花弁配置推定システムのための合成画像を用いた分類手法の評価

信田 浩希[1,a)]　内海 ゆづ子[1,b)]　藤本 仰一[2,c)]　岩村 雅一[1,d)]

概要：花弁配置は, 花弁の枚数とその位置, それらの前後関係を表すものであり, 種を超えた原理に基づくと考えられている. 数理モデルにより, 花弁の発生過程のシミュレーションが行われており, 数理モデルの考案や検証のために, 画像から花弁配置を自動推定するシステムが求められている. 従来手法では花弁の重なり位置の検出と前後関係の推定を別々に行い, 推定精度が 0.275 と低かった. 本研究では, Conditional DETR を用いて両者を統合的に処理し, 精度向上を試みた. 合成画像を用いた実験の結果, 推定精度は 0.343 となった.

1. はじめに

花は我々に身近なものであり, 植物の発生や進化の過程を解明するために様々な研究が行われている. 花の発生メカニズムを考察する際に手掛かりとなるものの一つに花弁配置がある. 花弁配置とは, 花弁の枚数とその位置, それらの前後関係を表すものである. 前後関係は**図 1** (左) に示すように, 注目している花弁に対して, 両隣の花弁が上にある外側, 両隣の花弁が下側にある内側, 一方の花弁が上側にありもう一方の花弁が下側にある交互の 3 種類がある. 図 1 (左) の花弁配置は図 1 (右) に示すようなダイアグラムで表現される. ダイアグラムの色は 3 色あり, 赤色は外側, 緑色は交互, 青色は内側に対応する.

花弁配置は, 種が異なっていても同じになったり, 同じ種でも異なったりすることがある. また, 同じ種で異なる複数の花弁配置を取る花が存在する [1]. つまり, 花弁配置と種は 1 対 1 で対応しない. このことから, 花弁配置の決定には種を超えた共通の原理が存在すると予測されている. この原理を解明すべく, 数理モデルに従って花弁の発生過程をシミュレーションする研究が行われている [2].

数理モデルの考案や検証を行うには, 花を実際に観察し, その花弁配置をモデルと照らし合わせる必要がある. 花弁配置を推測する作業は人手で行われており, 大変な労力を要する. そのため, 花弁配置を自動推定するシステムが求められている. 中谷らはパターン認識の技術を用いて花画像から花弁配置を自動で推定する手法を提案した [3]. しかし, 中谷らの手法の花弁配置の推定精度は 0.275 となっており, 実用に十分ではない. 中谷らの手法の全体像を**図 2** に示す. 中谷らの手法では, 画像から花弁配置を推定するために, 画像から花弁同士の重なり位置を検出し, その位置の前後関係を元に花弁配置を推定した. 花弁同士の重なり位置の検出に失敗すると前後関係の推定も失敗する. このように, どこかの処理の失敗がそのまま他の処理に伝播することで, 全体の精度が低下したと考えられる.

これまでの研究で, 複数タスクを一つのタスクとしてまとめ, まとめたタスクを 1 つのニューラルネットワークを用いて解くと, 精度が向上することが報告されている [4], [5], [6]. このうち, [4] は画像による文字認識を行った研究である. 従来の文字認識は, 画像から認識対象となる文字領域を特定するフィールド抽出, 連続した文字を個別の文字や単語に分割する文字セグメンテーション, 文脈から単語や文の構造を考慮した言語的制約を課す言語モデリングを含む複数のモジュールで構成されていた. [4] では, これら複数のタスクを 1 つの CNN ベースの手法にまとめることで全体の精度が向上することを示した. 画像認識対象が異なった場合でも, タスクを 1 つにまとめることで認識の精度向上が期待される.

そこで, 本研究では, 花弁同士の重なり位置の検出と前後関係の推定を 1 つのニューラルネットワークでまとめて実施することで, 花弁配置の推定精度の向上を目指す. 花

[1] 大阪公立大学大学院情報学研究科
Graduate School of Informatics, Osaka Metropolitan University, 1-1, Gakuencho, Naka, Sakai, Osaka 599-8531, Japan
[2] 広島大学大学院統合生命科学研究科
Graduate School of Integrated Sciences for Life, Hiroshima University, 1-3-2, Kagamiyama, Higashi-Hiroshima, Hiroshima 739-8524, Japan
[a)] sp25239d@st.omu.ac.jp
[b)] yuzuko@omu.ac.jp
[c)] kfjmt@hiroshima-u.ac.jp
[d)] masa.i@omu.ac.jp

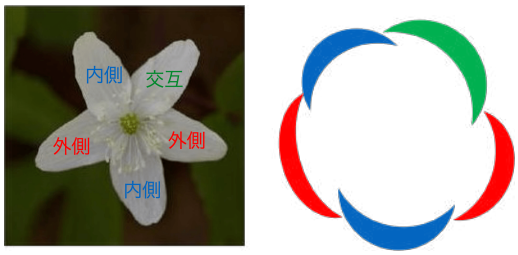


図 1: 花画像に前後関係を表示した図 (左) と花弁配置を表した図 (右)

図 2: 中谷らの手法の流れ.

弁の検出とその前後関係を知るには, 注目する花弁の位置を把握するだけでなく, 隣接する花弁との関係を考慮する必要がある. そのため, 局所的な特徴だけでなく, 離れた位置にある花弁の情報も扱う必要がある. そこで, 本研究では, Self-attention により離れた部位の関係を扱うことが可能な Transformer ベースのセグメンテーション手法である DETR を用いる.

実験では, DETR に大量の学習データが必要であること, また, 学習データを大量に用意することが困難であることから, 合成画像を用いてモデルの評価を行った. 合成画像は, 実画像から花弁や中心部位を切り取り, これまでの研究で明らかになっている花弁配置の規則 [1] に従って花弁を配置することで生成した. 実験の結果, 花弁配置の推定精度は 0.343 となった.

2. 関連研究

花弁配置の推定の関連研究として, 花画像を対象とした画像処理の例を紹介する. また, 本研究では花弁の重なりの順序を扱うことから, 順序の推定に関する研究についても紹介する.

2.1 花画像を対象とした画像処理

花は我々に身近な存在であり, 観賞用や農業など多岐にわたる分野で注目されている. そのため, 様々なアプローチで花画像を対象とした画像処理の研究が行われてきた. 花の種を識別するための手法として, テクスチャや色空間を特徴量として用いる手法 [7], [8] や, CNN を用いた手法 [9] が提案されている. これらの研究は, 花の種の分類を主な目的としており, 本研究で取り扱う花弁の詳細な配置は推定していない.

農業の分野では, 受粉作業の自動化や農作物の収量の予測のために圃場を撮影した画像から栽培している植物の花を検出する研究が行われている. 温室などの室内環境では, トマトの摘花作業の自動化 [10] や, ラズベリーやブラックベリーの受粉作業の自動化 [11] を目的として花の検出手法が提案されている. 屋外で栽培される作物の花の検出も盛んに行われており, イチゴ [12], リンゴ [13], ブドウ [14] などの受粉, 摘花の作業の自動化や収量予測のために, 圃場を撮影した画像中から花が検出されている. これらの研究では, 位置の推定を主な目的としており, 本研究で取り扱う花弁の詳細な配置は推定していない.

中谷ら [3] はパターン認識の技術を用いて花画像から花弁配置を自動で推定する手法を提案した. 中谷らの手法では, 画像から花弁同士の重なり位置を検出し, その位置に注目して花弁の前後関係を推定することで花弁配置を推定した. しかし, 中谷らの手法の花弁配置の推定精度は 0.275 と低く, 更なる精度の向上が求められている.

2.2 順序推定

順序推定は，ある有限個の要素があるとき，それらの要素を予め定義された基準に基づいて順序を推定する問題である．コンピュータビジョンやパターン認識分野において，単眼深度推定や奥行き推定や顔画像による年齢推定に主に利用されている．

単眼深度推定は，与えられた 1 枚の画像からシーンの奥行き情報を推定する問題である．通常の単眼深度推定は，シーンからカメラまでの連続的な距離を推定する回帰問題として扱われる．しかし，回帰ベースの推定では，モデルの学習にかかる時間が大幅に増えたり，畳み込み積分で十分な解像度の特徴量が得られず，細かい部位の奥行き推定精度が悪い問題があった．そこで，奥行き推定を順序回帰問題として扱うことで，モデルの学習にかかる時間を削減し，細かい奥行き推定精度の向上を実現している [15], [16]. これらの研究は車載カメラの映像を対象にしており，物体同士の深度の差が大きいものを扱っている．そのため，花の近接画像を対象とし，接触していて奥行きの差が小さい花弁の順序を推定するのは難しい.

顔画像を用いた年齢推定にも，順序回帰を適用する手法が提案されている [17]. [17] では，年齢をバイナリのベクトルで表現し，年齢の推定問題をバイナリのベクトルを推定する問題に置き換えることで，年齢の推定を回帰問題から順序回帰問題として解く．年齢推定を順序回帰問題として扱うことで，子供の頃は顔の形状，大人になると皮膚の質感といったように，年代ごとに異なる顔の特徴変化をモデルが学習できるため，高精度な年齢推定が可能となっている．今回，我々の扱う花弁配置は数珠順列であることから，花弁配置の推定を順序回帰問題として扱うことは難しい.

2.3 End-to-end モデル

1 で紹介した画像を用いた文字認識 [4] 以外でも, 複数タスクを一つのタスクとしてまとめ, まとめたタスクを 1 つ

のニューラルネットワークを用いて解くと, 精度が向上することが報告されている [5], [6].

従来の音声認識システムは, 音声信号をテキストに変換する過程において, 音声と音素の対応関係を学習する音響モデル, 音素の並びから語彙的意味を導く発音モデル, そして文脈に適した語句の選択を担う言語モデルといった複数のモデルで構成されていた. Chan らはこれらのモデルを 1 つのニューラルネットワークでまとめることで, 従来の複数モデルからなるアプローチと比較して精度が向上することを示した [5]. この統合アプローチにより, 各モデル間のエラー伝播が軽減され, end-to-end の最適化が可能となった. Jumper らはタンパク質構造予測において, 類似配列検索, エネルギー最適化, 構造予測といった従来別々に行われていた複数のタスクを 1 つのニューラルネットワークモデルに統合することで, 従来の手法を大幅に超える精度でタンパク質構造を予測した [6]. この統合アプローチにより, 各タスク間の相互関係を学習し, より正確な予測が可能となった.

複数のタスクの統合により, 中間結果の誤差蓄積を回避し, タスク間の相互関係を直接学習することが可能となるため, 全体的なシステム性能の向上につながると考えられる. 我々の研究でも精度向上が期待できることから, これまで分割していたタスクを統合し, 花弁の前後関係を推定する end-to-end モデルを構築する.

3. 提案手法

提案手法では, まず, 一輪の花が撮影された画像から花弁の位置を特定し, それぞれの花弁の前後関係を推定する. そして, 花弁の前後関係の推定結果の並びを使用して, これまでの観察で判明している花弁配置 [1] の中で最も近い並びを持つものに分類する.

3.1 問題の定式化

図 3 に中谷ら [3] の手法と提案手法の比較を示す. 本研究では図 3(a) に示す 2 つのタスクを図 3(b) に示すように同時に解くことで, 花弁配置の推定精度の向上を目指す. 花弁の前後関係は, 図 1 で示すように, 3 種類 (外側, 交互, 内側) がある. そこで, 物体検出モデルで検出する物体のクラスを花弁の前後関係に対応した 3 クラス (外側の花弁, 交互の花弁, 内側の花弁) に分類することで, 花弁の位置の特定と前後関係の推定を同時に行う. 本稿で検出対象とする 3 つのクラスはいずれも花弁であり, テクスチャによる識別が難しい. そのため, 花弁の形状や両隣の花弁とのオクルージョンに注目して物体検出するモデルが好ましい. 物体検出手法には, Convolutional Neural Network (CNN) を用いた手法が多く発表され, 良好な精度を示してきた [18]. これに加え, 近年, Transformer を用いた手法も提案され, よりよい精度を示している [19]. Transformer ベースの手法は CNN ベースの手法と比較して, オクルージョンに強く, 物体の形状や周辺領域との関係に注目して物体を検出する [20], [21], [22]. 従って, 本研究では花弁の前後関係の推定に Transformer ベースの Conditional DETR [23] を用いる.

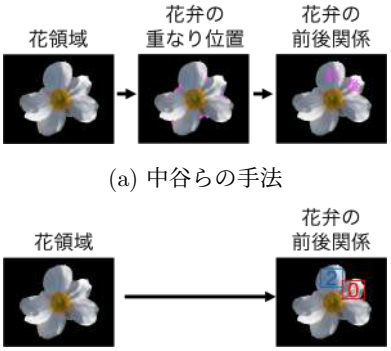


図 3: 中谷らの手法と提案手法の比較

3.2 Conditional DETR を用いた花弁の前後関係の推定

Conditional DETR は, DETR [19] を改良した物体検出モデルであり, 特に Cross-attention の機構を最適化することで学習速度の向上を実現している [23]. DETR は Transformer を活用して end-to-end で物体検出を行う. DETR は CNN ベースの手法よりも良好な物体検出精度を示す一方で, 学習の収束が遅い. これは, 物体の位置とカテゴリの双方を同時に学習しようとしている上, 位置情報推定の手がかりになる情報を与えられないためである. そこで, Conditional DETR は, 物体の位置とカテゴリを分けて学習する上, 物体の位置情報推定の手がかりとなる reference point を用いることで, 学習の収束を早める.

Conditional DETR の基本アーキテクチャは DETR と類似しており, **図 4** に示すように, backbone, encoder, decoder, feedforward network (FFN) を構成要素に持つ prediction heads から構成されている. まず, CNN ベースの backbone より入力画像から特徴マップを抽出する. この特徴マップは画像内の物体の視覚的情報を捉えたものである. 次に, encoder により特徴マップに対して Self-attention を適用し, 画像中で識別に重要な部分を抽出する処理を施す. 特徴マップの処理においては, 特徴マップに位置情報を付加する positional encoding を適用し, 空間的な位置情報を保持できるようにする. 画像内の物体を表すベクトルを物体クエリと呼び, 1 つのクエリが 1 つの物体に対応するように学習される. Decoder はこれらのクエリと encoder の出力を Cross-attention で組み合わせ, 物体の特徴を抽出する. その後, 各クエリに対して位置 (バウンディングボックス) とカテゴリ (クラスラベル) を予測する. 得られた位置とカテゴリの予測の中から, 確信度が閾値を超えたもの

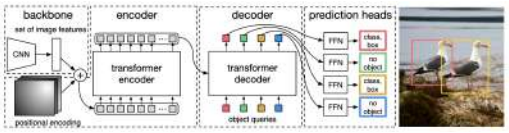


図 4: DETR のアーキテクチャ [19].

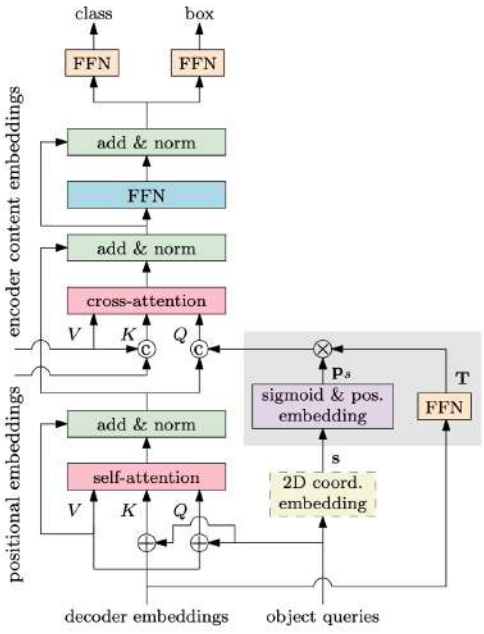


図 5: Conditional DETR のアーキテクチャ [23].

が最終的な検出結果として出力される. **図 5** に Conditional DETR の decoder のアーキテクチャを示す. Conditional DETR では参照点 (reference point) を用いて画像の特定領域に注意を集中させる空間クエリが導入されている. 従来の DETR では decoder の Cross-attention が画像全体に対して均一に適用されていたが, Conditional DETR では各物体クエリが特定の空間位置 (参照点) を持ち, その周辺領域に注意を向けることができる. これにより, モデルは画像内の特定領域に効率的に焦点を当て, 物体の局所的特徴をより正確に捉えることが可能となり，学習の収束が早くなる. 損失関数は Focal Loss [24] を用いた分類損失と, L1 Loss と Generalized Intersection over Union Loss [25] を用いたボックス回帰損失から構成される.

提案手法では, 一輪の花が撮影された画像から, 花弁の中心の位置を表す矩形と, それぞれの花弁の前後関係を表すクラスを出力するために Conditional DETR を使用する. 花弁の中心の位置を表す矩形の学習データは 20×20 pixels で固定した. これは, 学習データを作成しやすくするためである.

3.3 花弁配置の推定

推定された花弁の位置と前後関係の組を 14 種の既知の花弁配置と比較し, 編集距離が最小となる花弁配置を推定結果とする. 以降，花弁の配置と編集距離について詳細に説明する.

3.3.1 花弁の配置

花画像から推定された花弁の位置と前後関係を既知の花弁配置の候補と比較し, 花弁配置を推定する. 花弁配置は花の種ごとに異なり, 本研究で扱うイチリンソウとその近縁種は**図 6** に示す 14 種の既知の花弁配置を取りうる [1]. 花弁配置の左側にあるアルファベットと数字からなる文字は花弁配置のクラス名を表す. アルファベット A, B, C, D, E, F, G はそれぞれ花の花弁の枚数の合計が 4, 5, 6, 7, 8, 9, 10 枚であることを表し, 左の数字は同じ花弁の枚数の花の中でも異なる花弁配置を表す.

3.3.2 編集距離の推定

Conitional DERT のモデルを用いて, 花弁とその前後関係の情報が抽出される. そして, この花弁の前後関係の配置を数珠順列とみなし, これらの数珠順列が既知の花弁配置のどれにあてはまるかを推定する. 数珠順列同士の編集距離を計算し, 最も近いものを推定結果とする. **図 7** に, 花弁配置を花弁の前後関係のクラスの番号を用いた数珠順列で表現した例を示す. 数珠順列同士の編集距離は, 順列の編集距離をもとに計算される. まず, 順列の編集距離について説明する. 編集距離は, ある順列を別の順列に変換するために必要な最小の編集操作の回数を表す. 編集操作には, 要素を追加する挿入, 要素を削除する削除, 要素を別の要素に変更する置換がある. 今，2 つの順列 a, b が存在するとし, a, b の編集距離 $lev(a,b)$ を考える. $head(x,i)$ を, 順列 x の先頭から i 番目までの要素を取り出した部分順列, $tail(x,i)$ を順列 x の先頭から i 個の要素を取り除いた部分順列とし, $|x|$ を順列 x の長さを示すとすると，$lev(a,b)$ は以下のように表現できる.

$$lev(a,b) = \begin{cases} |a| & \text{if } |b| = 0, \\ |b| & \text{if } |a| = 0, \\ lev(tail(a,1), tail(b,1)) & \text{if } head(a,1) = head(b,1), \\ 1 + \min \begin{cases} lev(tail(a,1), b) \\ lev(a, tail(b,1)) \\ lev(tail(a,1), tail(b,1)) \end{cases} & \text{otherwise} \end{cases} \tag{1}$$

数珠順列同士の編集距離は，数珠順列を順列に変換した後，順列の編集距離を計算することで求める. 但し，数珠順列は始点の取り方の違いにより複数の順列に変換されるため，取り得る全ての順列について編集距離を計算し，そのうち最小のものを数珠順列同士の編集距離とする.

数珠順列 C, D の編集距離 $clev(C,D)$ を考える. C, D から, ある要素を先頭とし, その要素の一つ前を末尾として変換した順列を c, d とする. 数珠順列を順列に変換した際に取り得る全ての順列を考慮するために，順列 x の要素の順序を逆にした順列 $reverse(x)$ と順列 x の先頭を i 個ずらした順列 $rotate(x,i)$ を考える. $reverse(x)$, $rotate(x,i)$ は以下のように表される.

$$reverse(x) = \begin{cases} \emptyset & \text{if } |x| = 0, \\ reverse(tail(x, |x|-1)) + head(a,1) & \text{otherwise} \end{cases} \tag{2}$$

$$rotate(x,i) = tail(x,i) + head(x,i) \tag{3}$$

$reverse(x)$, $rotate(x,i)$ を用いて，数珠順列の編集距離 $clev(C,D)$ は, 以下のように表される.

$$clev(C,D) = \begin{cases} |c| & \text{if } |d| = 0, \\ |d| & \text{if } |c| = 0, \\ \min_{i=0}^{|c|-1} \begin{cases} lev(rotate(c,i), d) \\ lev(reverse(rotate(c,i)), d) \end{cases} \end{cases} \tag{4}$$

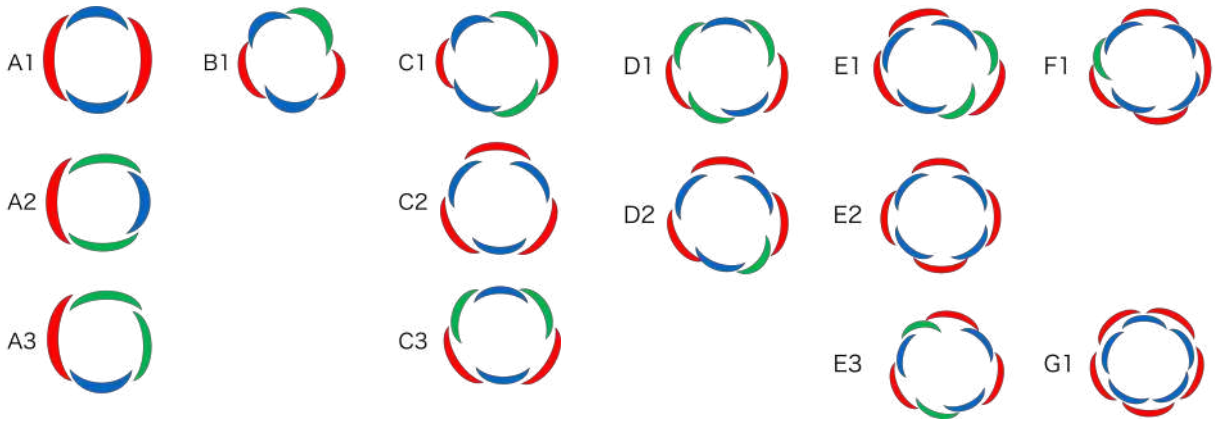


図 6: イチリンソウとその近縁種の花弁配置

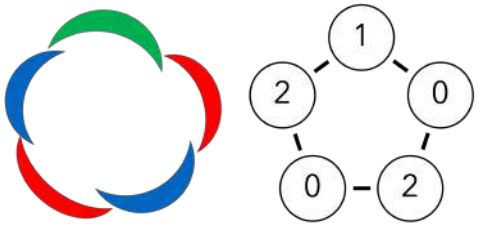


図 7: 花弁配置 (右) と対応する数珠順列 (右)

4. 実験

提案手法の有効性を確認するため, 合成画像を用いて実験を行った. まず, 花弁の前後関係の推定を行い, その精度を評価した. また, 花弁の前後関係の推定結果を入力として, 既知の花弁配置のうち最も編集距離が近いものに分類し, その精度を評価した.

4.1 実験データ

花弁の前後関係を推定するためのモデルの学習には合成画像を用いた. 本研究で用いた Conditional DETR は Transformer を用いており, Transformer ベースの手法で，性能を担保するには大量の学習データが必要である [26], [27]. このため, 本研究では, 実画像から合成画像を生成し, 合成画像を用いて学習を行った.

本稿で扱う花の花弁配置のうち，9 種 (A1, A3, B1, C2, D2, E2, E3, F1, G1) は花弁が一定の角度ずつずれながら発生することが知られており [1]，それらの花弁の枚数と花弁をずらす角度は**表 1** に示す通りである．そこで，これらの枚数と花弁同士の角度を用いて，合成画像を生成した. 花弁配置の残り 5 種 (A2, C1, C3, D1, E1) は，花弁のなす角が一定ではなかったため，今回は合成画像を作成しなかった.

合成画像は, 実際の花画像から切り取った花弁や花の中央部分を用いて作成した. まず, 表 1 に示す花弁配置のうちの 1 つをランダムに選択し, 花弁の枚数とずらす角度を決定した. そして, 花弁の枚数分だけ, 花弁を角度ずつずらしながら配置した. その際, 花弁の大きさとずらす角度に対して摂動を加えた. 最後に, 画像中央に花の中央部分を重ねることで，実際の花の見た目に近い合成画像を作成した. **図 8** に合成画像を作成する例を示す. 紫色の部分は各過程で新たに重ねた要素を示す． 本実験では**図 9(a)** に示す 5 種のイチリンソウの近縁種の花画像から作成した合成画像 50,000 枚 (1 種あたり 10,000 枚) を用いた. 図 9(b) に合成画像の例を示す. 合成画像の解像度は 1,400×1,400 pixels である. 合成画像一枚あたりに 4–10 枚の花弁が含まれており, 合成画像全体で合計 341,914 枚の花弁が含まれている.

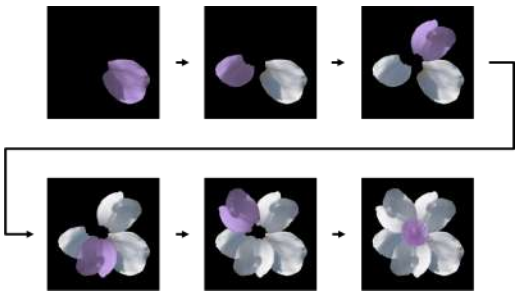


図 8: 合成画像の作成手順

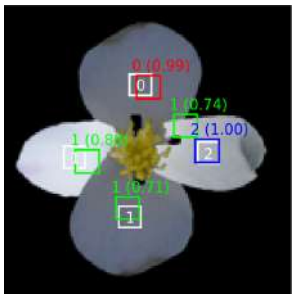
表 1: 花弁配置ごとの花弁の枚数とずらす角度

花弁配置	花弁の枚数	ずらす角度
A1	4	144°
A3	4	100°
B1	5	100°
B1	5	137°
C2	6	137°
D2	7	100°
D2	7	137°
E2	8	100°
E3	8	137°
F1	9	100°
F1	9	137°
G1	10	137°

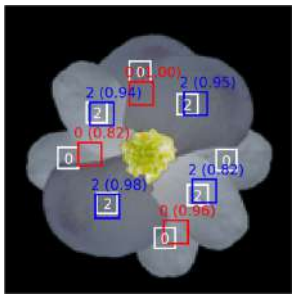
(a) 合成画像の素材の例

(b) 合成画像の例

図 9: 素材と合成した画像



(a) 過剰検出した例



(b) 検出漏れがある例

図 10: 花弁の前後関係の推定結果

4.2 実験条件

50,000 枚の合成画像のうち, 40,000 枚を学習データとして使用し, 10,000 枚をテストデータとして使用した. 学習データは, 合成画像の花弁の位置と前後関係を正解データとして使用した.

Conditional DETR の実装は, オープンソースのコードを使用した[*1]. Conditional DETR のバックボーンに, ResNet50 [28] を用いた. また, 3.2 節で述べたように, 損失関数には Focal Loss, L1 Loss, Generalized Intersection over Union Loss を用いた. 計算には, NVIDIA RTX A6000 (NVIDIA Tensor 336 コア, クロック周波数 1.5GHz, メモリ 48GB) を使用した.

花弁の前後関係の推定結果は, 検出された花弁の位置の矩形と正解データの矩形の Intersection over union (IoU) が 0.001 以上であり, クラスが一致している場合に正解とした. 花弁の大きさに対して, 出力する矩形が小さいため, 少しでも出力された矩形が正解の矩形に重なっていれば, 出力された矩形が花弁の領域に含まれる. そのため, 正しく検出されたとみなす IoU の閾値を低くした. 花弁の前後関係の推定結果は Precision, Recall, F1 Score で評価した. Precision は, 検出された花弁の前後関係のうち, 正しくクラスを推定した割合である. Recall は, 正解データの花弁の前後関係のうち, 正しく検出された割合である. F1 Score は, Precision と Recall の調和平均であり, Precision と Recall のバランスを示す指標である. 花弁配置の推定精度を高めるためには, 花弁の前後関係の Precision, Recall のどちらも高い値を示すことが重要である. そのため, これらの調和平均である F1 Score が重要である.

また, 花弁の前後関係の推定結果を用いて, 花弁配置を推定した. 花弁配置が正しく推定できた割合で花弁配置の推定精度を評価した.

4.3 実験結果

花弁ごとの前後関係の推定結果は, Precision は 0.831, Recall は 0.678, F1 Score は 0.747 であった. 提案手法により得られた花弁の前後関係の推定結果の一例を**図 10** に示す. 正解の位置と前後関係 (0: 外側, 1: 交互, 2: 内側) を白の矩形とその矩形内の数字で示している. また推定結果を赤 (0: 外側), 緑 (1: 交互), 青 (2: 内側) で示しており, 矩形の上の括弧内の数値は Conditional DETR が出力した確信度を示している.

図 10(a) のように, 一枚の画像に含まれる花弁の枚数が少ない場合は, 過剰検出 (false positive) が多い傾向にあった. 特に, 花弁同士が重なる部分が過剰検出されていた. 花弁の形状を正しく学習していない可能性があるため, 花弁全体の形状を検出するように矩形の大きさを変更すると過剰検出が減る可能性があると考えられる.

また, 図 10(b) のように, 一枚の画像に含まれる花弁の枚数が多い場合は, 検出漏れ (false negative) が多い傾向にあった. 特に, 他の花弁が覆い被さることで見えにくくなった花弁の検出漏れが目立った. これも同様に, 花弁の形状を正しく学習することで, 遮蔽された部分の形状を予測でき検出漏れが減る可能性があると考えられる.

F1 Score は 0.747 であり, Precision と Recall のバランスが取れていることがわかる. 花弁ごとの前後関係の推定精度を向上させるためには, 出力される矩形が各花弁を適切に囲むように設計し, 花弁の形状情報をより正確に学習可能とする手法が有効であると考えられる. 花弁配置の推定精度は 0.343 であった. **図 11** に花弁配置の推定結果の花弁配置と正解の花弁配置のデータ数を表す混同行列を示す. 縦軸が推定結果, 横軸が正解データを表している. 図 11 の赤い矩形で囲まれた対角線上には正しく推定されたデータ数を表すセルが存在している. 対角線で隔てられた右上の領域の方が, 左下の領域よりもデータが多いことから, 花弁の枚数が多い場合, 花弁の枚数が少ない花弁配置に誤って分類されることが多いことがわかる. このことから, 花弁の検出漏れが多いことがわかる.

5. まとめ

花弁配置を推測する作業は人手で行われており，その作業は労力を要する. そのため, 花弁配置の自動推定が求められている. しかし, これまでの研究では花弁配置の推定

[*1] https://github.com/Atten4Vis/ConditionalDETR

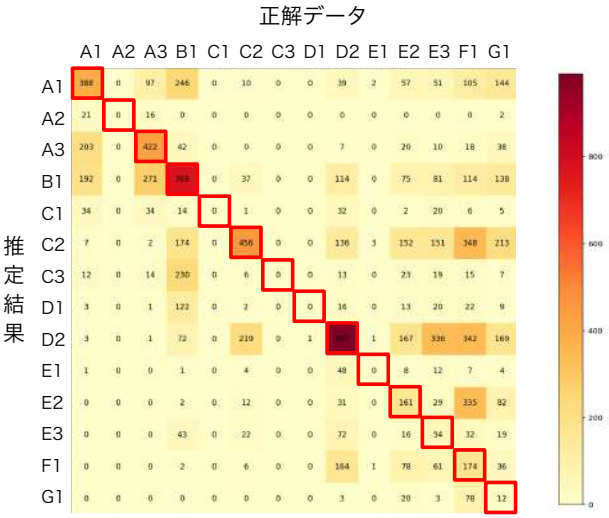


図 11: 花弁配置の推定結果の混同行列

精度が低く実用的ではなかった．花弁配置の推定精度が低下した要因として，タスクを複数の部分タスクに分解して解いていることが考えられた．そこで，本研究では，花弁配置の推定精度を高めるために，先行研究では花弁の重なり位置の推定とその位置の花弁の前後関係の推定という 2 つのタスクに分かれていた処理を花弁の前後関係を推定する 1 つの処理にまとめる手法を提案した．実験の結果，花弁の前後関係の推定精度は F1 Score で 0.747 であり，花弁配置の推定精度は 0.343 であった．

今後は，3 つの課題に取り組む．まず，花弁配置の精度向上のため，これまで花弁の中心の一部の領域を検出していたものを，花弁全体を囲む矩形となるよう変更する．このことで，物体検出の位置情報の推定の精度向上が見込め，より高い精度で花弁配置を推定することが可能となる．2 つ目は，今回合成しなかった花弁配置の画像を生成することである．今回，花弁同士のなす角が一定の花弁配置をもつ花画像のみを合成して実験しており，配置が知られている全ての配置については，学習や推定を行っていない．そこで，新たな合成画像生成手法を用いて，本研究で合成できなかった花弁配置の花画像を生成する．そして，再度花弁の前後関係や花弁配置の推定をし，学習データに花弁配置が増えたことがこれらの精度に与える影響を評価する．3 つ目は，実画像に提案手法を適用し，その性能を評価する．本稿では，モデルの学習のために大量の画像が必要であることから合成画像を用いたが，本研究の最終的な目的は，実際の花画像に対する花弁配置の推定である．合成画像を用いて学習したモデルを実画像を用いて fine-tuning し，実画像に対する花弁配置の推定性能を評価する．

参考文献

[1] Kitazawa, M. S. and Fujimoto, K.: Perianth phyllotaxis is polymorphic in the basal eudicot Anemone and Eranthis species, *Frontiers in Ecology and Evolution*, Vol. 8, No. 70, pp. 1–10 (2020).

[2] Nakagawa, A., Kitazawa, M. S. and Fujimoto, K.: A Design Principle for Floral Organ Number and Arrangement in Flowers with Bilateral Symmetry, *Development*, Vol. 147, No. dev182907, pp. 1–10 (2020).

[3] Nakatani, T., Utsumi, Y., Fujimoto, K., Iwamura, M. and Kise, K.: Image Recognition-Based Petal Arrangement Estimation, *Frontiers in Plant Science*, Vol. 15, No. 1334362, pp. 1–14 (2024).

[4] Lecun, Y., Bottou, L., Bengio, Y. and Haffner, P.: Gradient-based learning applied to document recognition, *Proceedings of the IEEE*, Vol. 86, No. 11, pp. 2278–2324 (1998).

[5] Chan, W., Jaitly, N., Le, Q. and Vinyals, O.: Listen, Attend and Spell: A Neural Network for Large Vocabulary Conversational Speech Recognition, *Proceedings of 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4960–4964 (2016).

[6] Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., Back, T., Petersen, S., Reiman, D., Clancy, E., Zielinski, M., Steinegger, M., Pacholska, M., Berghammer, T., Bodenstein, S., Silver, D., Vinyals, O., Senior, A. W., Kavukcuoglu, K., Kohli, P. and Hassabis, D.: Highly Accurate Protein Structure Prediction with AlphaFold, *Nature*, Vol. 596, No. 7873, pp. 583–589 (2021).

[7] Guru, D. S., Sharath Kumar, Y. H. and Manjunath, S.: Textural Features in Flower Classification, *Mathematical and Computer Modelling*, Vol. 54, No. 3, pp. 1030–1036 (2011).

[8] Nilsback, M.-E. and Zisserman, A.: Automated Flower Classification over a Large Number of Classes, *Proceedings of 2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pp. 722–729 (2008).

[9] Wang, Z., Wang, K., Wang, X. and Pan, S.: A Convolutional Neural Network Ensemble for Flower Image Classification, *Proceedings of the 2020 9th International Conference on Computing and Pattern Recognition*, pp. 225–230 (2021).

[10] Rahim, U. F. and Mineno, H.: Tomato Flower Detection and Counting in Greenhouses Using Faster Region-Based Convolutional Neural Network, *Journal of Image and Graphics*, Vol. 8, No. 4, pp. 107–113 (2020).

[11] Ohi, N., Lassak, K., Watson, R., Strader, J., Du, Y., Yang, C., Hedrick, G., Nguyen, J., Harper, S., Reynolds, D., Kilic, C., Hikes, J., Mills, S., Castle, C., Buzzo, B., Waterland, N., Gross, J., Park, Y.-L., Li, X. and Gu, Y.: Design of an Autonomous Precision Pollination Robot, *Proceedings of 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7711–7718 (2018).

[12] Lin, P., Lee, W. S., Chen, Y. M., Peres, N. and Fraisse, C.: A Deep-Level Region-Based Visual Representation Architecture for Detecting Strawberry Flowers in an Outdoor Field, *Precision Agriculture*, Vol. 21, No. 2, pp. 387–402 (2020).

[13] Wu, D., Lv, S., Jiang, M. and Song, H.: Using Channel Pruning-Based YOLO v4 Deep Learning Algorithm for the Real-Time and Accurate Detection of Apple Flowers in Natural Environments, *Computers and Electronics in Agriculture*, Vol. 178, No. 105742, pp. 1–12 (2020).

[14] Liu, S., Li, X., Wu, H., Xin, B., Tang, J., Petrie, P. R.

and Whitty, M.: A Robust Automated Flower Estimation System for Grape Vines, *Biosystems Engineering*, Vol. 172, pp. 110–123 (2018).

[15] Fu, H., Gong, M., Wang, C., Batmanghelich, K. and Tao, D.: Deep Ordinal Regression Network for Monocular Depth Estimation, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2002–2011 (2018).

[16] Meng, X., Fan, C., Ming, Y. and Yu, H.: CORNet: Context-Based Ordinal Regression Network for Monocular Depth Estimation, *Proceedings of IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 32, No. 7, pp. 4841–4853 (2022).

[17] Niu, Z., Zhou, M., Wang, L., Gao, X. and Hua, G.: Ordinal Regression with Multiple Output CNN for Age Estimation, *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4920–4928 (2016).

[18] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A.: You Only Look Once: Unified, Real-Time Object Detection, *Proceedings of 2016 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788 (2016).

[19] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A. and Zagoruyko, S.: End-to-End Object Detection with Transformers, *Proceedings of European Conference on Computer Vision*, pp. 213–229 (2020).

[20] Naseer, M. M., Ranasinghe, K., Khan, S. H., Hayat, M., Shahbaz Khan, F. and Yang, M.-H.: Intriguing Properties of Vision Transformers, *Proceedings of Advances in Neural Information Processing Systems*, Vol. 34, pp. 23296–23308 (2021).

[21] Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F. A. and Brendel, W.: ImageNet-Trained CNNs are biased towards texture, *Proceedings of International conference on learning representations* (2019).

[22] Baker, N., Lu, H., Erlikhman, G. and Kellman, P. J.: Deep Convolutional Networks Do Not Classify Based on Global Object Shape, *PLOS Computational Biology*, Vol. 14, No. e1006613, pp. 1–43 (2018).

[23] Meng, D., Chen, X., Fan, Z., Zeng, G., Li, H., Yuan, Y., Sun, L. and Wang, J.: Conditional DETR for Fast Training Convergence, *Proceedings of 2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3631–3640 (2021).

[24] Lin, T.-Y., Goyal, P., Girshick, R., He, K. and Dollar, P.: Focal Loss for Dense Object Detection, *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 318–327 (2017).

[25] Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I. and Savarese, S.: Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 658–666 (2019).

[26] Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J. and Amodei, D.: Scaling Laws for Neural Language Models, No. arXiv:2001.08361 (2020).

[27] Henighan, T., Kaplan, J., Katz, M., Chen, M., Hesse, C., Jackson, J., Jun, H., Brown, T. B., Dhariwal, P., Gray, S., Hallacy, C., Mann, B., Radford, A., Ramesh, A., Ryder, N., Ziegler, D. M., Schulman, J., Amodei, D. and McCandlish, S.: Scaling Laws for Autoregressive Generative Modeling, No. arXiv:2010.14701 (2020).

[28] He, K., Zhang, X., Ren, S. and Sun, J.: Deep Residual Learning for Image Recognition, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778 (2016).