

領域分類可能な 3次元物体生成モデル

Semantic Segmentation for 3D object generation model

高知大学 理工学部 情報科学科	
卒業論文	
卒業年度	令和7年度(2025年度)
指導教員	木脇 太一
氏名	中須賀 大輝

目次

第 1 章	序論	3
1.1	概要	3
1.2	導入	4
1.3	本論文の構成	4
第 2 章	先行研究	5
2.1	先行研究	5
第 3 章	研究手法	7
3.1	SAM を用いた多視点画像の領域分類	7
3.2	画像クラスタリングに用いる 3 つのアルゴリズムの概要	8
3.3	クラスタリング精度の評価方法	10
第 4 章	実行と結果	12
4.1	データセット	12
4.2	評価方法	13
4.3	結果	14
4.4	3D オブジェクトの生成	24
第 5 章	考察と課題	26
5.1	考察と今後の課題	26
謝辞		27
参考文献		28

第1章

序論

1.1 概要



図 1.1: 3DGS で出力した結果

近年, 図 1.1 のように多視点画像から 3 次元オブジェクトを生成する 3D Gaussian Splatting (3DGS)[5] に領域分類を組み込む取り組みが大きな注目を集めている。従来のアプローチでは, 主に 2D モデルである画像と言語を対応付ける CLIP[13] や, 教師なしで領域分類を行う Segment Anything Model (SAM)[6] を用いた研究が行われている.[12][7][18][8] しかしながら, これらの手法はテキストを介して 3D シーンの一部に対して領域分類を行うのが主流であり, 検知した全ての 3D シーンに対して一括して領域を実施する手法は未だ十分に研究されていない。

そこで, 本研究では, テキストを入力せずに SAM[6] を用いて全ての多視点画像に対して領域分類を実施し, 自動的に領域分類された 3D オブジェクトを生成することを目的とする。

本研究の手法により, テキスト情報を介さずに画像内の全てのオブジェクトに対して領域分類を行い, 3D オブジェクトを生成するシステムを構築する。

1.2 導入

Neural Radiance Fields (NeRF)[9] の登場により、多視点画像から高精度かつフォトリアリストイックな 3D オブジェクトの再構築が可能となり、3D コンテンツの生成に大きな注目が集まっている。従来の手法は主に、画像と言語を対応付ける CLIP[13] や、教師なしで領域分類を実施する Segment Anything Model (SAM)[6] などの 2D モデルを利用し、テキストを介した 3D シーンの一部への領域分類が主流であった。本研究で扱う 3D シーンとは、全ての物体の形状、配置、色彩などの視覚情報のことを指す。しかしながら、これらのアプローチは、テキストを入力しなければならなかったため、3D シーンの全オブジェクトに対して一括して領域分類を実施し、各オブジェクトを出力することは困難である。

そこで、本研究ではテキスト情報を用いずに、SAM[6] を活用して多視点画像全体に対して自動的に領域分類を行い、得られたセグメントを基に 3D オブジェクト全体のオブジェクトを抽出する手法を提案する。本手法の核となるのは、多視点画像のクラスタリングである。具体的には、以下のステップによりシステムを構築する。

- (i) 全ての多視点画像に対して SAM[6] を用いて各画像内のオブジェクト毎に領域分類を実施する。
- (ii) 領域分類結果に対して、cosine 類似度に基づくクラスタリングを行い、同一オブジェクトに属すると思われるセグメントを自動的にグループ化する。この際、オブジェクトが重なったり、撮影条件により一部が見切れてしまうケースが存在するため、そのような不確実なセグメントは、3D オブジェクト生成に有用でないと判断し、あえて一つのクラスタにまとめる方針を採用了した。
- (iii) クラスタリングされた各グループに基づき、個々の 3D オブジェクトを生成する。

1.3 本論文の構成

本論文は、全 5 章から構成される。

第 1 章 序論。

第 2 章 先行研究。

第 3 章 提案手法。

第 4 章 結果

第 5 章 考察と今後の課題。

第 2 章

先行研究

2.1 先行研究

Neural Radiance Fields (NeRF)

多視点画像からの 3 次元再構築において、近年最も注目を集めているのが Neural Radiance Fields (NeRF)[9] である。NeRF は、シーン内の 3 次元座標と視線方向を入力とし、その点からの放射輝度およびボリューム密度をニューラルネットワークで出力させる手法である。レンダリングはボリュームレンダリングの原理に基づき実行し、複数視点から撮影した画像のフォトメトリック損失を逆伝搬することでネットワークを学習する方式を取る。ここでフォトメトリック損失とは以下のようなものである。

$$\mathcal{L}_{\text{photo}} = \sum_{r \in \mathcal{R}} \left\| \hat{C}(r) - C(r) \right\|_2^2, \quad (2.1)$$

ここで、

- \mathcal{R} は訓練画像からサンプリングされた全てのレイの集合、
- $\hat{C}(r)$ はレイ r に沿ってネットワークがレンダリングしたカラー、
- $C(r)$ は対応する実際の（グラウンドトゥルースの）ピクセルカラーを示します。

NeRF は高い視覚的再現性を得られる一方、推論段階で大量のレイトレーシングを行う必要があり、リアルタイム応用や大規模シーンへの適用には依然として課題が残る。

3D Gaussian Splatting (3DGS)

これに対し、3D Gaussian Splatting (3DGS)[5] は、シーン中の各点を単なる座標ではなくガウス分布（位置・共分散・色など）として表現し、スプラット処理によって高速レンダリングを可能にする手法である。3DGS[5] では、NeRF[9] と同様に微分可能なレンダリングフレームワーク上でガウスのパラメータをフォトメトリック誤差に基づき最適化するため、多視点画像から 3 次元再構築が行える。点群ベースである 3DGS[5] は、軽量かつ高速な表示が期待できる点が特徴となっている。

さらに、ガウス分布に表面法線を埋め込む手法として Surface-Aligned Gaussian Splatting (SuGaR)[3] が提案されており、SuGaR[3] により、高精度なメッシュ再構築へ直接つなげる研究

も進められている。

CLIP と SAM との組み合わせ

最近では、CLIP[13] や SAM[6] といった大規模モデルを 3DGS[5] と統合し、テキスト情報を介して興味領域を抽出・セグメンテーションする取り組みも報告されている [6][12][7][18][8][2]。たとえば、CLIP を使って “chair” や “table” といったキーワードに合致するガウス分布を抽出したり、SAM を使って任意の画像領域をセグメンテーションし、それらを 3DGS 上で統合するなどの実験が報告されている。一方で、こうした手法の多くはテキスト情報を前提とし、ユーザが指定したクラスや名称に合致するオブジェクトを選択的に再構成・表示できるという特徴があるを用いた研究が行われている。

K-means

K-means[1] は、クラスタリング手法の一つで、各データ点をあらかじめ定められたクラスタ数 K に分割する手法である、目的は各クラスタ内部の分散、すなわち各データ点と所属するクラスタ中心との距離の 2 乗和を最小化することである、アルゴリズムは、初期にランダムにクラスタ中心を設定し、各データ点を最も近い中心に割り当てる、その後、割り当てに基づいて各クラスタの中心を再計算する、この操作を中心が収束するまで繰り返す、簡便さと計算効率の高さから、K-means は多くの応用分野で広く利用されている。

第3章

研究手法

3.1 SAM を用いた多視点画像の領域分類

本研究では、用意した多視点画像から目的とするシーンを自動的に抽出するため、まず最新の領域分類手法である SAM[6] を用いて各画像をセグメント化する。しかし、単に SAM により各画像をセグメント化するだけでは、図 3.1 に示すように、同一のシーンに属すべき対象が複数のセグメントに分割される場合がある。そこで、本研究では、領域分類された各領域の面積や位置情報を比較し、重複している（すなわち、大部分の面積が一致する）セグメントを除外する処理を追加している。この操作は、全ての多視点画像に対して一律に適用され、最終的に各画像は図 3.2 のように目的とするシーン毎に正確に分割される。



図 3.1: SAM を使って多視点画像の一枚をオブジェクトのシーン毎に分けて出力した図

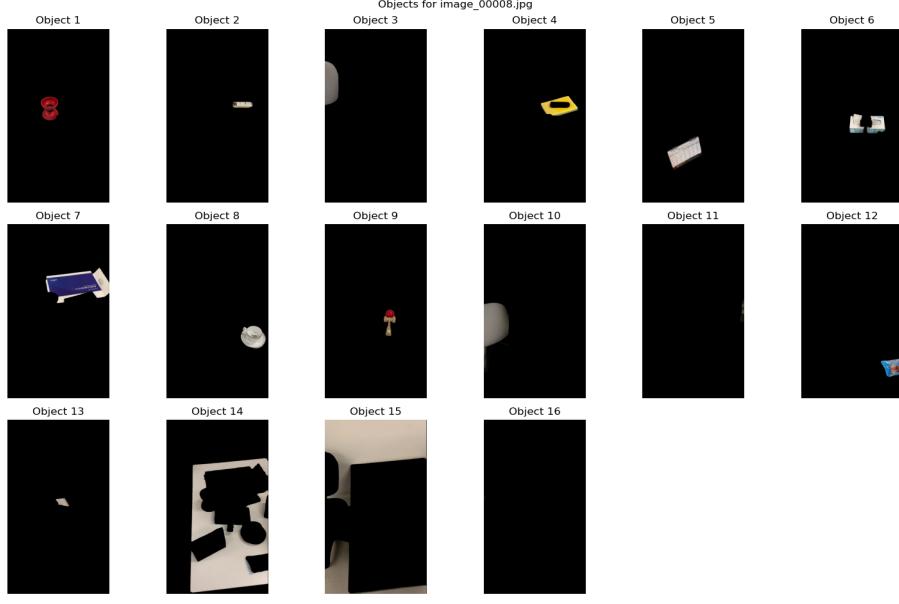


図 3.2: SAM を使って多視点画像の一枚を適切にオブジェクトのシーン毎に分けて出力した図

3.2 画像クラスタリングに用いる 3 つのアルゴリズムの概要

前節で述べたように, SAM[6] を用いることで, 多視点画像中の各画像から対象物ごとにシーンを分割することに成功した. すなわち, 各画像において, 写っているシーンが個々のオブジェクトに対応するセグメントとして抽出された. 次に, 抽出されたオブジェクト毎に, 画像のクラスタリングを実施する. これにより, 同一または類似するオブジェクトを持つ画像群がグループ化される.

本研究では、この画像のクラスタリングを行うために、以下の 3 つの異なるアルゴリズムを採用して比較を行った. それぞれの手法について、実装上の細部や特徴を以下に示す.

3.2.1 類似度・閾値ベースクラスタリング

この手法は、個々の画像間の cos 類似度に基づいて初期クラスタを生成し、さらに各クラスタの代表点（本実装では、メドイドまたはクラスタに最後に追加された画像特微量）同士の類似度が所定の閾値以上であればクラスタ同士を統合するというものである. この手法を採用した理由は、多視点画像の場合、カメラで連続して撮影されるため、 n 枚目と $n + 1$ 枚目の画像は非常に高い類似度を示す傾向にあることに着目した点にある. すなわち、クラスタにおいて最後に追加された画像（直近の画像）と新規に追加される画像との特微量の類似度を比較することで、連続するシーンの変化を正確に捉え、効率的かつ適切なクラスタリングが実現できると考えたためである.

- **初期クラスタ生成:** 各画像は、畳み込みニューラルネットワーク (CNN) の一種である resnet50[4] を用いて個別に特微量が抽出する. 抽出された特微量は、類似度計算を正確に行うために、各特徴ベクトルに対して L2 ノルム正規化を施す. すなわち、ある画像の特微量ベクトル $\mathbf{v} = [v_1, v_2, \dots, v_n]$ に対し、L2 ノルムは

$$\|\mathbf{v}\|_2 = \sqrt{v_1^2 + v_2^2 + \dots + v_n^2}$$

で定義され、正規化後のベクトルは

$$\mathbf{v}' = \frac{\mathbf{v}}{\|\mathbf{v}\|_2}$$

となる。この処理により、全ての画像の特徴量が同一スケールとなり、内積計算による cos 類似度が計算できるようになる。

次に、多視点画像においては、カメラの撮影条件から連続して撮影される画像、すなわち n 枚目と $n+1$ 枚目の画像は非常に高い類似度を示す傾向にある。この性質を利用し、本手法ではすべての多視点画像に対して、既に生成された各クラスタの最後に追加された画像の特徴量と、新たに抽出された画像の特徴量との内積（cos 類似度）を計算する。もし、この内積が所定の閾値 τ_1 以上であれば、その画像は既存のクラスタに追加される。一方、内積が τ_1 を超えなかった場合は、対象画像は新規のクラスタとして追加される。

このような処理により、隣接する多視点画像間の高い類似性を効率的に活用してクラスタリングを行うとともに、各画像の特徴量が正規化されることで、内積計算のみで類似度評価が可能となり、計算の効率化と精度の向上を実現している。

- **クラスタ統合:** 初期クラスタ生成後、各クラスタ内の画像から抽出された特徴量に基づいて、各クラスタの代表点として メドイド [20] を算出する。メドイドとは、クラスタ $C = \{x_1, x_2, \dots, x_n\}$ 内の各点 x_i について、全点との距離の総和

$$S(x_i) = \sum_{j=1}^n d(x_i, x_j)$$

が最小となる点 x_{i^*} を選ぶ手法である。ここで $d(x, y)$ はデータ点 x と y の間の距離を意味する、本研究ではコサイン距離

$$d(x, y) = 1 - \cos(x, y)$$

を使用する。クラスタの代表点として単純な算術平均を用いる方法と比較すると、メドイドは、クラスタ内の実際のデータ点の中から、外れ値の影響を受けにくい代表点を選出できるため、算術平均より頑健で解釈が容易である。したがって、本研究では、各クラスタの代表点としてメドイドを採用することにより、クラスタリングの精度向上および実際の画像の特徴をより正確に反映したクラスタ統合が可能になるとえた。その後、各クラスタのメドイド同士の cos 類似度を一括で計算し、所定の閾値 τ_2 以上であれば、Union-Find[16] を用いてクラスタを統合する。

- **類似度・閾値ベースクラスタリングの結果と後処理** 本研究における類似度・閾値ベースクラスタリングを用いた実験では、テストデータに対してクラスタリングを実施した結果、1~5 枚程度の小規模なクラスタが多数生成される傾向が確認された。しかし、3D オブジェクトの生成に必要な情報量を確保するためには、各クラスタ内に十分な数の画像が含まれている必要がある。そのため、実験結果から、クラスタ内の画像数が 5 枚以下のクラスタは、3D オブジェクト生成において有用でないと判断し、これらの小規模クラスタを評価および後処理の段階で排除する処理を導入した。

このアプローチにより、クラスタリング結果として得られたクラスタの中から、3D オブジェクト生成に実際に寄与する十分なサンプル数を持つクラスタのみを利用することができるようになった。

3.2.2 K-means ベース クラスタリング

K-means は、各画像の特徴量を用いてユークリッド距離の最小化を図りながらクラスタを分割する手法である。

- **前処理:** 各画像は ResNet50 で特徴量を抽出し、L2 ノルム正規化が施される。また Principal Component Analysis(PCA)[1] による次元削減を実施し、冗長な情報を除去する。本研究では、累積寄与率 0.95 を保持する最小の主成分数で次元削減を行う。K-means 法を用いる際に PCA による次元削減を実施することには、以下のようなメリットがある。
- **計算量の削減:** K-means は各データ点と各クラスタ中心との距離計算に依存するため、データの次元数に応じて計算量が増大する。PCA により低次元空間にデータを写像することで、各反復処理での距離計算が高速化され、アルゴリズムの収束が迅速になる。
- **ノイズおよび冗長情報の除去:** 元のデータには、しばしば冗長な情報やノイズが含まれている。PCA は、データの主要な分散方向に沿った成分のみを保持するため、クラスタリングにおいて冗長な次元やノイズの影響が排除され、K-means がより正確なクラスタリングを実施できるようになる。

3.2.3 HDBSCAN ベースクラスタリング

DBSCAN[20] は、密度に基づいたクラスタリング手法であり、データの密度に応じてクラスタを自動的に検出することが可能である。しかし、我々のテスト実験では、DBSCAN を適用した場合、全くクラスタリングが行われなかったり、一部の領域でのみクラスタが形成されるといった結果が観察された。そのため、よりロバストな手法として、HDBSCAN[20] を採用することにした。HDBSCAN は、データの局所的な密度構造を階層的に解析し、クラスタ数を自動で決定するため、ノイズや外れ値の影響を低減し、より安定したクラスタリング結果が得られると期待される。

- **前処理:** 各画像の特徴量は L2 ノルム正規化された状態で抽出され、さらに標準化される。PCA を用いて、累積寄与率 0.95 を保持する最小の主成分数で次元削減を行う。しかしながら、我々のテスト実験では、HDBSCAN を適用した場合、DBSCAN 同様に単純に累積寄与率 0.95 を基準とした次元削減では、全くクラスタリングが行われなかったり、一部の領域でのみクラスタが形成されるといった問題が観察された。このため、次元削減の際には、エルボー法 [20] を採用して各主成分の分散寄与率の変化（特に 2 階微分）が最も小さくなる転換点を検出し、最適な主成分数を数値的に推定する手法を用いた。

以上の手法を比較することで、最もデータの特性に応じた柔軟かつ高精度な画像クラスタリングが行えるアルゴリズムを探す。

3.3 クラスタリング精度の評価方法

本研究では、正解データと予測データの各画像に付与されたラベル情報を用いてクラスタリングの正答率を評価する。まず、両データにおいて共通する画像を対象とし、それぞれの画像に対する正解ラベルと予測ラベルを抽出する。次に、得られたラベル間の対応関係を混同行列により表現し、ハンガリ

アンアルゴリズムを用いて最適なラベル対応を求める。最終的に、この最適対応に基づいて正しく分類された画像の割合を正答率 Acc として、以下の式で定義する：

$$Acc = \frac{T}{N},$$

ここで T は最適対応により正しく分類された画像数、 N は対象とした画像の総数である。

第4章

実行と結果

4.1 データセット

4.1.1 多視点画像の枚数

本研究の多視点画像の枚数は 100 枚から 150 枚程度を用意している。テストデータではその内の 50 枚から 60 枚程度使用している。

4.1.2 テストデータの正解クラスタの作成

本研究では、テストデータの正解クラスタの作成に際して、各画像を手作業で振り分ける方法を採用した。これは、人が各画像を一枚ずつ確認し、同一のクラスに属するべき画像群を人力で割り当てる手法である。

しかし、多視点画像においては、以下のような問題が存在する。細かく個別のクラスタに振り分けることが極めて困難であった。

- (i) 多視点画像のカメラの位置条件の違いにより、同一オブジェクトであっても一部が見切れてしまう場合がある（図 4.1 の object9-12 参照）。
- (ii) 複数の画像において、オブジェクト同士が重なり合って写るケースが多く、同一オブジェクトとして認識することが難しい場合が存在する。

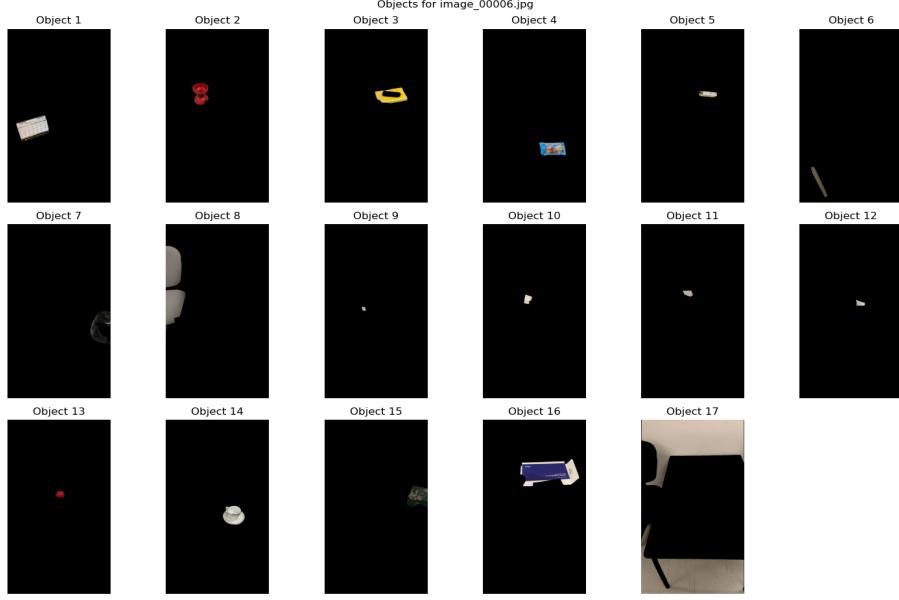


図 4.1: 手動でクラスタリングするのが困難なシーンの例

このようなケースは、個々にクラスタとして振り分けると、正解データとしての信頼性が低下するだけでなく、3D オブジェクトの生成においても有用な情報を提供しないと判断した。したがって、これらの問題が生じたオブジェクトについては、あえて一つのクラスタにまとめる方針を採用した。

この手法により、手動による正解クラスタの振り分けにおいて、オブジェクトの重なりや見切れによる不確実性を低減し、3D オブジェクト生成に有用なクラスタリング結果が得られるよう工夫している。

4.2 評価方法

4.2.1 各手法のパラメータ

(i) 類似度・閾値ベースクラスタリング 類似度閾値の最適化

このクラスタリング方法では、初期クラスタを作るための閾値とクラスタ同士を結合するために必要な閾値の 2 つのパラメータが存在する。この 2 つは本研究ではテストデータに対して $[0.85, 1.00]$ まで 0.01 ずつで総当たり探索を行い、Hold-out 検証 [1] で 1 番優秀な閾値を出力している。Hold-output 検証に用いた学習データは 6 つの学習データを用いて学習を行っている。 τ_1 は各画像が既存クラスタに追加されるか否かの判断に用いられ、 τ_2 は各クラスタのメドイドの類似度に基づくクラスタ統合に用いられる。ここでどんな多視点画像にも有用な閾値を求めなければならない。本研究では手動でクラスタリングを行った答えとなる正解クラスタと本アルゴリズムの結果である予測クラスタのペアであるデータセットを用いて、以下のステップにより各候補閾値ペア (τ_1, τ_2) を総当たりで施行し、クラスタリング正答率（平均類似度）が最大となる閾値を選定する指定された範囲（0.85 から 1.00）と刻み幅（0.01）に基づき、 τ_1 と τ_2 の候補値の集合

$$\{0.85, 0.86, \dots, 1.00\}$$

をあらかじめ生成する。次に、適した正解クラスタと予測クラスタのペアを求めるために、ハンガリアンアルゴリズム [21] を用いて最適なラベル対応を求め、共通する画像に対して正しくクラスタリングされた割合（正答率）を算出する。閾値ごとに各候補ペアごとに得られた正答率を記録し、最も高い正答率を与えた閾値ペア (τ_1^*, τ_2^*) を選定する。そしていくつかの多視点画像でこれを行い、平均的に最も良い予測をもたらす閾値ペア (τ_1^*, τ_2^*) を選定する。

- (ii) **K-Means クラスタリング 最適クラスタ数の決定:** このクラスタリング方法では、クラスタの数である K を決めなければならない。本研究では、シルエットスコア [1] を用いて自動的にクラスタ数である K を決めている。K-Means クラスタ数を事前に決定する必要がある。本研究では複数の候補クラスタ数に対してシルエットスコア [20] で評価指標を用い、最適なクラスタ数を自動的に決定する。候補となるクラスタ数は $k = 2$ から $k = 30$ の範囲とし、各 k に対して K-means を実行して得られたクラスタリング結果に基づき、シルエットスコアを計算した。
- (iii) **HDBSCAN クラスタリング パラメータ最適化:** HDBSCAN は、密度に基づくクラスタリング手法として、データの密度の不均一性に対してロバストであるが、クラスタの品質に大きく影響するパラメータとして、クラスタとして認識されるために必要な最小のサンプル数を指定するパラメータと各データ点の局所密度を評価するためのパラメータが存在する。本研究では、これらのパラメータ候補として、 $\{5, 10, 15, \dots, 50\}$ および $\{5, 10, 15, 20\}$ を用い、グリッドサーチによって各候補ペアで HDBSCAN を実行し、クラスタリング結果からノイズ（ラベル -1）を除いたデータに対してシルエットスコアを計算する。その際、シルエットスコアが最大となるパラメータ組み合わせである、クラスタとして認識されるために必要な最小のサンプル数を指定するパラメータと各データ点の局所密度を評価するためのパラメータを最適パラメータとして採用する。

4.3 結果

本章では、実際にクラスタリングを行った結果を用いて 3D オブジェクトを生成する。多視点画像としては、図 4.2、図 4.3 に示すような 2 組を使用しており、図 4.2、図 4.3 は多視点画像の一部を示している。これは少数のオブジェクトが含まれるケースと、それなりに多くのオブジェクトが含まれるケースを比較し、提案手法の有効性を確認するためである。それぞれクラスタリングに必要なパラメータとその結果を、各手法ごとに以下で示す。なお、クラスタリングの正答率が高くても、実際には本来別のオブジェクトが混在している可能性があると、3D オブジェクトを生成する際に影響が生じる。そこで、本研究では、実際にクラスタリングされたクラスタを確認し、その一部を抜粋して本論文に掲載した。

4.3.1 手法 1: 類似度・閾値ベースクラスタリングの結果

ここでは類似度・閾値ベースクラスタリングについての結果を論じる本節では、前章で述べた手法に基づく類似度・閾値ベースクラスタリングの結果について議論する。本手法は、クラスタ統合の際

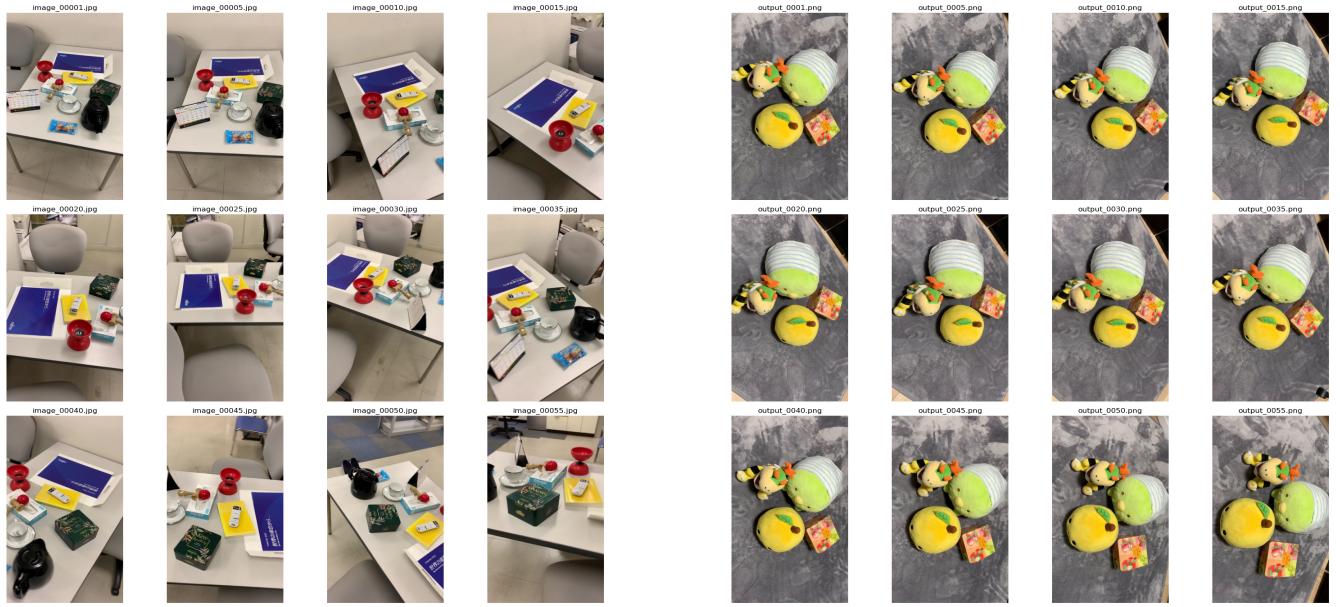


図 4.2: 多視点画像の一部 (デスク)

図 4.3: 多視点画像の一部 (ぬいぐるみ)

に用いる 2 種類の閾値、すなわち τ_1 (各画像の追加判定に用いる閾値) と τ_2 (クラスタ統合時の代表特徴量間の類似度の閾値) を必要とする.

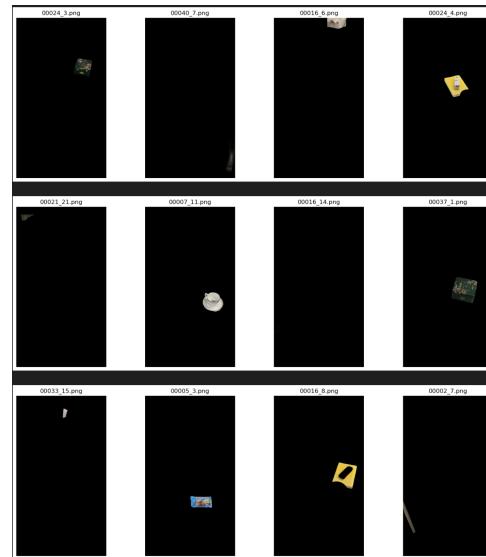
– **結果の定量評価:**

閾値の最適化の結果、 $\tau_1 = 0.98$ および $\tau_2 = 0.94$ と判定され図 4.4 では 57.4% であったのに対し、図 4.5 では 95.3% であった.

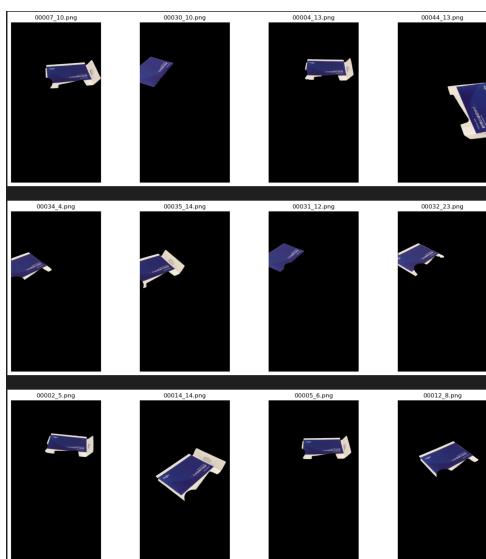
– **結果の図示:**



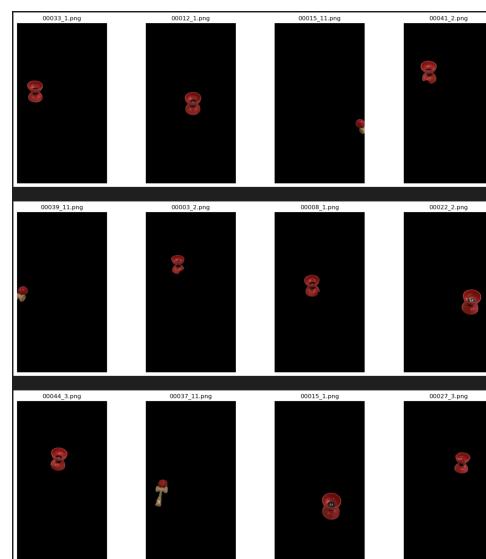
(A): ベースクラスタリングの結果 1/4



(B): ベースクラスタリングの結果 2/4

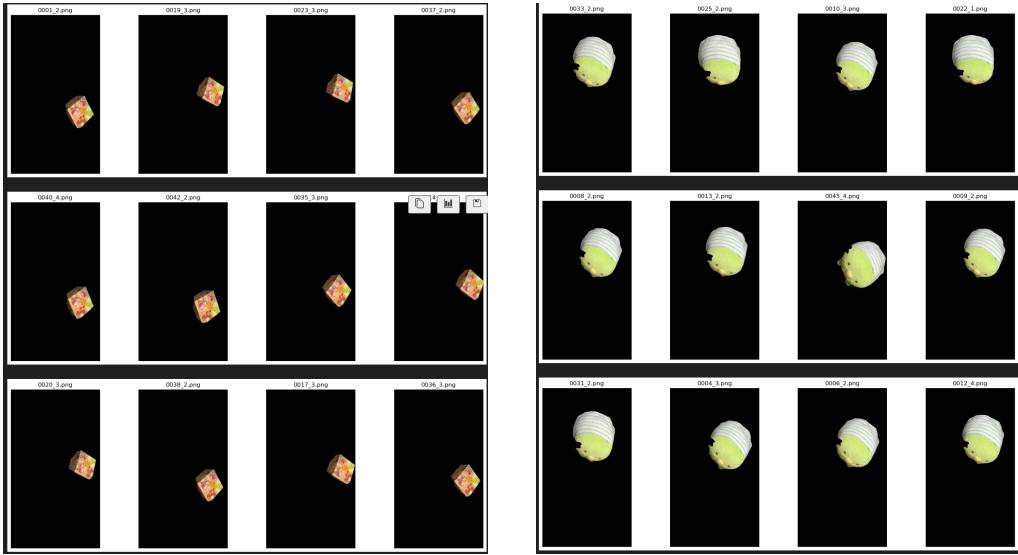


(C): ベースクラスタリングの結果 3/4

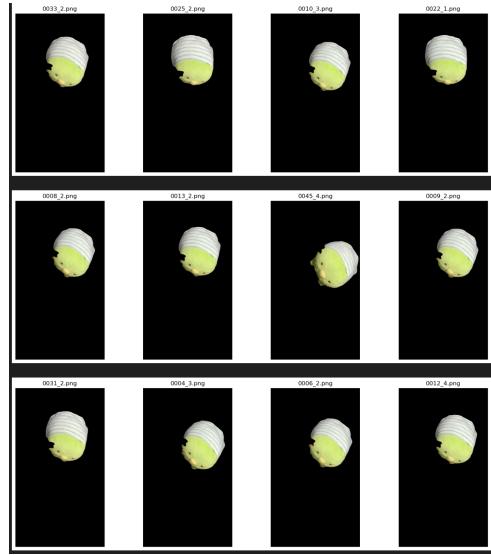


(D): ベースクラスタリングの結果 4/4

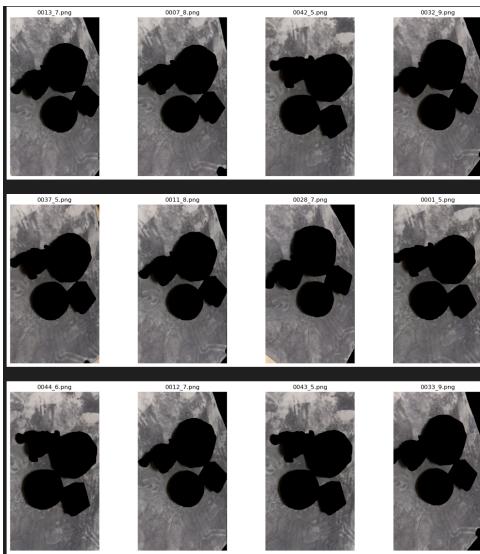
図 4.4: 多視点画像 (デスク) をベースクラスタリングした結果の一部



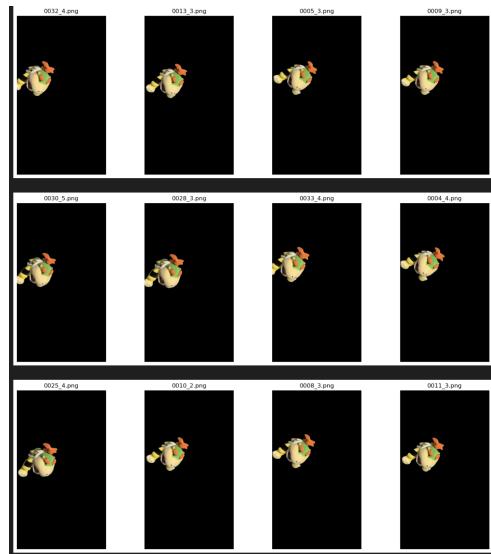
(A): ベースクラスタリングの結果 1/4



(B): ベースクラスタリングの結果 2/4



(C): ベースクラスタリングの結果 3/4



(D): ベースクラスタリングの結果 4/4

図 4.5: 多視点画像 (ねいぐるみ) をベースクラスタリングした結果の一部

– 図示された結果から得られる考察 図 4.4 の図 (A) と図 (C) に示すように、一部のクラスタは適切に分割されており、クラスタリング結果は良好と言える。しかし、図 (B) と図 (D) では複数のオブジェクトが単一のクラスタとして扱われてしまっていることが確認できる。

cosine 類似度を用いたクラスタリングがどの閾値でも成立しないのかを検証するために、個別に閾値を設定し、出力結果を確認しながら閾値の調整を行った。その結果、図 4.6 および図 4.7 に示すようにクラスタリングが成立していることが確認された。よって、単純に最適化の過程で求めた閾値が不適切であったことが原因であり、cosine 類似度を用いたクラスタリングそのものは十分に有効であると考えられる。

一方、図 4.9 で示す結果では、比較的少数のオブジェクト数を扱う場合、クラスタリングが適切に機能していることが確認できる。すなわち、オブジェクト数がこの程度であれば、従来のパラメータ設定でも十分に妥当なクラスタリングを得られることがわかった。

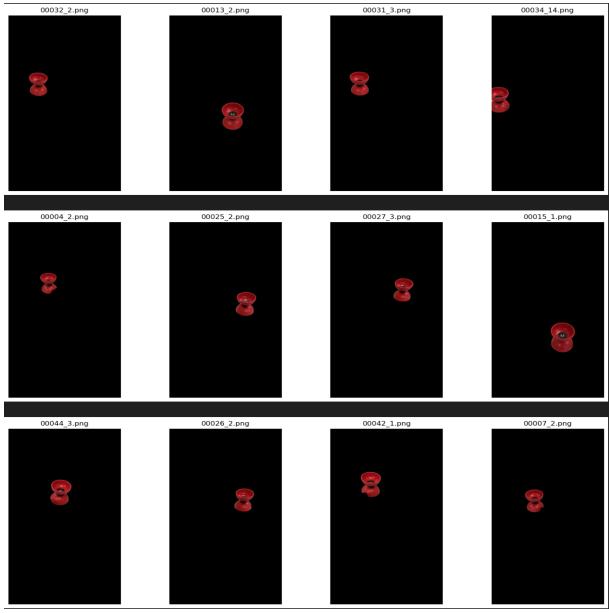


図 4.6: 個別に閾値を設定した時の結果

1/2

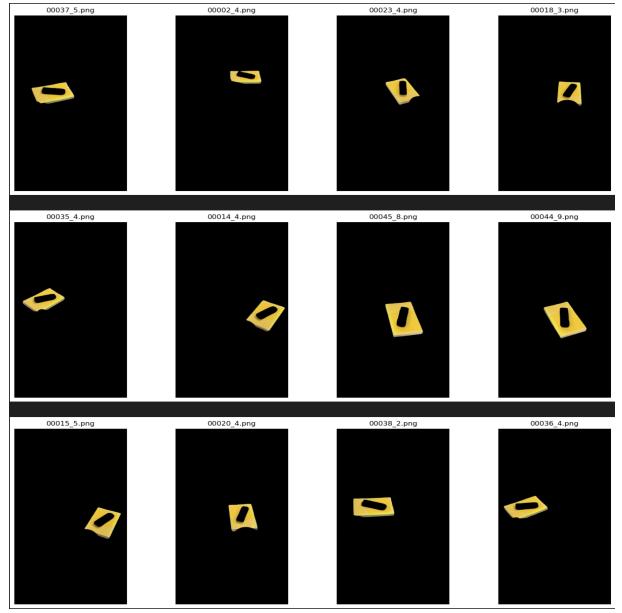


図 4.7: 個別に閾値を設定した時の結果

2/2

4.3.2 手法 2: K-means ベースクラスタリングの結果

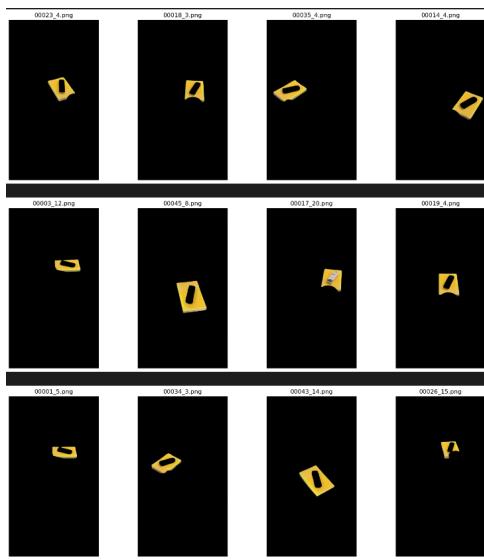
本節では、各画像の特徴量に対して K-means ベースクラスタリングの結果を紹介する。K-means ベースクラスタリングの最適なクラスタ数を決定するために、シルエットスコアを用いた検証を行った。候補となるクラスタ数は $k = 2$ から $k = 30$ の範囲とし、各 k に対して K-means を実行して得られたクラスタリング結果に基づき、シルエットスコアを計算した。各試行におけるシルエットスコアの平均値を算出した結果、以下のようにになった。

- 結果の定量評価:

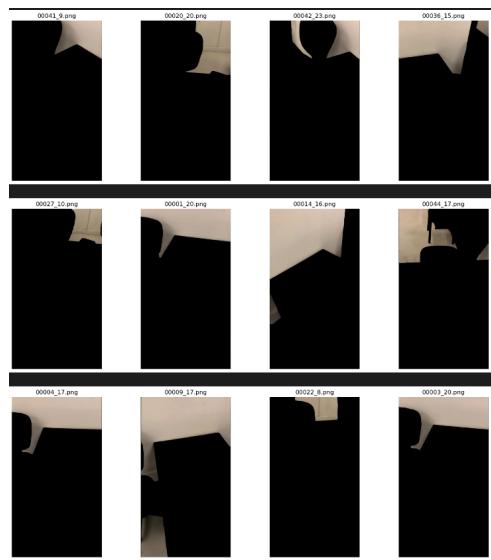
図 4.8 に示す結果では、PCA による次元削除の設定を適用し、累積寄与率 95% を保持するために 135 個の主成分が選択された。その結果、最適なクラスタ数は $k^* = 23$ (シルエットスコア = 0.3195) と判定され、正答率は 65.2% であった。

図 4.9 に示す結果では、PCA による次元削減の設定を適用し、累積寄与率 95% を保持するために 38 個の主成分が選択された。その結果、最適なクラスタ数は $k^* = 8$ (シルエットスコア = 0.7202) と判定され、正答率は 96.7% が得られた。

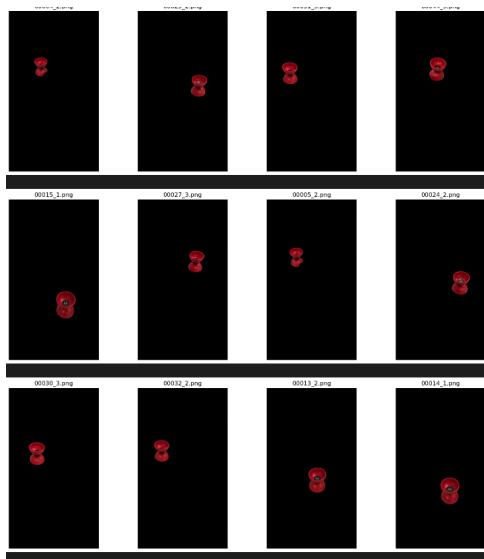
- 結果の図示: クラスタリングした結果を幾つか表示する。



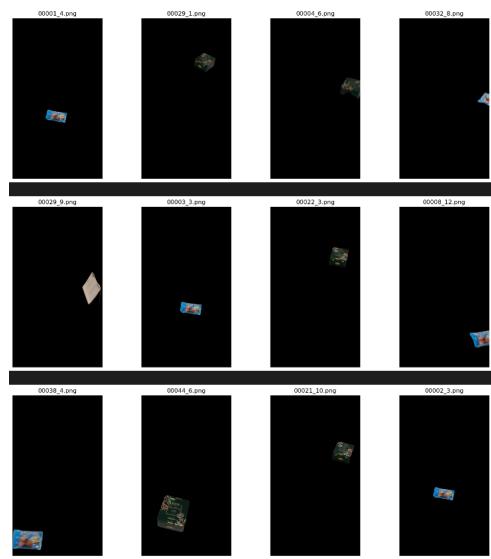
(A): K-means ベースを行った結果の一部
1/4



(B): K-means ベースを行った結果の一部
2/4

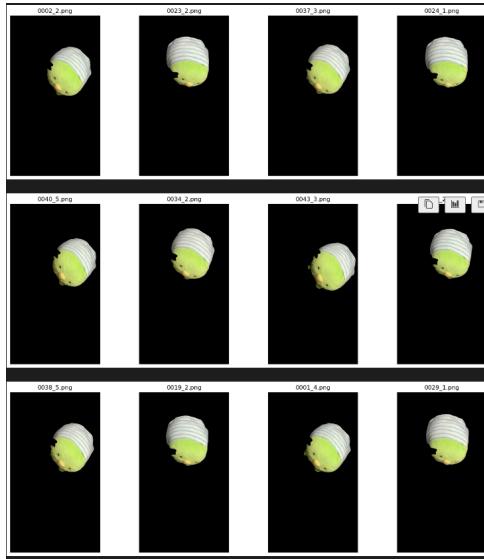


(C): K-means ベースを行った結果の一部
3/4

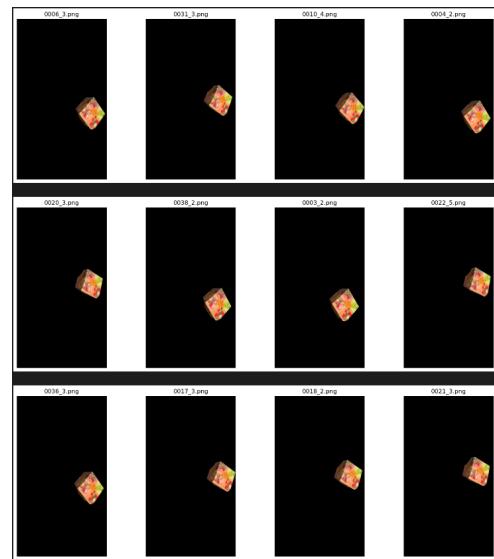


(D): K-means ベースを行った結果の一部
4/4

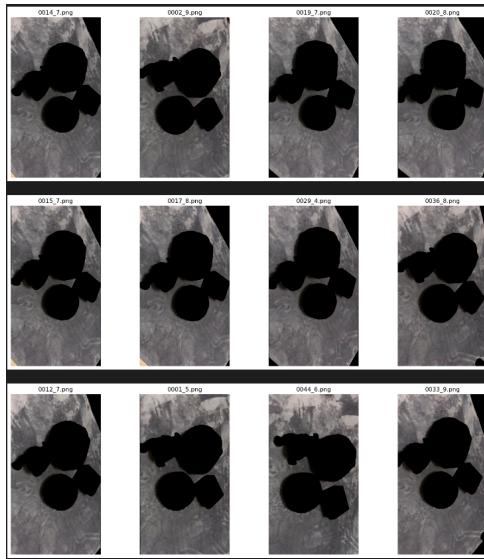
図 4.8: 多視点画像 (デスク) を K-means ベース クラスタリングした結果の一部



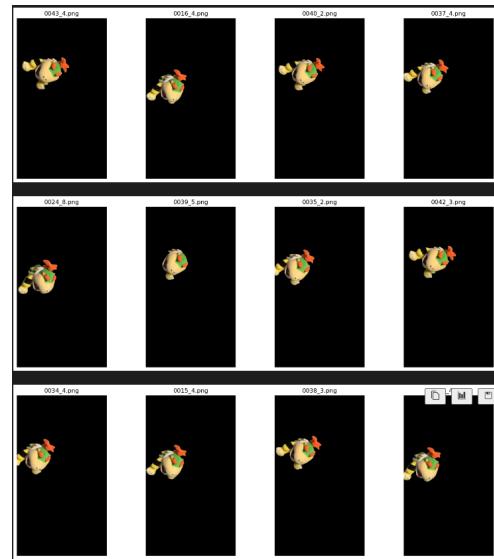
(A): K-means ベースを行った結果の一部
1/4



(B): K-means ベースを行った結果の一部
2/4



(C): K-means ベースを行った結果の一部
3/4



(D): K-means ベースを行った結果の一部
4/4

図 4.9: 多視点画像 (ぬいぐるみ) を K-means ベースクラスタリングした結果の一部

- 図示された結果から得られる考察

図 4.8 の図 (A) から図 (C) に示すように、一部のクラスタは適切に分割されており、クラスタリング結果は良好といえる。しかし、図 (D) では複数のオブジェクトが単一のクラスタとして扱われてしまっていることが確認できる。

この問題を検証するため、本研究で得られた最適解に対して、仮にクラスタ数 K を「最適解 + 10」に設定して再度クラスタリングを実施した。その結果、図 4.10 および図 4.11 に示すように、単純な K の最適化では十分に対応できない可能性が示唆された。

一方、図 4.9 で示す結果では、比較的少数のオブジェクト数を扱う場合、クラスタリングが適切

に機能していることが確認できる。すなわち、オブジェクト数がこの程度であれば、従来のパラメータ設定でも十分に妥当なクラスタリングを得られることがわかった。

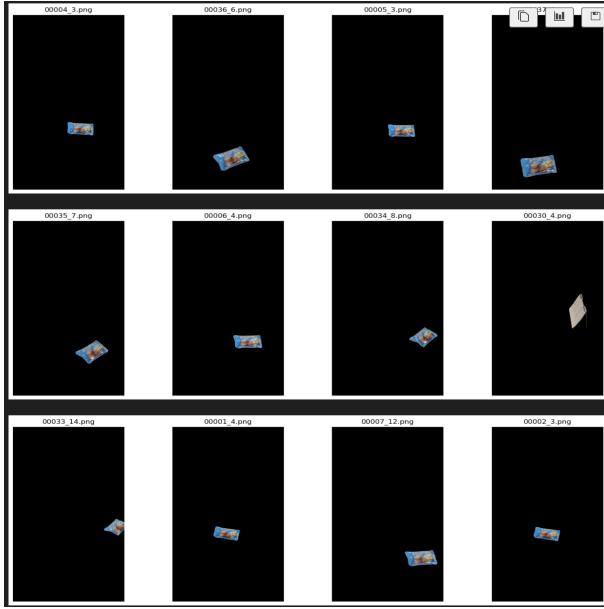


図 4.10: k の数を +10 した時の結果 1/2

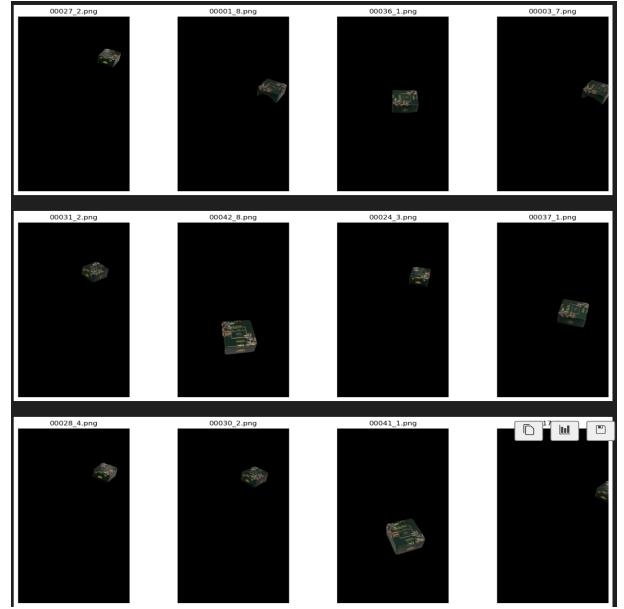


図 4.11: k の数を +10 した時の結果 2/2

4.3.3 手法 3: HDBSCAN クラスタリングの結果

本手法では、HDBSCAN を用いて密度に基づくクラスタリングを実施する。密度に基づくクラスタリング手法である HDBSCAN では、クラスタの質および分解能に大きく影響する主要パラメータとして、クラスタとして認識されるために必要な最小のサンプル数を指定するパラメータである `min_cluster_size` と 各データ点の局所密度を評価するためのパラメータを最適パラメータ `min_samples` が存在する。本研究では、これらのパラメータ候補をそれぞれ

$$\text{min_cluster_size} \in \{5, 10, 15, \dots, 50\}, \quad \text{min_samples} \in \{5, 10, 15, 20\}$$

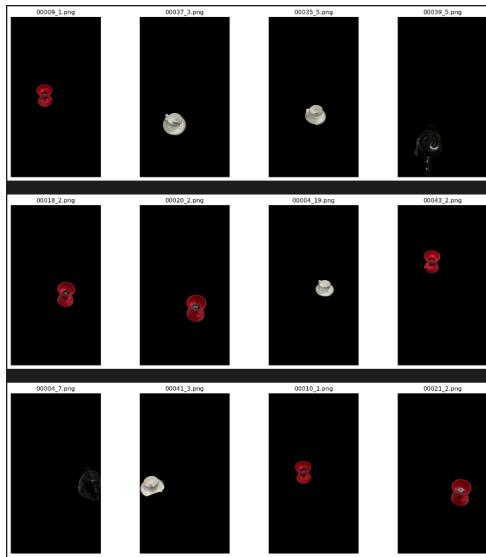
と設定し、グリッドサーチによって最適な組み合わせを自動選定する手法を採用した。

- 結果の定量評価:

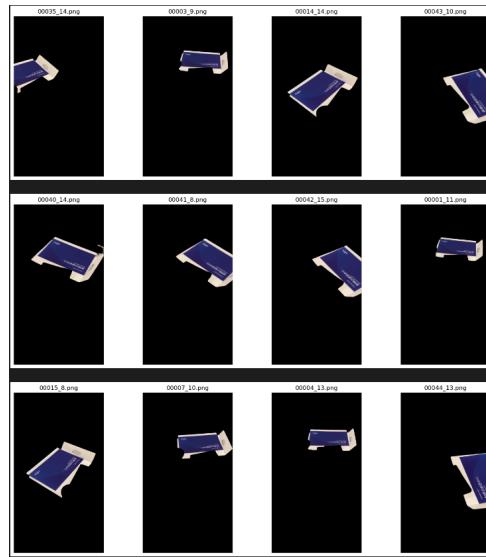
図 4.2 では、最適な PCA 主成分数が 6 個となり、HDBSCAN パラメータが `min_cluster_size` = 10, `min_samples` = 15 であり（シルエットスコア 0.5198）, 正答率は 42.6%

図 4.3 では、最適な PCA 主成分数が 3 個となり、HDBSCAN パラメータが `min_cluster_size` = 45, `min_samples` = 20 であり（シルエットスコア 0.8039）, 正答率は 87.5%

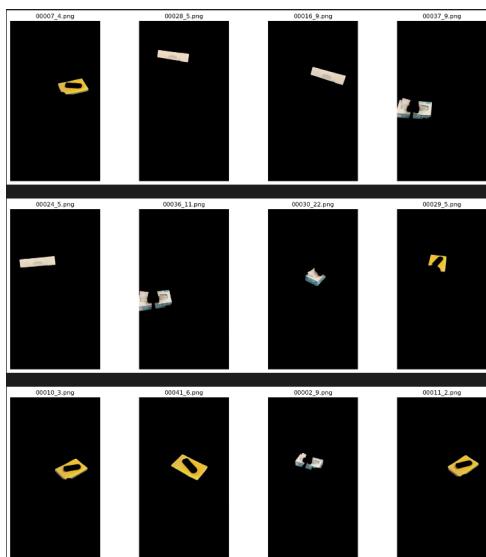
- 結果の図示: クラスタリングした結果を幾つか表示する。



(A): HDBSCAN クラスタリング 1/4



(B): HDBSCAN を行った結果の一部 2/4

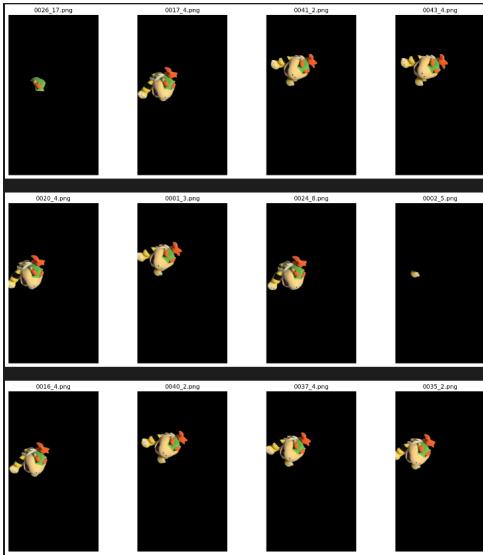


(C): HDBSCAN を行った結果の一部 3/4

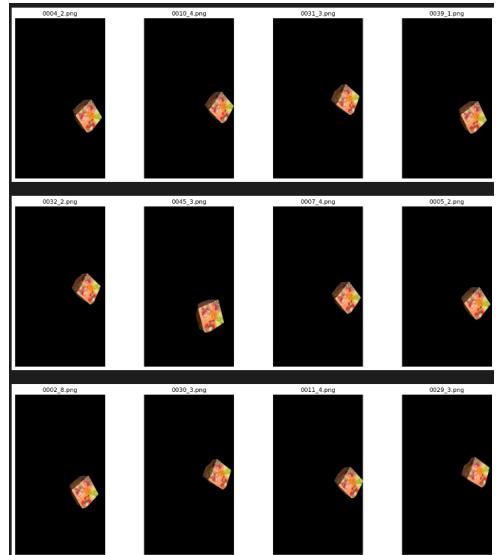


(D): HDBSCAN を行った結果の一部 4/4

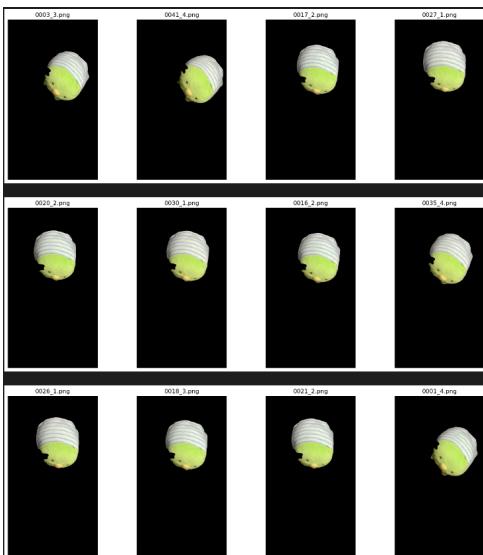
図 4.12: 多視点画像 (デスク) を HDBSCAN クラスタリングした結果の一部



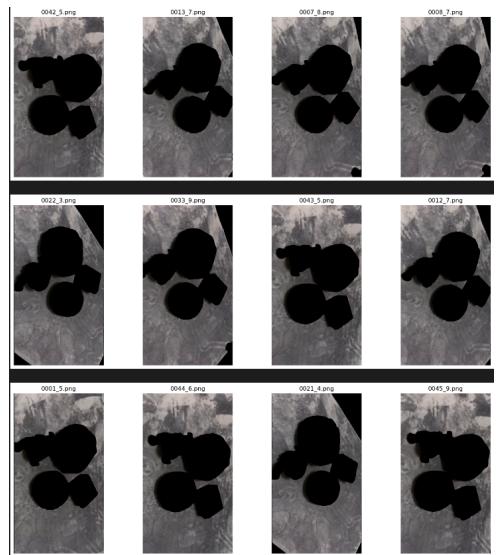
(A): HDBSCANを行った結果の一部 1/4



(B): HDBSCANを行った結果の一部 2/4



(C): HDBSCANを行った結果の一部 3/4



(D): HDBSCANを行った結果の一部 4/4

図 4.13: 多視点画像(ぬいぐるみ)を HDBSCAN クラスタリングした結果の一部

- 図示された結果から得られる考察

図 4.13 に示す結果では、比較的少数のオブジェクト数を扱う場合、クラスタリングが適切に機能していることが確認できる。また、図 (A) に示すように、類似度閾値ベース,k-means ベースとは異なり、別オブジェクトと見なせる領域が同一クラスタに含まれている例も観察される。その要因としては、SAM による領域分類の際、同一オブジェクトであっても他の画像の切り分け方とは異なり、より細分化されたセグメントが生成されるケースが挙げられる。k-means ベースでは、こうした細分化されたオブジェクトが別のオブジェクトと見なされ、別のクラスタに振り分けられる。一方、HDBSCAN を用いたクラスタリングでは、こうした細分化されたセグメントも同一クラスタとして取り込むため、図のように本来は異なるオブジェクトが一括りにされる状況が生じている。このような分割パターンは、他の手法では見られない特徴的な

現象であると考えられる。

一方、図 4.12 では図 (B), 図 (D) は上手くクラスタリングが行われているが、図 (A), 図 (C) を見ると複数のオブジェクトが 1 つのクラスタとして、クラスタリングされている。ここで k-means と同じように幾つかのパラメータを増やしても、クラスタの数が 1,2 個変わるだけであり、?? 程度のオブジェクトをクラスタするには至らなかった

4.3.4 3 手法のクラスタリングの結果

3 つのクラスタリングを行った結果 K-means ベースでクラスタリングを行った結果が最も良い正答率となった。よって本研究で 3D オブジェクトを生成するときに K-means でクラスタリングした結果を使用して 3D オブジェクトの出力を行う。

4.4 3D オブジェクトの生成

本節ではクラスタリングを行った多視点画像に対して実際に、3DGS[5] を用いて 3D オブジェクトの生成を行う。3DGS で生成した結果の一部を以下に示す。



図 4.14: 3DGS で生成した結果 (ぬいぐるみ)1/2



図 4.15: 3DGS で生成した結果 2(ぬいぐるみ)2/2



図 4.16: 3DGS で生成した結果 2(ぬいぐるみ)1/2



図 4.17: 3DGS で生成した結果 2(ぬいぐるみ)2/2



図 4.18: 3DGS で生成した結果 (デスク)1/2



図 4.19: 3DGS で生成した結果 (デスク)2/2

クラスタリングされた画像の背景を黒に設定していたため、背景と一体化して出力されているが、図 4.3においては、図 4.14、図 4.15、図 4.18、図 4.19 のように領域分類されてオブジェクトが出力されていることを示した。しかし、図 4.2では、クラスタリングされた多視点画像を 3DGS にかけた結果、オブジェクトが上手く出力図、出力されたとしても図 4.18、図 4.19 のような結果となった。

第 5 章

考察と課題

5.1 考察と今後の課題

本研究では、図 4.14 および図 4.15 に示すように、テキスト情報を介さず領域分類を実施し、その結果を 3D オブジェクトとして取得することに成功した。しかし、閾値やクラスタ数 K などのパラメータ調整を依然として手動で行わなければならない。

図 4.2 および図 4.3 における多視点画像のクラスタリング結果では、特に図 4.2 の例で 3D オブジェクトの出力が不十分であった、これは 3D オブジェクト生成に必要なカメラ位置情報が十分に取得できなかったことが原因と考えられる。

3DGS では、3D オブジェクト出力前に colmap[14] の Structure from Motion (SfM)[10] といった技術を用い多視点画像からカメラ位置を推定する必要があるが、本来は画像全体の特徴量に基づいて推定すべきカメラ位置が、本研究ではオブジェクトごとに分割された画像では十分な情報が得られず、推定精度に課題があると予測される。従って、多視点画像を分割して分類する前にカメラ位置を取得し、分割した画像とカメラ位置を適切に紐付ける手法の導入が望まれる。

また、3DGS では背景の黒も含めて出力されるため、目的のオブジェクトのみを抽出して 3D オブジェクトを生成する必要がある。さらに、図 4.13 に示すように、カメラ位置から見てオブジェクト同士が重なって出力された場合、図 4.18 や図 4.19 に示すように穴が生じるため、補間処理の実装も検討すべき課題である。

謝辞

本論文の研究と執筆にあたりその細部に至るまで終始懇切なる御指導と御鞭撻を賜りました、高知大学理工学部情報科学科 教授陣に謹んで深謝の意を申し上げます。

また、研究室において常に熱心な御討論を頂きました、OB・学生の方々に感謝の意を表します。

参考文献

- [1] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. 1st ed. Springer, 2007. ISBN: 0387310738. URL: <http://www.amazon.com/Pattern-Recognition-Learning-Information-Statistics/dp/0387310738%2FSubscriptionId%3D13CT5CVB80YFWJEPWS02%26tag%3Dws%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3D0387310738>.
- [2] Jiazhong Cen et al. *Segment Any 3D Gaussians*. 2025. arXiv: 2312.00860 [cs.CV]. URL: <https://arxiv.org/abs/2312.00860>.
- [3] Antoine Guédon and Vincent Lepetit. “SuGaR: Surface-Aligned Gaussian Splatting for Efficient 3D Mesh Reconstruction and High-Quality Mesh Rendering”. In: *CVPR* (2024).
- [4] Kaiming He et al. *Deep Residual Learning for Image Recognition*. 2015. arXiv: 1512.03385 [cs.CV]. URL: <https://arxiv.org/abs/1512.03385>.
- [5] Bernhard Kerbl et al. *3D Gaussian Splatting for Real-Time Radiance Field Rendering*. 2023. arXiv: 2308.04079 [cs.GR]. URL: <https://arxiv.org/abs/2308.04079>.
- [6] Alexander Kirillov et al. *Segment Anything*. 2023. arXiv: 2304.02643 [cs.CV]. URL: <https://arxiv.org/abs/2304.02643>.
- [7] Hao Li et al. *LangSurf: Language-Embedded Surface Gaussians for 3D Scene Understanding*. 2024. arXiv: 2412.17635 [cs.CV]. URL: <https://arxiv.org/abs/2412.17635>.
- [8] Yun-Jin Li et al. *SADG: Segment Any Dynamic Gaussian Without Object Trackers*. 2024. arXiv: 2411.19290 [cs.CV]. URL: <https://arxiv.org/abs/2411.19290>.
- [9] Ben Mildenhall et al. *NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis*. 2020. arXiv: 2003.08934 [cs.CV]. URL: <https://arxiv.org/abs/2003.08934>.
- [10] Onur Ozyesil et al. *A Survey of Structure from Motion*. 2017. arXiv: 1701.08493 [cs.CV]. URL: <https://arxiv.org/abs/1701.08493>.
- [11] Songyou Peng et al. *Shape As Points: A Differentiable Poisson Solver*. 2021. arXiv: 2106.03452 [cs.CV]. URL: <https://arxiv.org/abs/2106.03452>.
- [12] Ben Poole et al. *DreamFusion: Text-to-3D using 2D Diffusion*. 2022. arXiv: 2209.14988 [cs.CV]. URL: <https://arxiv.org/abs/2209.14988>.
- [13] Alec Radford et al. *Learning Transferable Visual Models From Natural Language Supervision*. 2021. arXiv: 2103.00020 [cs.CV]. URL: <https://arxiv.org/abs/2103.00020>.
- [14] Johannes Lutz Schönberger and Jan-Michael Frahm. “Structure-from-Motion Revisited”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

- [15] Johannes Lutz Schönberger et al. “Pixelwise View Selection for Unstructured Multi-View Stereo”. In: *European Conference on Computer Vision (ECCV)*. 2016.
- [16] H.C. THOMAS et al. 世界標準 MIT 教科書 | アルゴリズムイントロダクション 第 3 版 総合版. 世界標準 MIT 教科書. 近代科学社, 2013. ISBN: 9784764904088. URL: <https://books.google.co.jp/books?id=NgCoDwAAQBAJ>.
- [17] Sudheendra Vijayanarasimhan et al. *SfM-Net: Learning of Structure and Motion from Video*. 2017. arXiv: 1704.07804 [cs.CV]. URL: <https://arxiv.org/abs/1704.07804>.
- [18] Wenbo Zhang et al. *Bootstrapping Clustering of Gaussians for View-consistent 3D Scene Understanding*. 2024. arXiv: 2411.19551 [cs.CV]. URL: <https://arxiv.org/abs/2411.19551>.
- [19] M. Zwicker et al. “Surface Splatting”. In: *ACM Transactions on Graphics (Proc. ACM SIGGRAPH)*. July 2001, pp. 371–378.
- [20] 真太朗 福島. *Python 機械学習プログラミング : 達人データサイエンティストによる理論と実践*. Python machine learning : unlock deeper insights into machine learning with this vital guide to cutting-edge predictive analytics. 東京, Japan: インプレス, 2016 2016.
- [21] 宏和 穴井, 努 斎藤, and 講談社サイエンティフィク. *今日から使える!組合せ最適化 : 離散問題ガイドブック*. 講談社, 2015. URL: <https://ci.nii.ac.jp/ncid/BB18979319>.