 の IoU



(c) オブジェクト 3 の IoU



(d) オブジェクト 4 の IoU

図 13: モデル F における各オブジェクトの IoU のタイムステップ変化

次に Siamese RPN の結果と計算時間を表 10 に示す.

	オブジェクト 1	オブジェクト 2	オブジェクト 3	オブジェクト 4
平均 IoU	0.626	0.461	0.306	0.382
平均計算時間 [s]	0.0509	0.0488	0.0484	0.0490

表 10: モデル F の Siamese RPN の計算結果

平均的な IoU がモデル A や D と比べて大きいが，計算時間は約 2 倍となっている．このように取り出した特徴量を標準化したのち BAM に入力として与えた．BAM のアトラク

ターに記憶させる特徴量は，表 11 に示す通りである．

オブジェクト 1 のアトラクター	509 フレーム目
オブジェクト 2 のアトラクター	873 フレーム目
オブジェクト 3 のアトラクター	37 フレーム目
オブジェクト 4 のアトラクター	153 フレーム目

表 11: モデル F シミュレーション時の各オブジェクトのアトラクター

このときの BAM での認識精度は図 14 に示すとおりである．

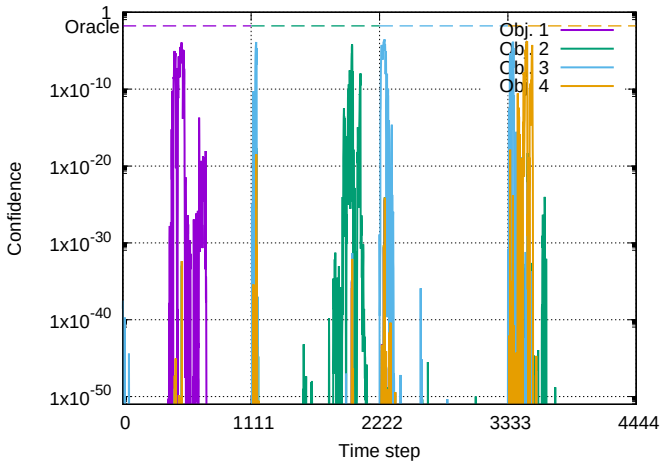


図 14: モデル F で取り出した特徴量に対する BAM のオブジェクト信頼度

各ステップにおける信頼度が一番高い選択肢を意思決定結果とするときの認識精度を表 6 に示す．

オブジェクト 1 の認識精度 [%]	69.9
オブジェクト 2 の認識精度 [%]	82.7
オブジェクト 3 の認識精度 [%]	40.1
オブジェクト 4 の認識精度 [%]	24.8
全体 [%]	54.36
計算時間 [s/f]	0.0657

表 12: モデル F で抽出した特徴量における BAM の認識精度

4.5　考察

平均 IoU が 0.5 近いかもしくは超えているであるモデル A のオブジェクト 2，モデル D のオブジェクト 2，モデル F のオブジェクト 1，2 は BAM によるオブジェクト推定の精度がそれぞれ 55.4%，76%，69.9%，82.7%と比較的高く出ている．しかし，特にモデル A のオブジェクト 1 は IoU が 0.484 に対して，オブジェクト推定精度は 5.94%，モデル D のオブジェクト 3 は IoU が 0.196 に対してオブジェクト推定精度は 94.7%とそれぞれ平均 IoU に対しかなり離れた値が観測できる．この点について，まずモデル A のオブジェクト 1 に対するバウンディングボックス推定の 1 フレームとモデル D のオブジェクト 3 に対するバウンディングボックス推定のあるフレームを取り出した（図 15）．

(a) モデル A のオブジェクト 1 の 領域推定 162 フレーム目　(b) モデル D のオブジェクト 3 の領域推定 401 フレーム目

図 15: モデル A のオブジェクト 1 とモデル D のオブジェクト 3 の領域推定

図 15(a) はオブジェクト 1 の領域を推定しているので，IoU は図 9 からも 0.4 前後と推測できる．また図 15(b) はオブジェクト 3 の領域を推定しているので，IoU は図 11 からも 0.1 前後と推測できる．IoU の比較ではモデル A のオブジェクト 1 における領域推定のほうがモデル D のオブジェクト 3 における領域推定よりも優れているように見えるが，図 15(a) でみられるように，この予想バウンディングボックス内にはオブジェクト 2 が含まれている割合が大きい．よってこのバウンディングボックスから特徴量を取り出すとき，オブジェクト 2 の特徴がかなり含まれる．実際に図 10 を見ると，162 フレーム目ではオブジェクト 2 の確信度が高まっていることが分かる．また，図 15(b) でみられるように，この予想バウンディングボックス内に，ほかのオブジェクトはほとんど見られず，含まれているオブジェクト 2 もオブジェクト 3 と比べて小さい．IoU が小さいものの，オブジェクト推定自体はできていると考えられる．

また，考察点として，与えるアトラクターの違いが考えられる．今回は予測バウンディン

グボックスの中からよい IoU のものから取り出した特徴量を与えたが，例えば図 11(c) を見ると，最大でも IoU が 0.6 ほどとなっている．つまり与えたアトラクターが持つ特徴量はそのオブジェクトを示すものである割合が小さいものとなっている．これにより背景などが，そのオブジェクトを示すものとしてとらえられてしまっていると考えられる．

これらから IoU は大きい値であることの重要性が高く，またアトラクターはそのものの特徴量であることの重要性が高いことが分かった．これを満たすためにはバウンディングボックスの切り出し精度がまず第一に考えなければいけない課題である．その点において，AlexNet による切り出しは，IoU の観点からすると有効であると言える．

5　おわりに

本報告では，人の脳の備える，不確実な観測データから高精度な意思決定を行う特徴を用いたオブジェクト認識手法の実装を行った．我々の先行研究において用いていた，あらかじめ与えられた映像領域から特徴量を取り出すアーキテクチャを拡張し，映像領域から認識を行いたいオブジェクトの存在する領域の推定と，領域からの特徴量抽出とを同時に行うアーキテクチャの実装を行った．領域推定の手順としては，まず映像領域全体に対する特徴量マップと，検出したいオブジェクトのみが写ったテンプレート画像に対する特徴量マップを取り出し，その特徴量マップによる畳み込み演算を行い，領域推定結果を出力する．映像領域全体に対する特徴量マップから推定領域に対応する位置の特徴量を取り出し，これをBAM への入力として用いた．公開データセットである YCB データセットを用いた評価の結果，BAM におけるオブジェクト認識精度は 54.36%であった．計算時間については，1 フレームの映像に対して Siamese RPN による領域推定と特徴量取り出しに平均 0.0473 秒を要し，BAM による認識に平均 0.0666 秒を要した．合計では 0.1 秒以上となり，10fps 以下である．一般に動画像に用いられるフレームレートである 30fps や 60fps よりも遅いため，実用に向けて計算時間の削減が必要である．認識精度に関しては，機械学習を用いたフレームワークで向上することは，計算時間の観点から現実的ではなく，映像以外の情報を用いた認識結果とのマルチモーダル統合認識が有力であると考えている．この手法の提案，実装，および評価も今後の課題である．

謝辞

本報告を終えるにあたり，大阪大学大学院情報科学研究科の村田正幸教授には，御多忙の中貴重なご指導を賜りましたこと深謝いたします．ならびに，研究の方針や進捗，行き詰った点などにも助言，ご指導を手厚くしていただきました，大阪大学大学院情報科学研究科の小南大智助教に心より感謝申し上げます．また，平素よりご指導いただきました大阪大学大学院情報科学研究科の荒川伸一准教授，大阪大学先導的学際研究機構大下裕一准教授に，厚く感謝申し上げます．最後に日々の学生生活を支えてくださった研究室の皆様に感謝の意を表して謝辞とさせていただきます．

参考文献

[1] N. Nikolakis, V. Maratos, and S. Makris, “A cyber physical system (CPS) approach for safe human-robot collaboration in a shared workplace,” *Robotics and Computer-Integrated Manufacturing*, vol. 56, pp. 233–243, 2019.

[2] B. He and K.-J. Bai, “Digital twin-based sustainable intelligent manufacturing: A review,” *Advances in Manufacturing*, vol. 9, no. 1, pp. 1–21, 2021.

[3] S. Bitzer, J. Bruineberg, and S. J. Kiebel, “A Bayesian attractor model for perceptual decision making,” *PLoS Computational Biology*, vol. 11, no. 8, p. e1004442, 2015.

[4] 関良我，小南大智，下西英之，村田正幸，藤若雅也，野上耕介, “脳のマルチモーダルな情報処理に着想を得た物体推定手法の提案と評価,” **電子情報通信学会技術研究報告**（CQ2021-14）, vol. 121, no. 15, pp. 59–64, 2021.

[5] L. Bo, Y. Junjie, W. Wei, Z. Zheng, and H. Xiaolin, “High performance visual tracking with siamese region proposal network,” *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, no. 1, pp. 8971–8980, 2018.

[6] G. Koch, R. Zemel, R. Salakhutdinov *et al.*, “Siamese neural networks for one-shot image recognition,” *ICML deep learning workshop*, vol. 2, no. 1, p. 8, 2015.

[7] G. Yao, C. Rui, T. Ying, C. Xuehong, and L. Ruiyu, “Combining siamese network and regression network for visual tracking,” *IEICE Transactions on Information and Systems*, vol. E103.D, no. 8, pp. 1924–1927, 2020.

[8] R. Shaoqing, H. Kaiming, G. Ross, and S. Jian, “Faster R-CNN: towards real-time object detection with region proposal networks,” *NIPS’15: Proceedings of the 28th International Conference on Neural Information Processing Systems*, vol. 1, no. 1, pp. 91–99, 2015.

[9] “Youtube-boundingboxes dataset,” available at https://research.google.com/youtube-bb/,Accessed : 2021-12-20.

[10] K. Alex, S. IIIya, and E. H. Geoffrey, “Imagenet classification with deep convolutional neural networks,” *Advances in Neural Information Processing Systems*, vol. 25, no. 1, p. 1097–1105, 2012.

[11] R. Olga, D. Jia, J. Krause, S. Sanjeev, M. Sean, H. Zhiheng, K. Andrej, K. Aditya, B. Michael, C. Alexander, and F.-F. Li, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[12] "YCB benchmarks–object and model set," available at http://www.ycbbenchmarks.com/,Accessed : 2021-12-20.