

領域認識用データセット

**MS COCO**

Microsoft COCO: Common Objects in Context

1. COCO の認識課題と領域注釈の実例
2. 画像の収集方法と物体注釈の構造
3. 物体検出・領域分割の評価とモデル性能
4. COCO の利用動向と画像・注釈の入手方法

# 1. COCO の認識課題と領域注釈の実例

# COCO の公式サイト

**COCO**  
Common Objects in Context

info@cocodataset.org

[Home](#) [People](#) [Dataset](#) [Tasks](#) [Evaluate](#)

## News

- We are pleased to announce the [LVIS 2021 Challenge and Workshop](#) to be held at ICCV.
- Please note that there will not be a COCO 2021 Challenge, instead, we encourage people to participate in the LVIS 2021 Challenge.
- We have partnered with the team behind the open-source tool [FiftyOne](#) to make it easier to download, visualize, and evaluate COCO
- [FiftyOne](#) is an open-source tool facilitating visualization and access to COCO data resources and serves as an evaluation tool for model analysis on COCO.

## What is COCO?

COCO is a large-scale object detection, segmentation, and captioning dataset.  
COCO has several features:

- ✓ Object segmentation
- ✓ Recognition in context
- ✓ Superpixel stuff segmentation
- ✓ 330K images (>200K labeled)
- ✓ 1.5 million object instances
- ✓ 80 object categories
- ✓ 91 stuff categories
- ✓ 5 captions per image
- ✓ 250,000 people with keypoints

## Collaborators

Tsung-Yi Lin Google Brain  
Genevieve Patterson MSR, Trash TV  
Matteo R. Ronchi Caltech  
Yin Cui Google  
Michael Maire TTI-Chicago  
Serge Belongie Cornell Tech  
Lubomir Bourdev WaveOne, Inc.  
Ross Girshick FAIR  
James Hays Georgia Tech  
Pietro Perona Caltech  
Deva Ramanan CMU  
Larry Zitnick FAIR  
Piotr Dollar FAIR

## Sponsors

CVDF  
Microsoft  
facebook  
Mighty Ai

[Research Paper](#) [Download](#)

## 物体のカテゴリ・位置・領域を推定する

- COCO は、画像内の複数物体を対象とする検出・領域分割用データセットである。
- 正解は画像全体のラベルだけでなく、物体ごとのカテゴリ・位置・領域を含む。
- 物体検出、インスタンス分割、Panoptic 分割では、予測する単位が異なる。

## 認識課題ごとの予測対象と出力単位

| 課題          | 予測するもの         |
|-------------|----------------|
| 物体検出        | 物体ごとのカテゴリと矩形   |
| インスタンス分割    | 物体ごとのカテゴリと画素領域 |
| 意味的領域分割     | 各画素のカテゴリ       |
| Panoptic 分割 | 各画素のカテゴリと領域 ID |

検出は位置を矩形で示す。分割は画素単位で領域を示し、個体を区別するか、背景まで含めるかが異なる。

## 物体検出はカテゴリと外接矩形を出力する

- 物体検出では、画像内の各物体にカテゴリと矩形の位置を割り当てる。
- 同じカテゴリの物体が複数あれば、それぞれ別の矩形として検出する。
- 矩形には背景の画素も含まれる。物体の輪郭そのものを表す領域マスクとは異なる。

## インスタンス分割は物体ごとに画素を分類する

- インスタンスとは、画像に写る個々の物体を指す。2本の傘は同じカテゴリでも別の個体である。
- 各物体について、その物体に属する画素の集合を領域マスクとして予測する。
- 重なりや遮蔽のある場面でも、見えている領域を個体ごとに区別する。

## 食卓の複数の物体を領域ごとに分ける



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000009 / COCO128-seg 公開変換注釈の輪郭を重ねて表示

## 遮蔽された人と重なり合う傘を分ける



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000315 / COCO128-seg 公開変換注釈の輪郭を重ねて表示

## 同一画像に付いた物体領域と背景の注釈



[1] <https://github.com/cocodataset/panopticapi>

COCO image ID 000000142238 / 公式 Panoptic API サンプル画像。

## 個体ごとの分割では人の領域を区別する



14 人とボールを  
個別の領域で表す

木・空・芝は対象外

色は領域の識別用

[1] <https://github.com/cocodataset/panopticapi>

COCO image ID 000000142238 / 公式注釈の thing 領域のみ表示。黒は背景または未注釈。

## 意味的領域分割は画素ごとのカテゴリを決める

- 意味的領域分割は、各画素に人・木・空などのカテゴリを割り当てる課題である。
- 同じカテゴリの人が複数いても、個体ごとの識別は要求しない。
- インスタンス分割は個体を区別するが、通常は草や空のような背景領域を対象外とする。

## 意味的分割では 14 人を同じカテゴリで表す



人は同じ青色

木・空・芝は  
それぞれ別カテゴリ

個体 ID は使わない

[1] <https://github.com/cocodataset/panopticapi>

COCO image ID 000000142238 / 公式 Panoptic 注釈をカテゴリ単位へ統合。未注釈画素は黒。

# Panoptic 分割は全体のカテゴリと個体を統合する

- Panoptic 分割は、各画素にカテゴリと領域 ID を割り当て、画像全体を重複しない領域へ分ける。
- 数えられる物体を thing と呼び、人やボールを個体ごとに区別する。
- 広がりを持つ背景を stuff と呼び、木・空・芝にもカテゴリを付ける。未注釈領域は別扱いとする。

[1] Alexander Kirillov et al. (2019). IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9404–9413. Panoptic Segmentation.

<https://arxiv.org/abs/1801.00868>

[2] <https://github.com/cocodataset/panopticapi>

# Panoptic 分割では人と背景を同時に表す



[1] <https://github.com/cocodataset/panopticapi>

COCO image ID 000000142238 / 公式 Panoptic API サンプルの正解領域

## 2. 画像の収集方法と物体注釈の構造

## ImageNet は大規模な画像分類を整備した

- 2009 年に発表された ImageNet は、単語の概念を整理した WordNet に対応付けて画像を集めたデータベースである。
- 画像分類大会 ILSVRC 2012 では、1,000 クラスの画像分類が広く使われた。
- COCO は分類カテゴリの拡大よりも、同一画像内の複数物体の位置と画素領域を重視する。

## PASCAL VOC は 20 カテゴリの物体認識を評価した

- PASCAL Visual Object Classes （ VOC ） は、物体の分類・検出・領域分割を扱う先行ベンチマークである。
- VOC 2012 は人・動物・乗り物など 20 物体カテゴリを扱い、矩形や領域の正解を提供する。
- COCO の 80 物体カテゴリは、 VOC の 20 カテゴリの 4 倍に当たる。

[1] <http://host.robots.ox.ac.uk/pascal/VOC/voc2012/>

[2] <https://cocodataset.org/#external>

## 先行データと COCO では正解の単位が異なる

| 対象              | 代表的な正解単位 | カテゴリ  |
|-----------------|----------|-------|
| ILSVRC 2012 分類  | 画像全体のラベル | 1,000 |
| PASCAL VOC 2012 | 物体の矩形・領域 | 20    |
| COCO 2017 検出    | 物体の矩形・領域 | 80    |

[1] <https://www.image-net.org/>

[2] <https://cocodataset.org/#format-data>

## COCO は複数物体と遮蔽を含む場面を対象とする

- 単一の大きな物体を中心に置いた画像だけでは、日常場面の物体認識を十分に評価できない。
- COCO は、複数カテゴリの共起、小さな物体、重なりや遮蔽を含む場면을重視した。
- Tsung-Yi Lin らが ECCV 2014 で発表し、画像選定と物体領域の注釈をデータ設計の中心に据えた。

## 画像選定と領域注釈を段階に分ける

- 写真共有サービス Flickr から日常場面の画像を収集し、物体カテゴリの組合せで多様な文脈を探した。
- カテゴリの存在確認、個体の特定、輪郭の作成を分け、各段階で注釈品質を確認した。
- 現在の物体注釈は、カテゴリ・矩形・領域マスクなどを同じレコードにまとめる。

[1] Tsung-Yi Lin et al. (2014). European Conference on Computer Vision (ECCV), pp. 740–755. Microsoft COCO: Common Objects in Context. <https://arxiv.org/abs/1405.0312>

[2] <https://cocodataset.org/#format-data>

## カテゴリ確認から物体の輪郭へ注釈を具体化する

- カテゴリの存在確認では、画像に「人」や「傘」が写っているかを判定する。
- 個体の特定では、何人・何本の傘があり、どの位置にあるかを分ける。
- 輪郭注釈では、それぞれの個体の見えている境界を画素領域として記録する。

## 画像・カテゴリ・個々の注釈を ID で対応付ける

- 1 枚の画像に複数の物体注釈が対応し、各注釈は画像 ID とカテゴリ ID を参照する。
- 物体注釈は、矩形 bbox、領域 segmentation、面積 area、集団領域の指定 iscrowd などを持つ。
- 領域の記録には、輪郭点を並べる多角形と、同じ画素値の連続長で表すランレングス符号化（RLE）が使われる。

## bbox は左上座標と矩形の幅・高さを表す

- bbox は [x, y, width, height] の順で、画像左上を原点として記録する。
- たとえば [20, 30, 100, 60] は、左上が (20, 30)、右下が (120, 90) の矩形を表す。
- area は物体の領域マスクの面積を表す。

## 多角形は物体の輪郭を座標列で表す

- 多角形は  $[x1, y1, x2, y2, \dots]$  の順に輪郭点を並べ、隣接する点と最後・最初の点を結ぶ。
- たとえば  $[1, 1, 4, 1, 4, 3, 1, 3]$  は、4 頂点で囲まれた矩形形状の領域を表す。
- 個体が複数の部分に分かれる場合は、1 物体の segmentation に複数の多角形を持たせられる。

[1] <https://cocodataset.org/#format-data>

[2] <https://github.com/cocodataset/cocoapi/blob/master/PythonAPI/pycocotools/mask.py>

## 領域マスクは各画素を 0 または 1 で表す

|   |   |   |   |   |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

1 は対象の画素

0 は対象外の画素

この領域の面積は 6 画素

画像の大きさは 4 行 × 5 列

マスクは輪郭点の列ではなく、画像と同じ縦横を持つ配列である。

[1] <https://github.com/cocodataset/cocoapi/blob/master/PythonAPI/pycocotools/mask.py>

4 行 × 5 列の符号化説明例。

## RLE は列方向の走査を連続長に置き換える

|   |   |   |   |   |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 0 | 0 |

左の列の上から下へ読む。  
次の列も上から下へ読む。

0 が 5 個 → 1 が 2 個 → 0 が 2 個  
1 が 2 個 → 0 が 2 個  
1 が 2 個 → 0 が 5 個

```
counts = [5, 2, 2, 2, 2, 2, 5]
```

先頭は 0 の個数。最初の画素が 1 なら、先頭の連続長は 0 になる。

[1] <https://github.com/cocodataset/cocoapi/blob/master/PythonAPI/pycocotools/mask.py>

4 行 × 5 列の符号化説明例。

## 公式 API で同じマスクを符号化・復元する

```
from pycocotools import mask
rle = {"size": [4, 5], "counts": [5, 2, 2, 2, 2, 2, 5]}
encoded = mask.frPyObjects(rle, 4, 5)
print(encoded["counts"])          # b'5220003'
print(int(mask.area(encoded)))    # 6
restored = mask.decode(encoded)
```

配布用の圧縮 RLE では連続長を文字列に符号化する。

[1] <https://github.com/cocodataset/cocoapi/blob/master/PythonAPI/pycocotools/mask.py>

## COCO 2017 の画像分割

| 分割    | 画像数の概数 | 正解領域 |
|-------|--------|------|
| train | 118K   | 公開   |
| val   | 5K     | 公開   |
| test  | 41K    | 非公開  |

2017 年に、2014 年版の train 83K / val 41K を同じ画像で再分割した [1]。

[1] <https://cocodataset.org/#download>

## COCO 全体と 2017 年版 train ・ val の画像数

- 公式サイトは全体を約 330K 画像、うち 200K 超を注釈付きとし、約 150 万物体を収録すると説明する。
- 2017 年版の train + val は約 123K 画像。検出用の正解領域が公開されている。

## 物体・背景のカテゴリ数と注釈体系

- 物体検出・インスタンス分割は 80 カテゴリ。人、乗り物、動物、食品、家具などを含む。
- COCO-Stuff は 80 物体 + 91 背景の 171 カテゴリ。
- COCO Panoptic は 80 物体 + 53 背景の 133 カテゴリ。背景カテゴリを統合した体系を使う [1, 2]。

[1] <https://github.com/nightrome/cocostuff>

[2] [https://github.com/cocodataset/panopticapi/blob/master/panoptic\\_coco\\_categories.json](https://github.com/cocodataset/panopticapi/blob/master/panoptic_coco_categories.json)

# 実画像の傘で面積の大きさを比べる



**A small**

582 画素

1,024 画素未満

**B medium**

3,629 画素

1,024 ~ 9,216 画素

**C large**

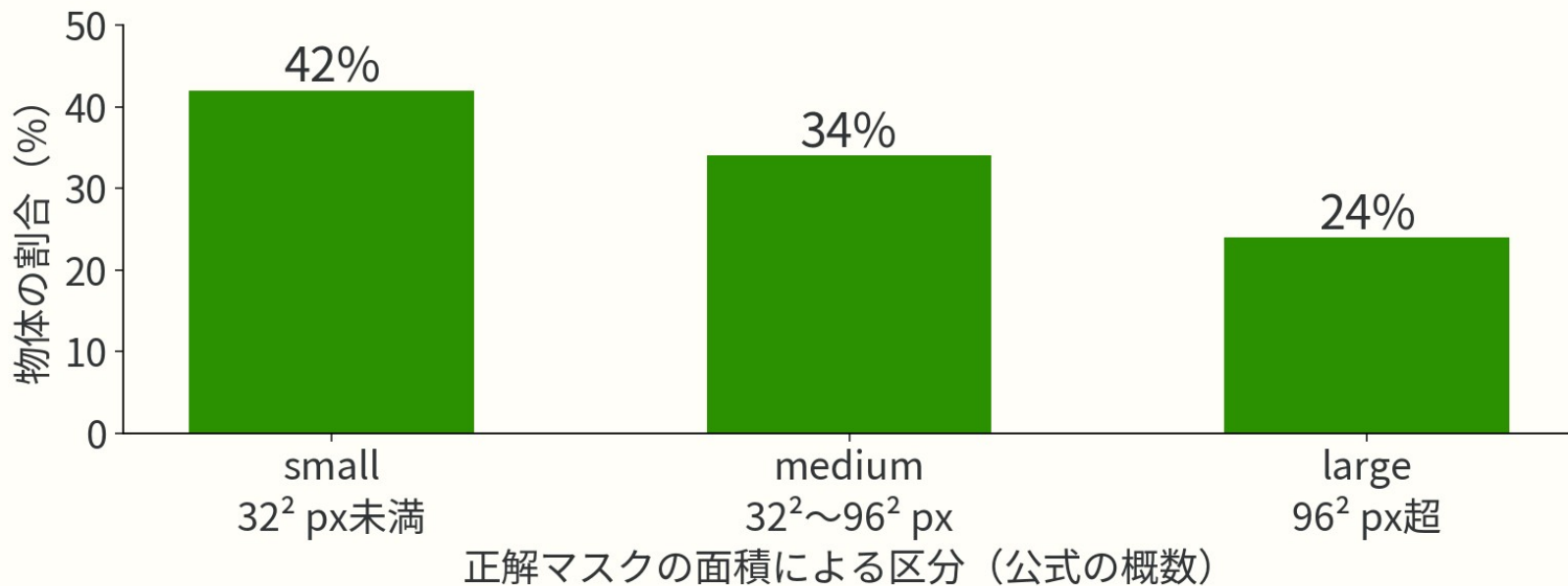
10,803 画素

9,216 画素超

[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000315 / COCO128-seg 公開変換ポリゴンを画素化した面積

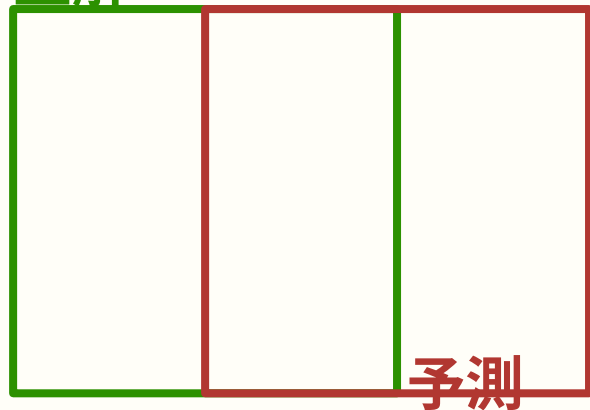
## 物体面積の区分と COCO 全体の構成比



## 3. 物体検出・領域分割の評価とモデル性能

## IoU は領域の重なりを測る

正解



$$\text{IoU} = \text{共通部分} \div \text{和集合}$$

この矩形例では  $2 \div 6 = 1/3$   
IoU 0.50 で一致とはならない。  
マスクは画素単位で評価する。

## 予測と正解の対応から TP ・ FP ・ FN を数える

- 正しいカテゴリで正解領域へ一致した予測を真陽性 TP とする。
- 正解へ一致しない予測や、同じ物体への重複した予測を偽陽性 FP とする。
- 予測されなかった正解物体を偽陰性 FN とする。カテゴリと IoU 閾値で一致の条件が決まる。

## 適合率と再現率は異なる分母で計算する

| 指標            | 分母               | 値              |
|---------------|------------------|----------------|
| 適合率 Precision | 予測 3 件 = TP + FP | $2/3 = 66.7\%$ |
| 再現率 Recall    | 正解 4 件 = TP + FN | $2/4 = 50.0\%$ |

誤検出が増えると適合率が下がり、見落としが増えると再現率が下がる。

[1] <https://cocodataset.org/#detection-eval>

TP=2、FP=1、FN=2 とした計算例。

## AP は全予測を集計して評価する

- AP（平均適合率）は、カテゴリごとに予測を信頼度順に並べ、正解との一致を使って求める。
- 再現率 0 ～ 1 の 101 点で適合率を補間し、その平均を AP とする。
- COCO の主指標は、カテゴリと IoU 0.50 ～ 0.95 の 10 閾値で平均した AP である。矩形の検出は box AP、個体領域は mask AP と呼ぶ。

## AP50 と主指標 AP では要求する位置精度が異なる

- AP50 は IoU 閾値 0.50 で計算する。IoU 0.62 の予測は、他の一致条件も満たせば対応候補になる。
- IoU 閾値 0.75 では、同じ予測は対応候補にならない。
- COCO の主指標 AP は、0.50 から 0.95 まで 0.05 刻みの 10 閾値で求めた AP を平均する。

## 数えにくい集団には crowd 注釈がある

- 密集して個体ごとの区切りが難しい領域は、iscrowd=1 でまとめて注釈する。
- crowd への一致は通常の実陽性に数えず、条件を満たす予測を評価から除外する。
- 同じ物体への重複検出や、小さい対象の見落としも AP に影響する [1]。

[1] <https://github.com/cocodataset/cocoapi/blob/master/PythonAPI/pycocotools/cocoeval.py>

## 物体検出モデルの box AP と評価集合

| 方式・設定                   | 年           | box AP | 評価集合     |
|-------------------------|-------------|--------|----------|
| Faster R-CNN R50-FPN 3x | 2015 / 2017 | 40.2   | val2017  |
| RetinaNet R50-FPN 3x    | 2017        | 38.7   | val2017  |
| RT-DETRv2-L 640 px      | 2024        | 53.4   | COCO val |

R50 は 50 層の画像特徴モデル、FPN は多解像度の特徴統合、3x は学習期間の設定を表す。

[1] [https://github.com/facebookresearch/detectron2/blob/main/MODEL\\_ZOO.md](https://github.com/facebookresearch/detectron2/blob/main/MODEL_ZOO.md)

[2] <https://github.com/lyuwenyu/RT-DETR>

## 検出方式は領域候補を使うかどうかで異なる

- Faster R-CNN （2015）は候補領域を作り、その領域についてカテゴリと位置を推定する二段階方式である。
- RetinaNet （2017）は画像上の多数の位置から直接検出する一段階方式で、学習中の易しい負例の影響を抑える。
- RT-DETRv2 （2024）は、物体間の関係を扱う Transformer を用いた実時間検出器 RT-DETR （2023）を改良した。

[1] Ren et al. (2015). NeurIPS. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. <https://arxiv.org/abs/1506.01497>

[2] Lin et al. (2017). ICCV. Focal Loss for Dense Object Detection. <https://arxiv.org/abs/1708.02002>

[3] Lv et al. (2024). arXiv:2407.17140. RT-DETRv2: Improved Baseline with Bag-of-Freebies for Real-Time Detection Transformer. <https://arxiv.org/abs/2407.17140>

## R50 ・ FPN ・ 3x はモデルと学習設定を表す

- R50 は、画像から特徴を抽出する 50 層の ResNet-50（2016）を指す。
- FPN は、異なる解像度の特徴を統合する Feature Pyramid Networks（2017）を指す。
- 3x は公開実装の学習スケジュール名で、標準的な 1x より長く学習する設定を示す。

[1] He et al. (2016). CVPR, pp. 770–778. Deep Residual Learning for Image Recognition. <https://arxiv.org/abs/1512.03385>

[2] Lin et al. (2017). CVPR. Feature Pyramid Networks for Object Detection. <https://arxiv.org/abs/1612.03144>

[3] [https://github.com/facebookresearch/detectron2/blob/main/MODEL\\_ZOO.md](https://github.com/facebookresearch/detectron2/blob/main/MODEL_ZOO.md)

## Panoptic 分割を Panoptic Quality で評価する

- Panoptic Quality (PQ) は、領域形状の精度と物体の検出精度をまとめて測る指標である。
- 同じカテゴリの正解と予測を  $\text{IoU} > 0.50$  で対応付け、TP・FP・FN を数える。
- 対応した領域の  $\text{IoU}$  の和を  $\text{TP} + 0.5\text{FP} + 0.5\text{FN}$  で割り、カテゴリごとの値を平均する。

## 1 領域の一致から求める PQ の計算例

- 同一カテゴリの正解 2 領域に対し、予測 2 領域のうち 1 領域が IoU 0.80 で一致したとする。
- $TP=1$ 、 $FP=1$ 、 $FN=1$  なので、 $PQ=0.80 \div (1 + 0.5 + 0.5)=0.40$ 。
- 実評価はカテゴリごとに全画像の一致・誤検出・見落としを集計し、カテゴリ間で平均する [1]。

[1] <https://github.com/cocodataset/panopticapi/blob/master/panopticapi/evaluation.py> / 数値は計算説明の例

## 4. COCO の利用動向と画像・注釈の入手方法

## COCO の公開・分割更新と検出モデルへの利用

- 2014 年に物体の文脈と領域を重視したデータとして登場し、2017 年に学習・検証分割を再編した。
- 2018 年には Panoptic 分割を追加した [1]。
- 2026-10-15 の確認では、RT-DETR 系の公式実装が COCO での評価設定と結果を公開している [2]。

[1] <https://cocodataset.org/#download>

[2] <https://github.com/lyuwenyu/RT-DETR>

## LVIS は COCO 画像を大語彙の物体認識へ拡張した

- Large Vocabulary Instance Segmentation （ LVIS 、 2019 ） は、COCO 画像へ 1,000 以上の物体カテゴリの注釈を追加した。
- 出現頻度の低いカテゴリを多く含み、頻出・少数カテゴリ別の認識性能を評価する。

[1] Agrim Gupta et al. (2019). IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5356–5364. LVIS: A Dataset for Large Vocabulary Instance Segmentation. <https://arxiv.org/abs/1908.03195>

[2] <https://github.com/lvis-dataset/lvis-api>

## COCO 2017 の配布ファイルと圧縮容量

| 公式配布物              | 圧縮容量の目安 |
|--------------------|---------|
| train2017 画像 ZIP   | 18 GB   |
| val2017 画像 ZIP     | 1 GB    |
| train + val 注釈 ZIP | 241 MB  |
| Panoptic 注釈 ZIP    | 821 MB  |

画像の利用条件は Flickr 側のライセンスも確認する [1]。

[1] <https://cocodataset.org/#download>

## 公式 API で物体注釈を読み込む

```
from pycocotools.coco import COCO

coco = COCO("annotations/instances_val2017.json")
ids = coco.getImgIds(catIds=coco.getCatIds(catNms=["cat"]))
info = coco.loadImgs(ids[0])[0]
anns = coco.loadAnns(coco.getAnnIds(imgIds=[info["id"]]))
```

pycocotools を導入し、公式配布 ZIP を展開して使う。画像ファイルは画像 ZIP から別に取得する。

[1] <https://github.com/cocodataset/cocoapi/blob/master/PythonAPI/pycocoDemo.ipynb>

## API は画像 ID から物体注釈とマスクを取り出す

- COCO.getImgIds は条件に合う画像 ID を返し、loadImgs でファイル名・幅・高さなどを読み込む。
- getAnnIds と loadAnns で、その画像に対応する物体注釈を取り出す。category\_id はカテゴリ表と照合する。
- annToMask は多角形や RLE を画像と同じ縦横の二値マスクへ変換する。

## COCO は複数物体の位置と領域を評価する

- COCO は、物体のカテゴリ・矩形・個体領域を注釈し、日常場面の複数物体認識を評価する。
- Panoptic 分割は背景領域も含め、各画素のカテゴリと領域 ID を統合する。
- 検出・個体分割は AP、Panoptic 分割は PQ で評価する。

# Appendix

1. 開発者の研究と領域認識の評価仕様
2. 128 画像の部分集合における注釈数の分布
3. 画像と物体領域の追加例
4. 全 80 物体カテゴリと画像・注釈の利用条件
5. 物体検出・領域分割の関連文献

## 原著の著者と研究の位置付け

- 第一著者 Tsung-Yi Lin は、物体認識・検出の研究者。後の Feature Pyramid Networks でも第一著者を務めた。
- 最終著者 C. Lawrence Zitnick は、視覚認識や画像理解の研究者。
- 原著は ECCV 2014 で発表され、日常場面の文脈と物体領域の注釈をデータ設計の中心に据えた [1]。

[1] Tsung-Yi Lin et al. (2014). European Conference on Computer Vision (ECCV), pp. 740–755. Microsoft COCO: Common Objects in Context. <https://arxiv.org/abs/1405.0312>

## 同じ画像でもタスクごとに正解が異なる

| 課題          | 出力単位         | 主指標     |
|-------------|--------------|---------|
| 物体検出        | 物体ごとの矩形      | box AP  |
| インスタンス分割    | 物体ごとの領域      | mask AP |
| 意味的領域分割     | 各画素のカテゴリ     | mIoU    |
| Panoptic 分割 | カテゴリ + 個体 ID | PQ      |

[1] <https://cocodataset.org/#panoptic-eval>

[2] <https://github.com/nightrome/cocostuff>

## 通常注釈と crowd 注釈の扱い

- 通常の一一致判定は、交差面積を和集合面積で割る IoU を用いる。
- crowd との判定は交差面積を予測面積で割る。一つの crowd 領域へ複数の予測を対応付けられる。
- crowd 正解は再現率の分母から除外され、そこへ一致した予測も適合率の集計から除外される [1]。

[1] <https://github.com/cocodataset/cocoapi/blob/master/common/maskApi.c>  
<https://github.com/cocodataset/cocoapi/blob/master/PythonAPI/pycocotools/cocoeval.py>

## AP の平均には複数の軸がある

- 各カテゴリ・各 IoU 閾値で、信頼度順の予測から適合率・再現率を作る。
- 再現率 0.00 ~ 1.00 の 101 点で補間した適合率を平均し、カテゴリと 10 個の IoU 閾値で平均する。
- 通常 AP は maxDets=100 。公式 Python 実装では、画像・カテゴリごとに上位 100 予測を使う [1]。

[1] <https://github.com/cocodataset/cocoapi/blob/master/PythonAPI/pycocotools/cocoeval.py>

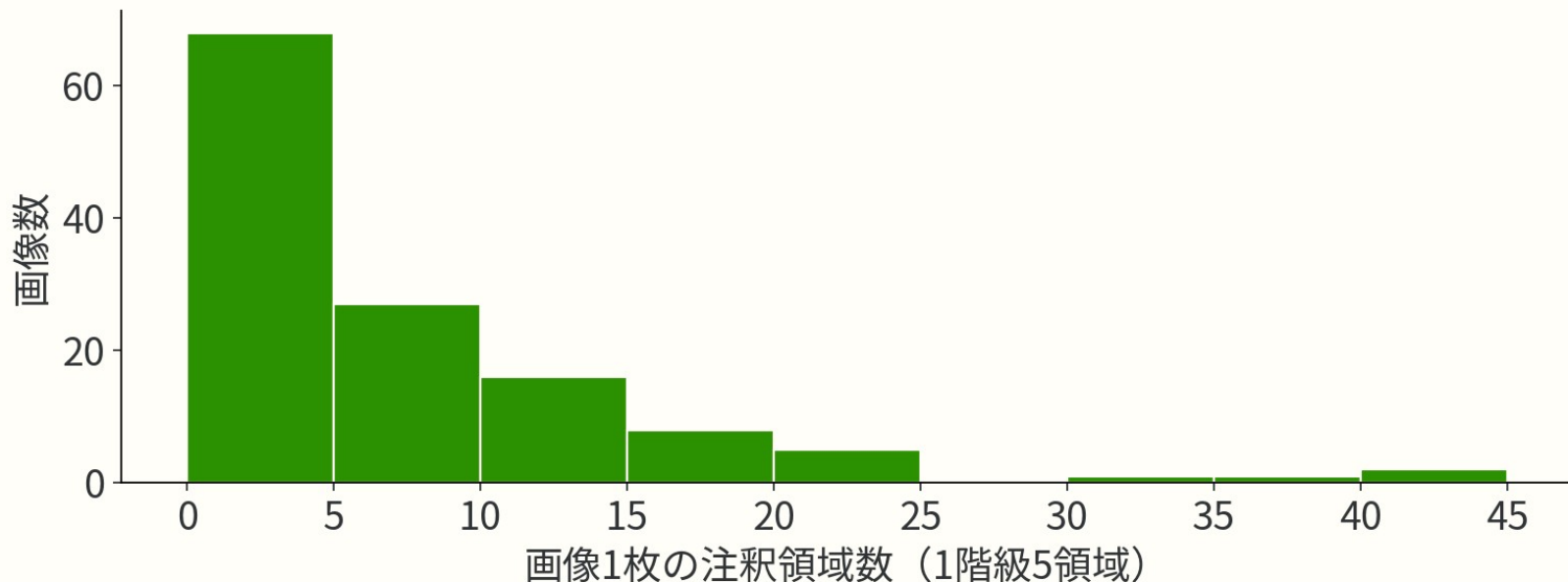
## 注釈数の分析対象とする 128 画像の部分集合

分析対象は COCO128-seg の 128 画像である [1]。

公開変換された領域注釈を画像ごとに数え、注釈ファイルがない 2 画像は 0 領域として含める。

[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

## 128 画像の注釈領域数の分布



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

128 画像すべて。画像ごとの注釈行数、1 階級 5 領域、表示範囲外 0 画像。

## 128 画像の注釈領域数の記述統計

| 統計量            | 注釈領域数／画像 |
|----------------|----------|
| 平均／中央値         | 7.26 ／ 4 |
| 母標準偏差          | 7.76     |
| 第 1 ／ 第 3 四分位点 | 2 ／ 10   |
| 最小／最大          | 0 ／ 42   |

全 128 画像を等重みで計算。分位点は線形補間。80 カテゴリ中 71 カテゴリの注釈を含む。

[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

## 最多の 42 領域を含む食卓



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000196 / COCO128-seg 公開変換注釈の輪郭を重ねて表示

## 注釈が 0 領域の画像にも物体は写る



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000250 / COCO128-seg 公開変換注釈の輪郭を重ねて表示

# キリン giraffe



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000025 / COCO128-seg 公開変換注釈の輪郭を重ねて表示

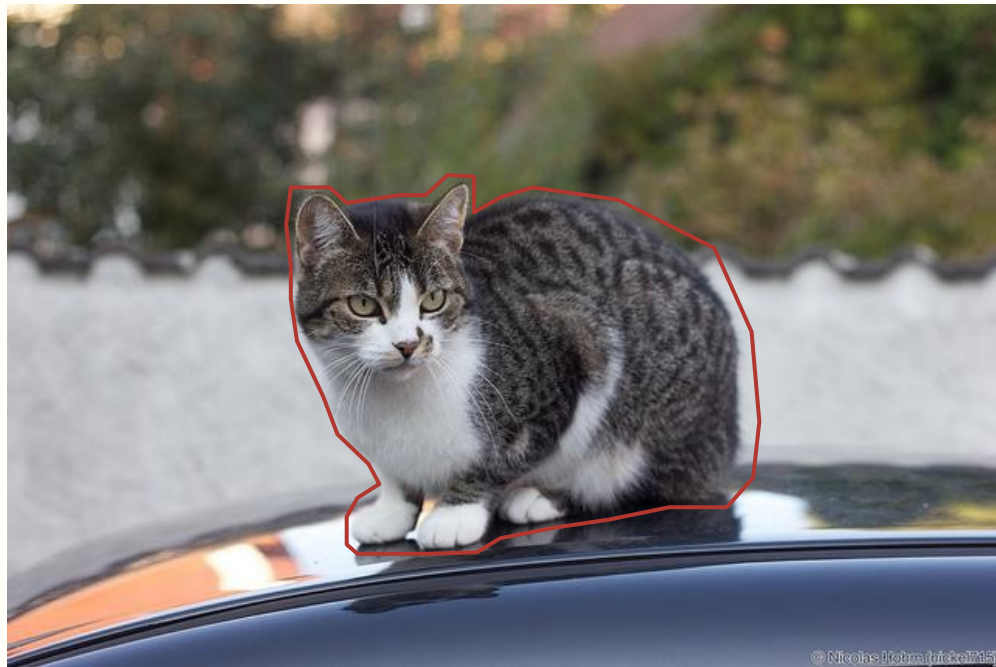
## シマウマ zebra



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000034 / COCO128-seg 公開変換注釈の輪郭を重ねて表示

# 猫 cat



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000650 / cat 注釈の輪郭。COCO128-seg 公開変換注釈。

## 街路の時計 clock



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000064 / COCO128-seg 公開変換注釈の輪郭を重ねて表示

# 重なり合うオートバイ motorcycle



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000073 / COCO128-seg 公開変換注釈の輪郭を重ねて表示

# 飛行機 airplane



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000081 / COCO128-seg 公開変換注釈の輪郭を重ねて表示

## ケーキとフォーク cake / fork



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000092 / COCO128-seg 公開変換注釈の輪郭を重ねて表示

# 野球の選手とバット person / baseball bat



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000192 / COCO128-seg 公開変換注釈の輪郭を重ねて表示

## スノーボードと遠方の人 person / snowboard



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000201 / COCO128-seg 公開変換注釈の輪郭を重ねて表示

## 前後に重なるゾウ elephant



[1] <https://github.com/ultralytics/assets/releases/download/v0.0.0/coco128-seg.zip>

COCO image ID 000000000263 / COCO128-seg 公開変換注釈の輪郭を重ねて表示

## 全 80 カテゴリ （ 1/4 ）

- person （ 1 カテゴリ ） : person
- vehicle （ 8 カテゴリ ） : bicycle, car, motorcycle, airplane, bus, train, truck, boat
- outdoor （ 5 カテゴリ ） : traffic light, fire hydrant, stop sign, parking meter, bench

## 全 80 カテゴリ （ 2/4 ）

- animal （ 10 カテゴリ ） : bird, cat, dog, horse, sheep, cow, elephant, bear, zebra, giraffe
- accessory （ 5 カテゴリ ） : backpack, umbrella, handbag, tie, suitcase
- sports （ 10 カテゴリ ） : frisbee, skis, snowboard, sports ball, kite, baseball bat, baseball glove, skateboard, surfboard, tennis racket

## 全 80 カテゴリ （ 3/4 ）

- kitchen （ 7 カテゴリ ） : bottle, wine glass, cup, fork, knife, spoon, bowl
- food （ 10 カテゴリ ） : banana, apple, sandwich, orange, broccoli, carrot, hot dog, pizza, donut, cake
- furniture （ 6 カテゴリ ） : chair, couch, potted plant, bed, dining table, toilet

## 全 80 カテゴリ（4/4）

- electronic（6 カテゴリ）：tv, laptop, mouse, remote, keyboard, cell phone
- appliance（5 カテゴリ）：microwave, oven, toaster, sink, refrigerator
- indoor（7 カテゴリ）：book, clock, vase, scissors, teddy bear, hair drier, toothbrush

## 画像・注釈・ソフトウェアの利用条件

- 注釈と Web サイトは COCO Consortium の CC BY 4.0 で提供される。
- 画像の著作権は撮影者側にあり、Flickr の画像ごとの利用条件に従う。
- 公式 API は BSD 形式の利用条件を持つ [1]。

[1] <https://cocodataset.org/#termsofuse>  
<https://github.com/cocodataset/cocoapi/blob/master/license.txt>

## Microsoft COCO: Common Objects in Context

Tsung-Yi Lin et al. (2014). European Conference on Computer Vision (ECCV), pp. 740–755.

COCO の原著。日常場面の画像収集、領域注釈と初期評価を報告する。

<https://arxiv.org/abs/1405.0312>

---

## COCO API

COCO Consortium (n.d.; 参照日 2026-10-15). GitHub, cocodataset/cocoapi.

画像・注釈の読み込み、マスク表現、検出・分割の評価を担う公式実装。

<https://github.com/cocodataset/cocoapi>

## COCO-Stuff: Thing and Stuff Classes in Context

Holger Caesar, Jasper Uijlings, Vittorio Ferrari (2018). IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1209–1218.

COCO へ背景の画素注釈を加え、物体と背景を含む場面理解を扱う研究。

<https://arxiv.org/abs/1612.03716>

---

## Panoptic Segmentation

Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, Piotr Dollár (2019). IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9404–9413.

意味的領域分割と個体ごとの分割を統合し、共通指標 PQ を提案する。

<https://arxiv.org/abs/1801.00868>

## Mask R-CNN

Kaiming He, Georgia Gkioxari, Piotr Dollár, Ross Girshick (2017). IEEE International Conference on Computer Vision (ICCV), pp. 2961–2969.

物体ごとのマスク予測を検出器に加えた、COCO の代表的な分割モデル。

<https://arxiv.org/abs/1703.06870>

---

## LVIS: A Dataset for Large Vocabulary Instance Segmentation

Agrim Gupta, Piotr Dollár, Ross Girshick (2019). IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5356–5364.

COCO 画像に大語彙の領域注釈を加え、まれなカテゴリの認識を評価する。

<https://arxiv.org/abs/1908.03195>