
大規模言語モデルの展開
マルチモーダルへ

大規模言語モデルの展開
-- マルチモーダルへ --
Agenda

- はじめに
- **Part 1** 画像認識とAttention
  - 画像認識の課題とWindows Sliding
  - Caption生成の試みとAttention
- **Part 2** Vision Transformer：Inductive Bias Free
  - Vision Transformer の画像embedding
  - Vision Transformer 内部表現の分析

大規模言語モデルの展開
-- マルチモーダルへ --
Agenda

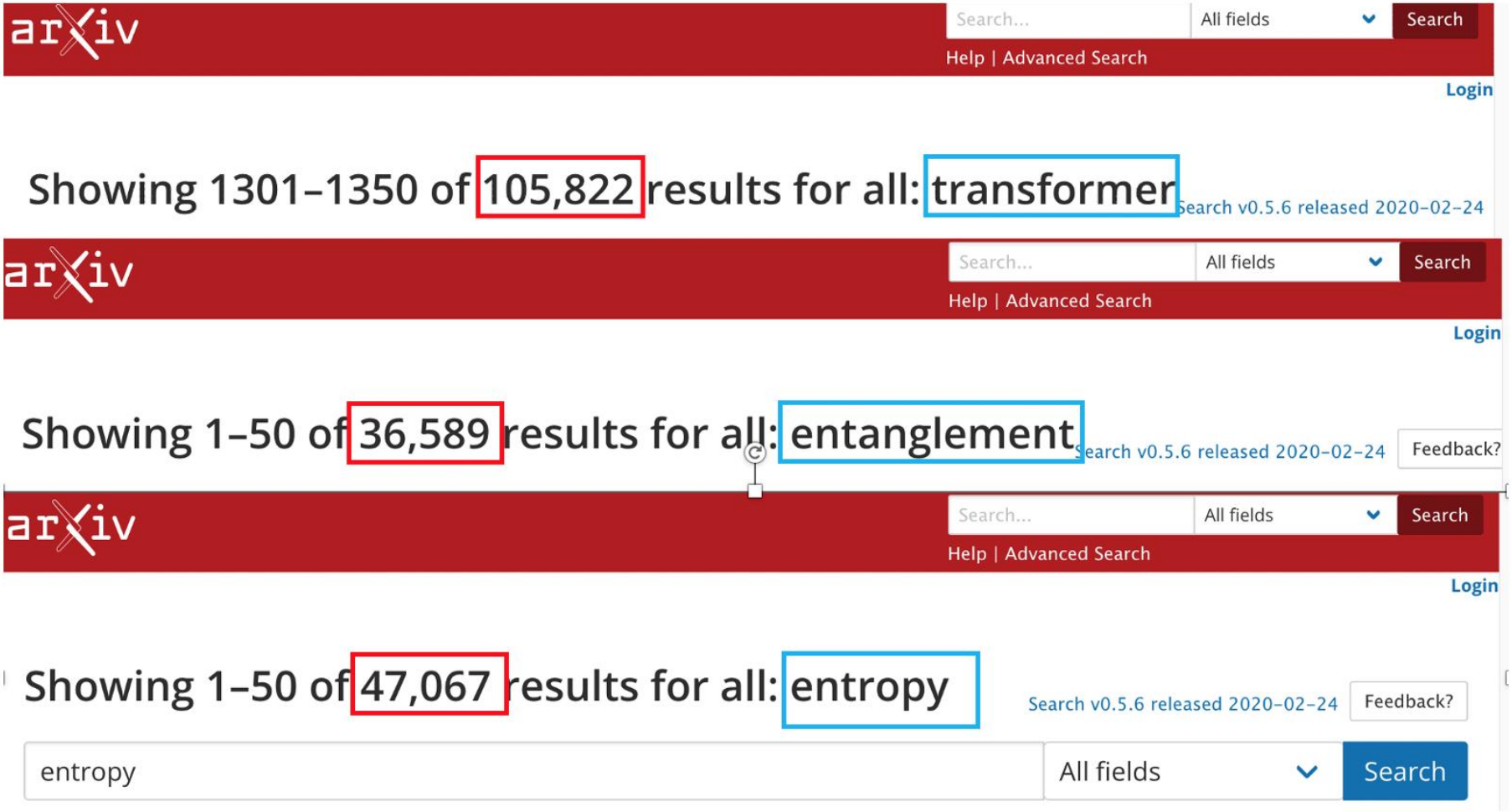
- **Part 3** CLIP: Connecting text and images
  - CLIPのアプローチ：Natural Language Supervision
  - CLIPのデータセットと予測サンプル
  - CLIP：Contrastive Representation Learning

はじめに

セミナーで考えたいこと
急激な変化

ChatGPTの急速な普及を転換点として、かってない規模とエネルギーで、多くの研究者・開発者・企業が人工知能の分野に参入しようとしています。

arXivへの投稿数


arXiv
Search...
All fields
Search
Help | Advanced Search
Login
Showing 1301–1350 of 105,822 results for all: transformer
Search v0.5.6 released 2020-02-24
arXiv
Search...
All fields
Search
Help | Advanced Search
Login
Showing 1–50 of 36,589 results for all: entanglement
Search v0.5.6 released 2020-02-24
Feedback?
arXiv
Search...
All fields
Search
Help | Advanced Search
Login
Showing 1–50 of 47,067 results for all: entropy
Search v0.5.6 released 2020-02-24
Feedback?
entropy
All fields
Search

セミナーで考えたいこと
技術的背景と現実的な技術的焦点

次回のセミナーでは、第一に、現在進行中のこの変化がどのような技術的背景を持つのかを考えてみたいと思っています。

第二に、現時点での現実的な技術的焦点がどの辺にあるのかを考えようと思います。

大規模言語モデルの展開

第一点の現在の急激な変化の技術的背景についてですが、僕は、次のように考えています。

それは、自然言語処理だけではなく、コード生成、視覚情報の処理、分子構造と反応のモデリング等の様々な領域においても、大規模言語モデルが極めて優秀な能力を発揮できることが明らかになったことだと思います。

登場しつつある新しい人工知能技術が、現在の自然言語ベースの大規模言語モデルを超えるものだというイメージを持っている人も少なくないと思いますが、それは少し違うと思います。現在の展開には、技術的連続性があります。セミナーのタイトルを、「大規模言語モデルの展開」としたのはそのためです。

現在の技術的焦点
Multimodalな人工知能へ

もっとも、技術に連続性があると言っても、技術は変化します。現時点での技術的焦点は何かを考えることは大事なことです。

僕は、それは「テキストの世界とイメージの世界の統合」だと考えています。

OpenAIのGPT-4でのMultimodalな機能の追加は、とても印象的なものでした。Googleも、それに追従しようとしています。

今回のセミナーでは、人工知能技術の現在の技術的焦点の一つが、「Multimodalな人工知能」にあると考えて、その分野でのいくつかの基本的な技術を紹介しようと思います。

セミナーで取り上げる
二つのプロジェクト

今回のセミナーでは、大規模言語モデルのマルチモーダルへの展開として、主要に次の二つのプロジェクトを取り上げます。

- GoogleのVision Transformer
- OpenAIのCLIP

AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE

Alexey Dosovitskiy[*,†], **Lucas Beyer**[*], **Alexander Kolesnikov**[*], **Dirk Weissenborn**[*], **Xiaohua Zhai**[*], **Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby**[*,†]

[*]equal technical contribution, [†]equal advising

Google Research, Brain Team

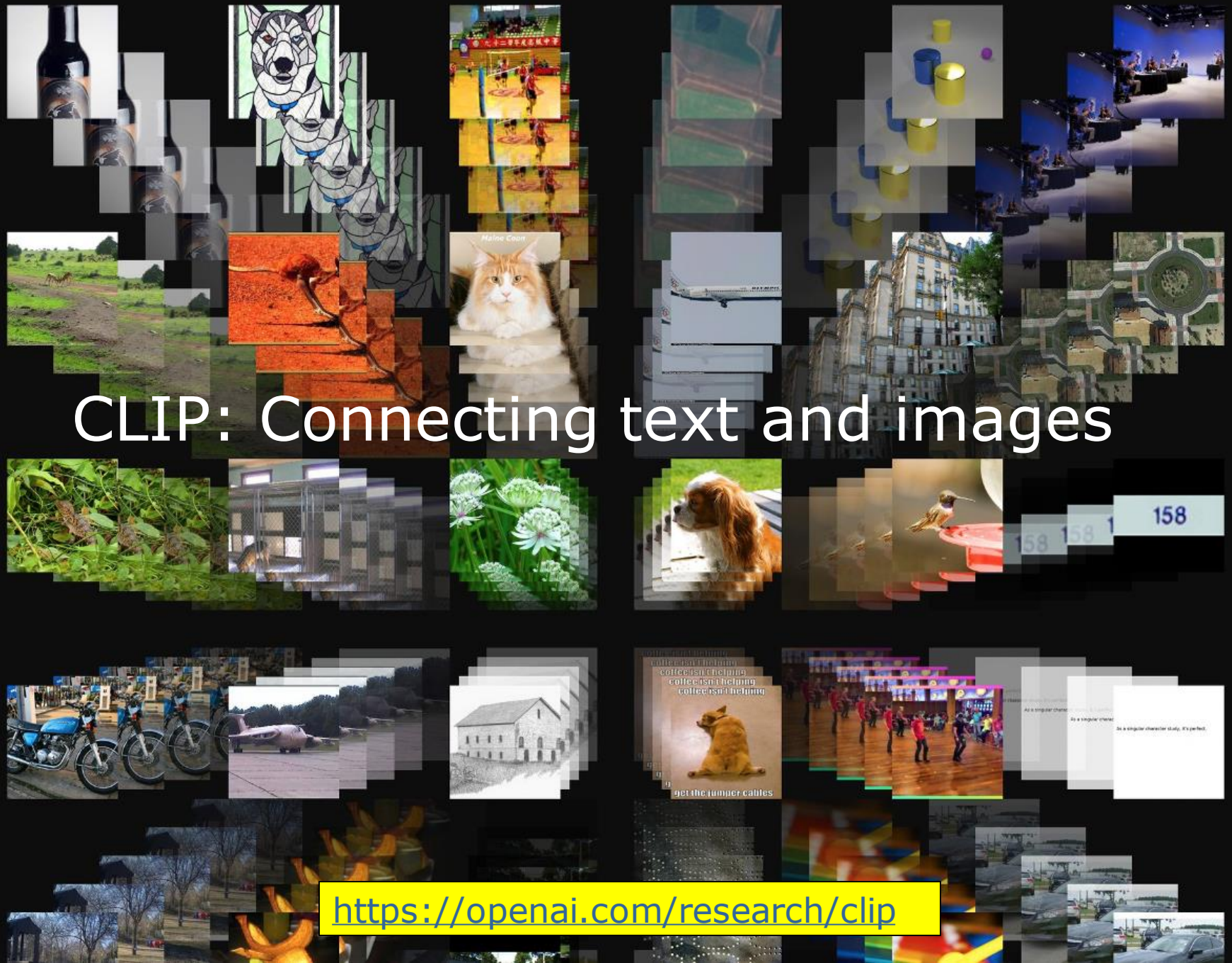
{adosovitskiy, neilhoulsby}@google.com

ABSTRACT

While the Transformer architecture has become the de-facto standard for natural language processing tasks, its applications to computer vision remain limited. In vision, attention is either applied in conjunction with convolutional networks, or used to replace certain components of convolutional networks while keeping their overall structure in place. We show that this reliance on CNNs is not necessary and a pure transformer applied directly to sequences of image patches can perform very well on image classification tasks. When pre-trained on large amounts of data and transferred to multiple mid-sized or small image recognition benchmarks (ImageNet, CIFAR-100, VTAB, etc.), Vision Transformer (ViT) attains excellent results compared to state-of-the-art convolutional networks while requiring substantially fewer computational resources to train.[1]

https://arxiv.org/pdf/2010.11929.pdf

こうした動きの先駆となった、Transformerのエンジンで画像認識も可能であることを明らかにした、Googleの論文



CLIP: Connecting text and images

https://openai.com/research/clip

新しいアプローチに注目

重要なことは、これらのプロジェクトの中で、AIに対する新しいアプローチが生まれていることです。セミナーでは、次のようなアイデアに注目して、その概要を紹介したいと思います。

- Inductive Bias Free
- Natural Language Supervision
- Contrastive Representation Learning

あらためて「言語モデル」の意味を考える

少し前までは、自然言語の意味理解にはRNNが、画像認識にはCNNが主なエンジンとして使われていました。

上に見たような多様な応用への展開が、大規模言語モデルの飛躍を生み出したTransformer という一つのエンジンで可能になるかもしれないというのは、確かに魅力的なビジョンです。

ただ、それほど簡単な話ではないのだということも、今回のセミナーで伝えたいことです。

多様で大きな変化が起きつつあるのですが、同時に、それらの変化のドライビング・フォースが何かを見極めることは重要だと考えています。

僕は、その中心的な力は、大規模言語モデルが採用した「意味の分散表現」だと考えています。

あらためて「大規模言語モデル」の意味を考えることは大事なことです。それについては、今回のセミナーとは別に整理していきたいと考えています。

OpenAIのマルチモーダル・サービスの開始

現時点(2023/09/25)での嬉しいニュースの一つは、OpenAIが ChatGPTのマルチモーダル・サービスの公開を予告したことです。

僕は、それは、AI利用の拡大の新しいゲーム・チェンジャーになると考えています。(ChatGPT等の新しいAI技術の利用者は、多く見積もっても数億人で、それはスマホの利用者よりはるかに少ないのです。)

AI利用の拡大は、今回のセミナーが取り上げたテキストと画像の統合が主要な舞台となるというよりは、AIの入力と出力が音声で可能になるというのが大きいのではと考えています。自然言語というのは、もともとは、「文字」ではなく、音声で伝えられるものです。

僕は、長い目で見たとき、大規模言語モデルの人間にとっての最大の貢献は、人間にことばの壁を乗り越える現実的手段を初めて提供したことにあると考えています。それは、「幻想」に悩まされることもない、大規模言語モデルの最良の能力を利用することでもあります。

GPT-4

MA

GPT-4は、画像の入力を受け付けていますか？

2023/09/13

いいえ、GPT-4はテキストベースのモデルであり、直接的に画像の入力を受け付けることはできません。私はテキスト情報のみを処理し、応答します。画像の内容を分析するための専用のモデルやフレームワークも存在しますが、GPT-4自体はそれを持っていません。しかし、テキストに関する質問や情報のリクエストに応答することは得意ですので、どんな質問でもお気軽にどうぞ！

ChatGPT can now see, hear, and speak

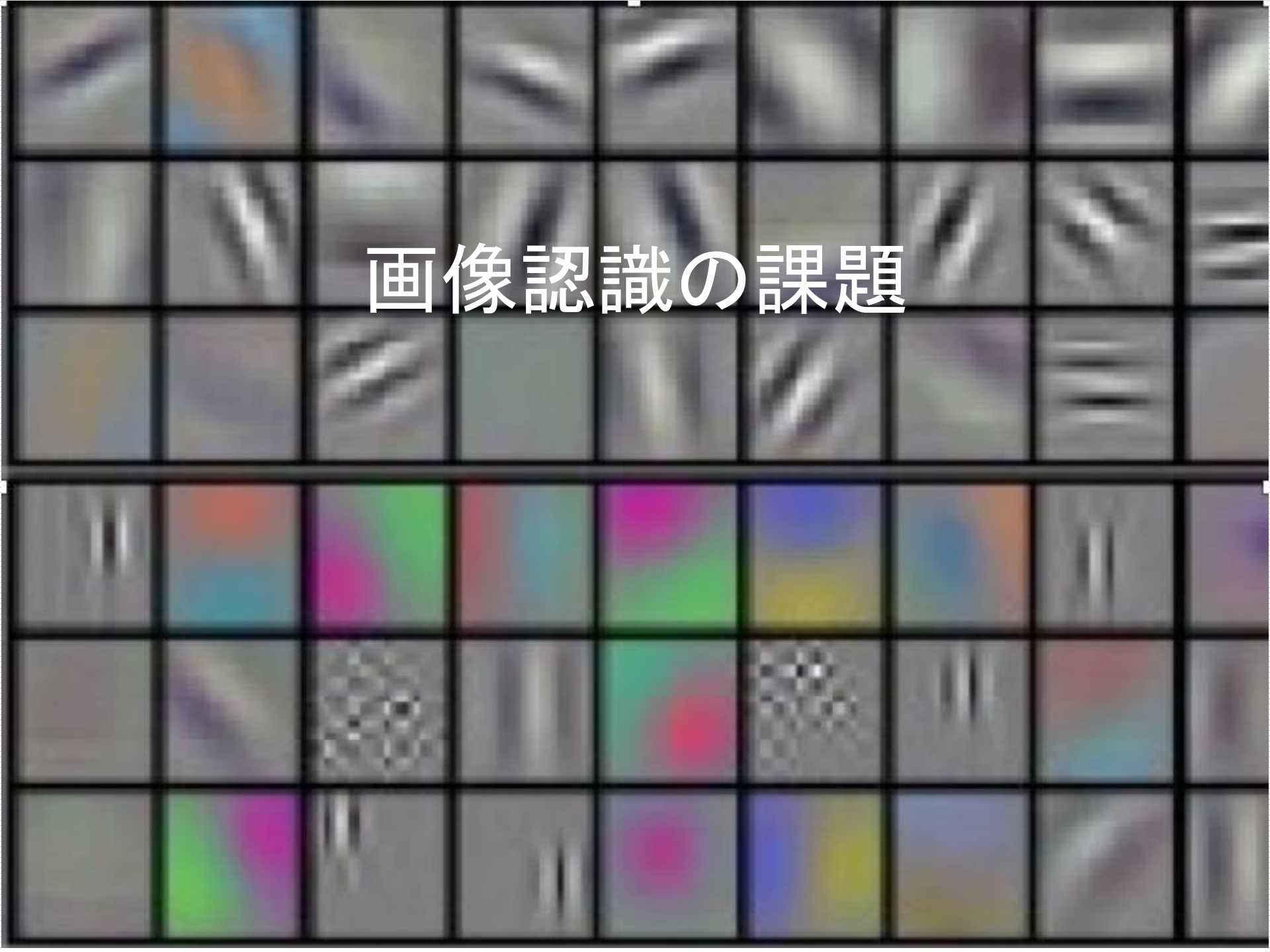
2023/09/25

We are beginning to roll out new voice and image capabilities in ChatGPT. They offer a new, more intuitive type of interface by allowing you to have a voice conversation or show ChatGPT what you're talking about.

https://openai.com/blog/chatgpt-can-now-see-hear-and-speak


Adobe Firefly


Part 1
画像認識とAttention


画像認識の課題

画像認識技術の課題

画像認識では、次のような技術が求められます。

- オブジェクトが一つの場合
  - オブジェクトのカテゴリーの認識（分類）
  - オブジェクトの位置の認識
  - オブジェクトの切り出し

- オブジェクトが複数の場合
  - 複数のオブジェクトのカテゴリーの認識（分類）
  - 複数のオブジェクトの位置の検出
  - 複数オブジェクトの切り出し

単一オブジェクトの場合

カテゴリー分類

CAT

単一オブジェクトの場合

カテゴリー分類

CAT

カテゴリー分類と
位置決め

CAT

単一オブジェクトの場合

カテゴリー分類

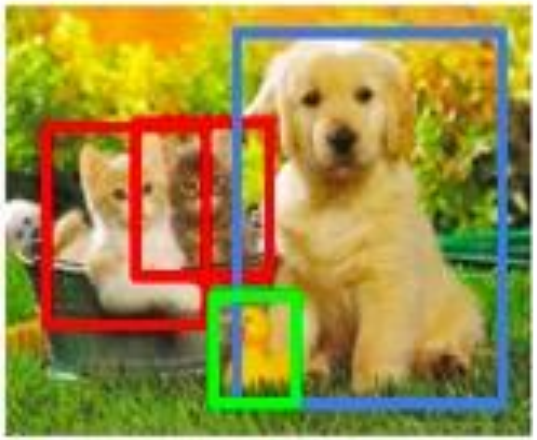
カテゴリー分類と
位置決め

単一のオブジェクト

オブジェクトの認識と位置の検出

複数オブジェクトの場合

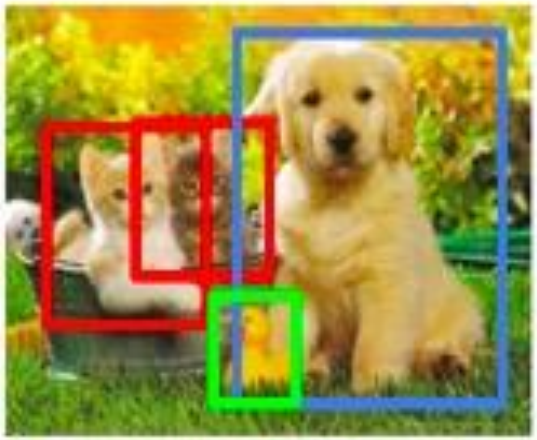
オブジェクトの検出



CAT, DOG, DUCK

複数オブジェクトの場合

オブジェクトの検出



CAT, DOG, DUCK

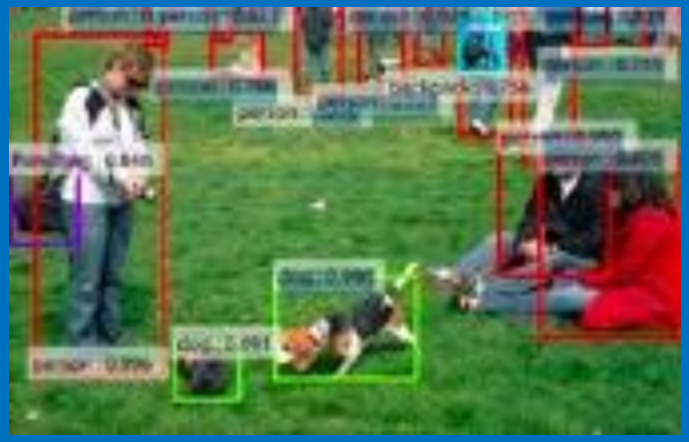
インスタンスの
切り出し

CAT, DOG, DUCK

複数オブジェクトの場合

複数オブジェクトの認識とインスタンスの切り出し

オブジェクトのカテゴリーと位置の検出



オブジェクトの位置の検出

基本的には、元の画像より少し大きな画像の中で、ウインドウの位置・大きさを変えてスライドさせて、最適な認識結果を返すものを探す。Windows Sliding

Windows Sliding

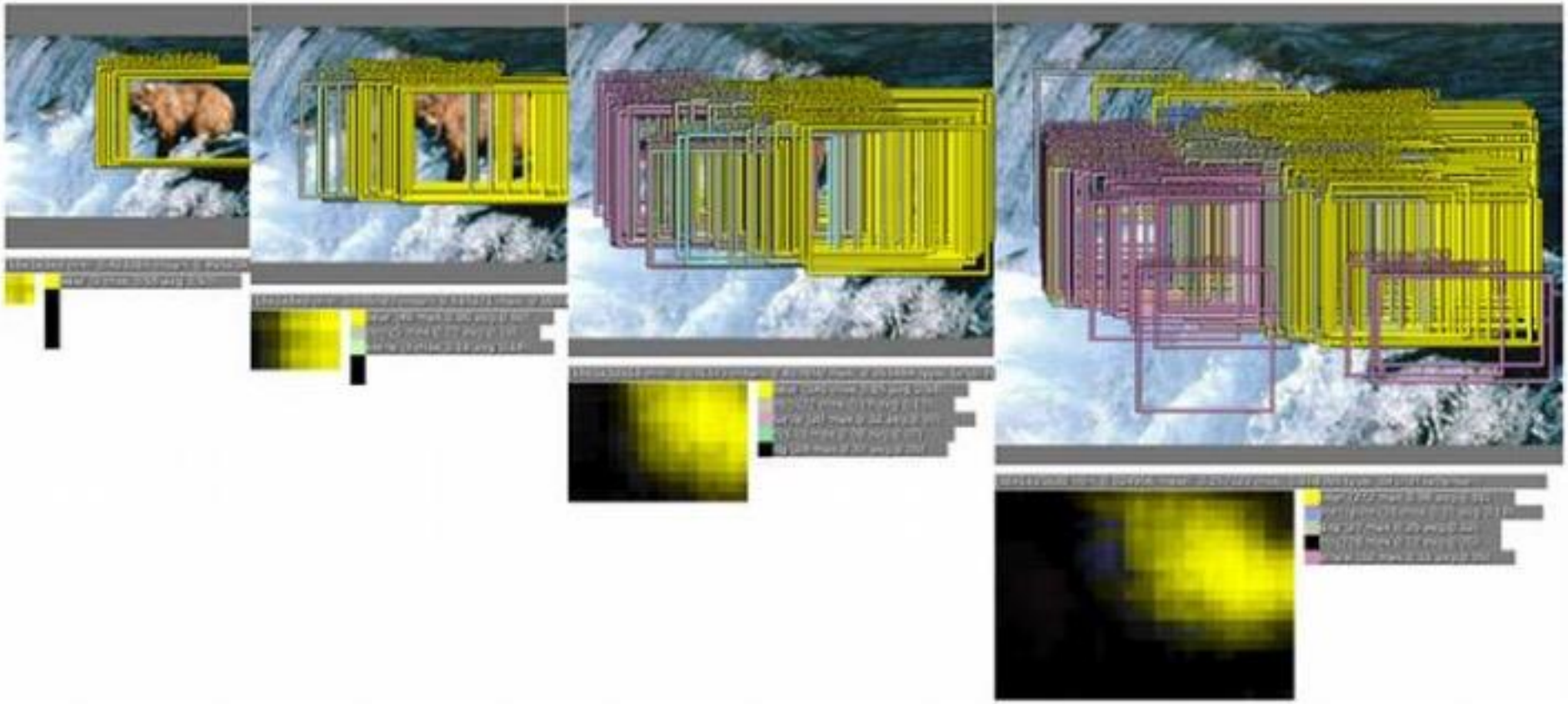
OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks

Pierre Sermanet et al.
https://arxiv.org/pdf/1312.6229.pdf

2014年

分類器/検出器は、ウィンドウをスライドさせながら、
各位置ごとにオブジェクトのクラスとその信頼度を出力する。

これらの予測の解像度は、第3節で説明した方法で
向上させることができる。



そして回帰計算で、各ウィンドウに対するオブジェクトの
位置スケールを予測する。

これらの境界ボックスはマージされ、
少数のオブジェクトに蓄積される。

複数オブジェクトの検出

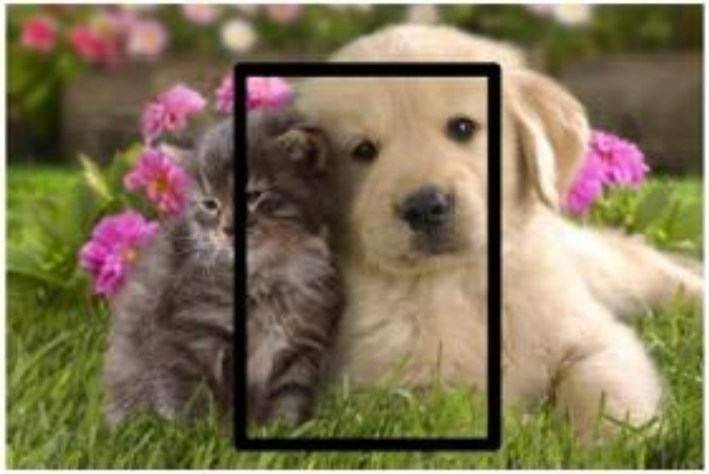
CAT & DOG

CAT? NO

DOG? NO

CAT? YES!

DOG? NO



CAT? NO

DOG? NO

複数オブジェクトの位置検出

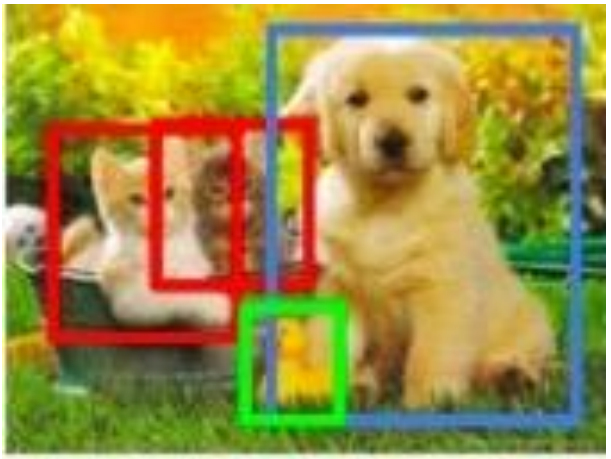
DOG, (x, y, w, h)
CAT, (x, y, w, h)

= 8 numbers

複数オブジェクトの位置検出

DOG, (x, y, w, h)
CAT, (x, y, w, h)

= 8 numbers



CAT, DOG, DUCK

DOG, (x, y, w, h)
CAT, (x, y, w, h)

= 8 numbers

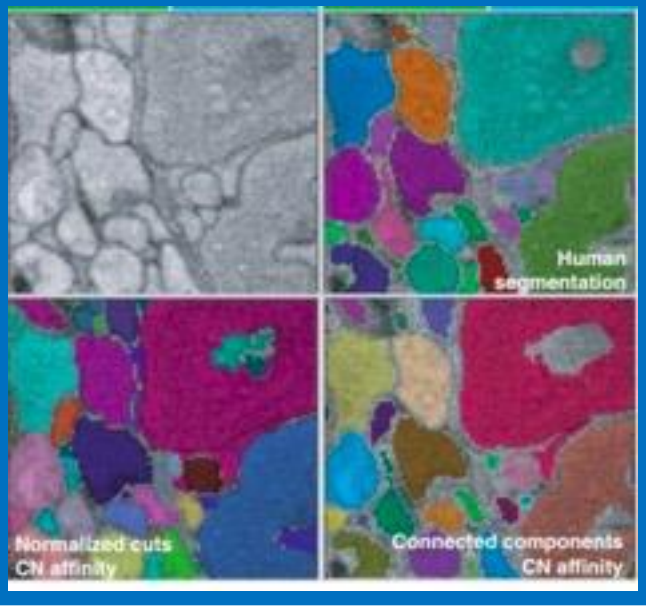
CAT, (x, y, w, h)
CAT, (x, y, w, h)
....
CAT (x, y, w, h)

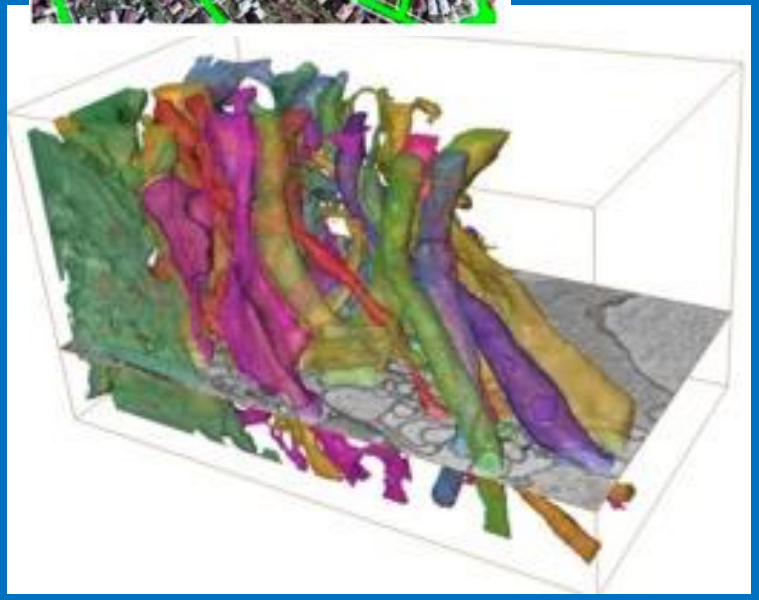
= many numbers

問題点： 非常に多くの位置とスケールをテストする必要がある。

解決策： マシンが早かったら、それを全部やる！

オブジェクトの切り出し (segmentation)

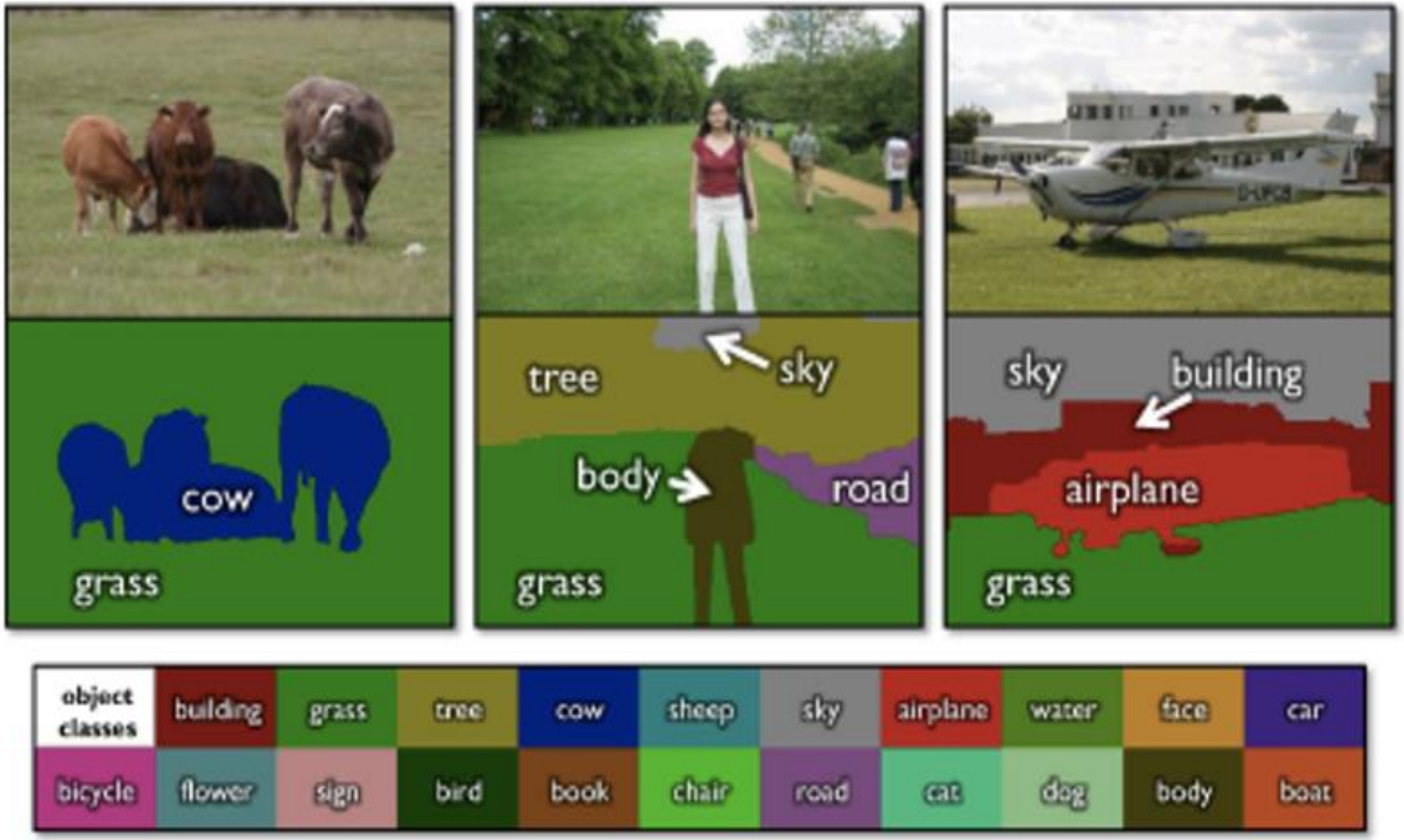




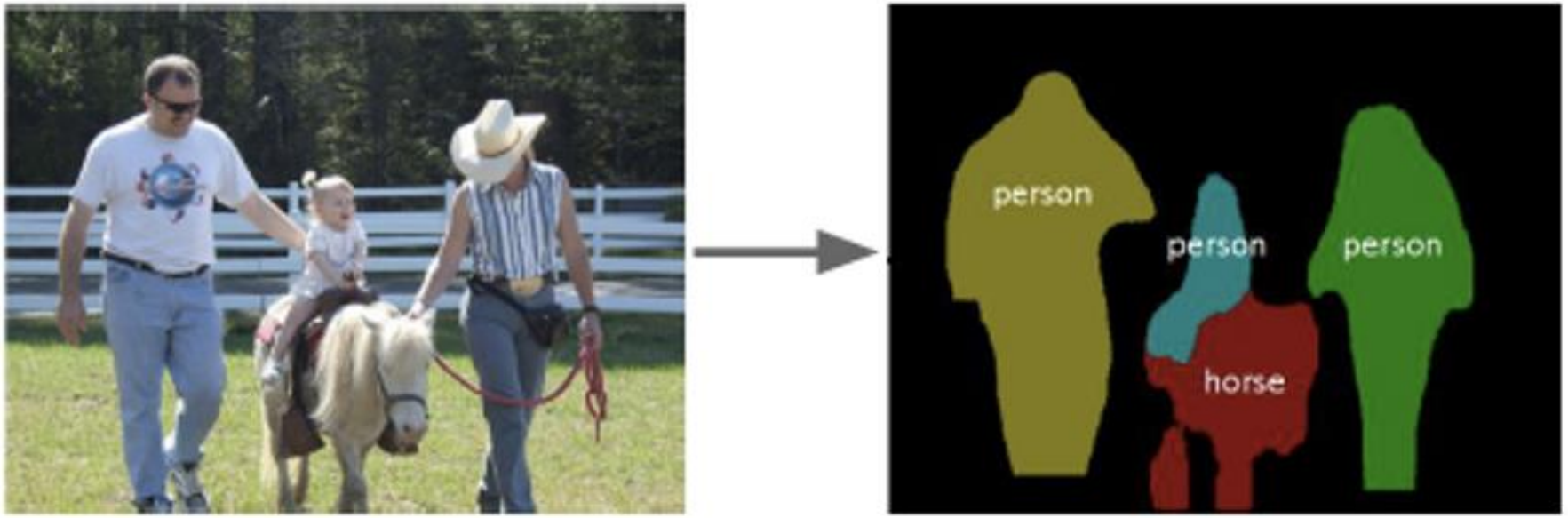
Semantic SegmentationとInstance Segmentation

- Semantic Segmentation
  - 全てのピクセルにラベルをつける。
  - インスタンスごとに区別しない。
  - 古くからあるコンピュータ・ビジョンの問題。
- Instance Segmentation
  - インスタンスを検出して、分類して、ピクセルにラベルをつける
  - 同時検出、切り出し。
  - 比較的最近の技術。(MS COCO)

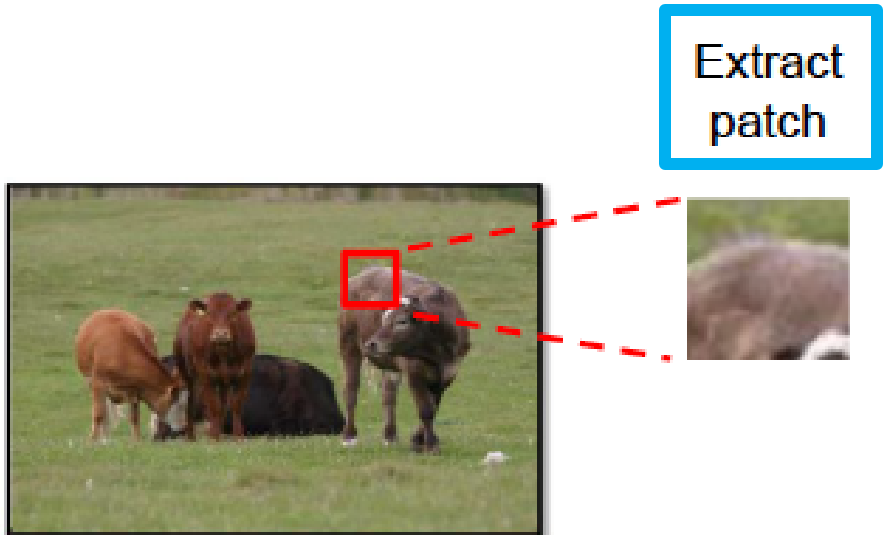
Semantic Segmentation



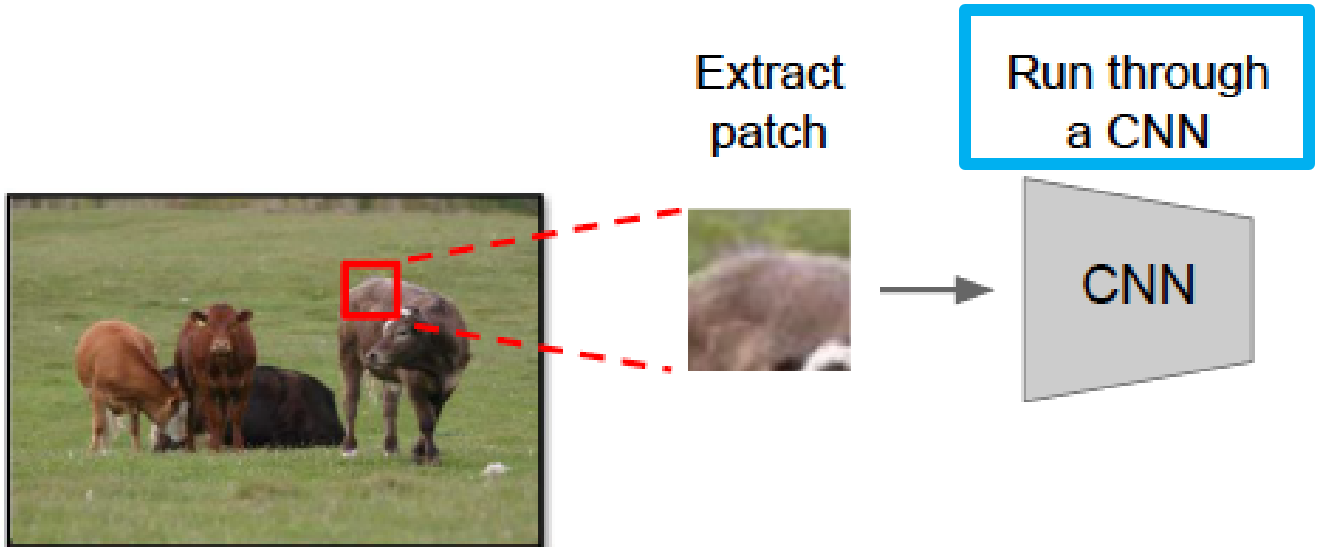
Instance Segmentation



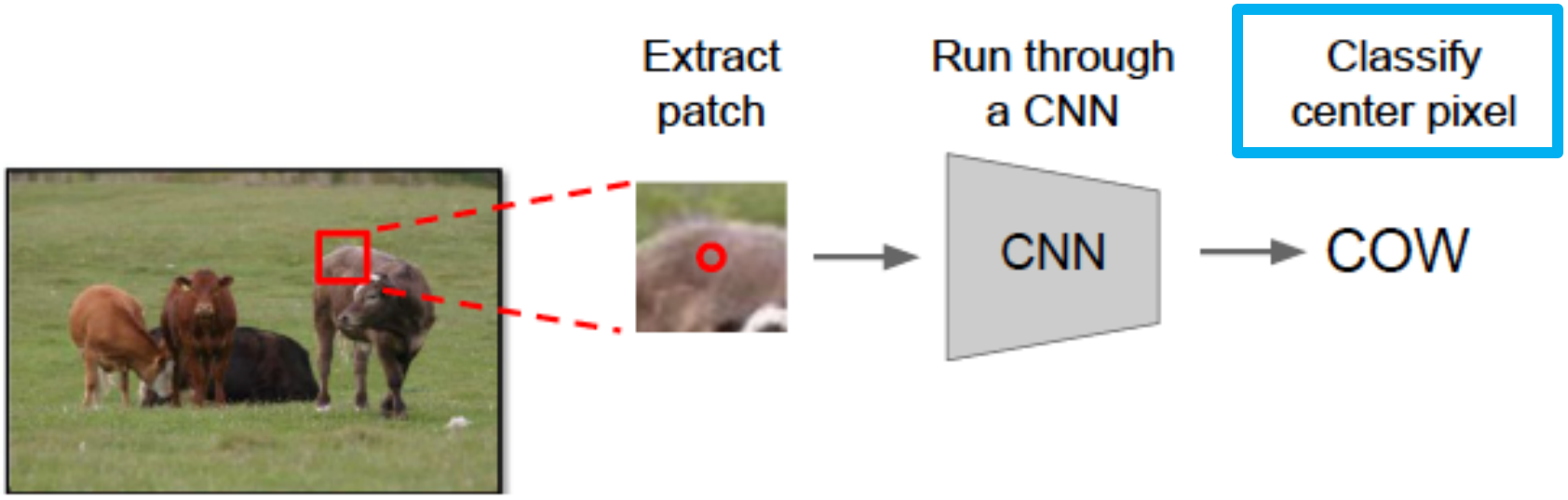
Semantic Segmentation



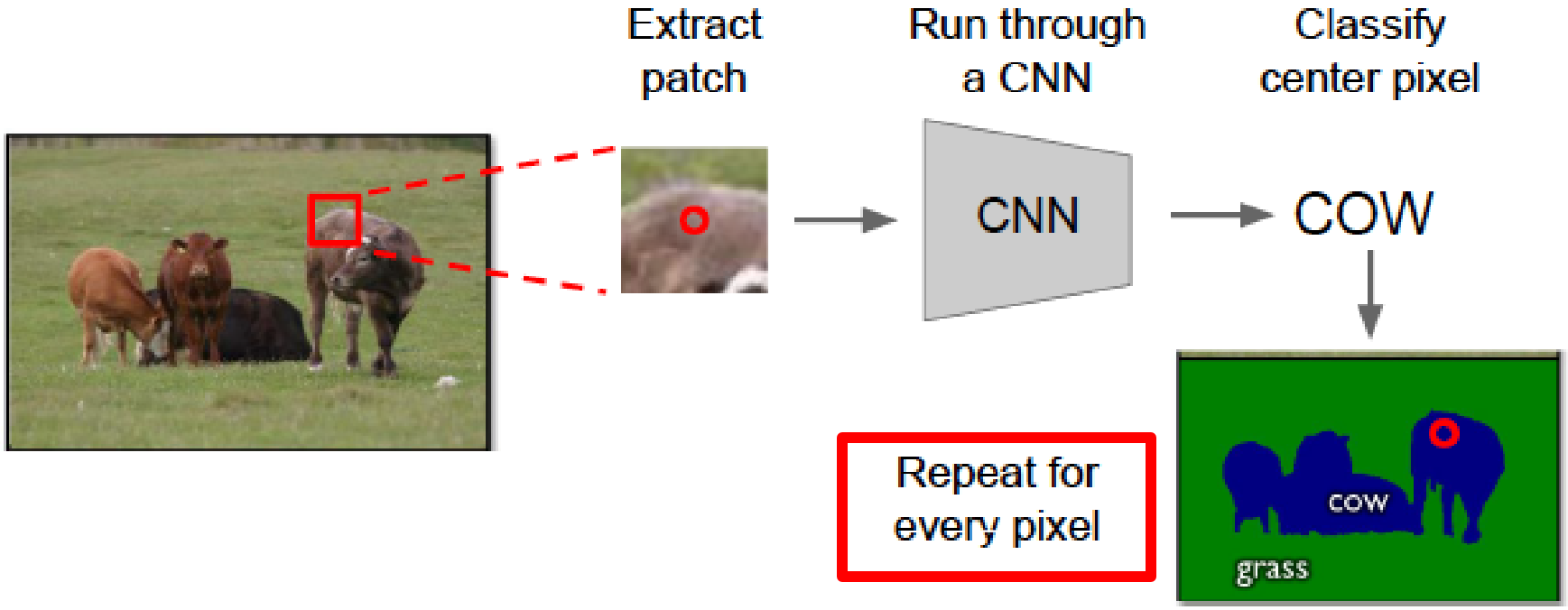
Semantic Segmentation



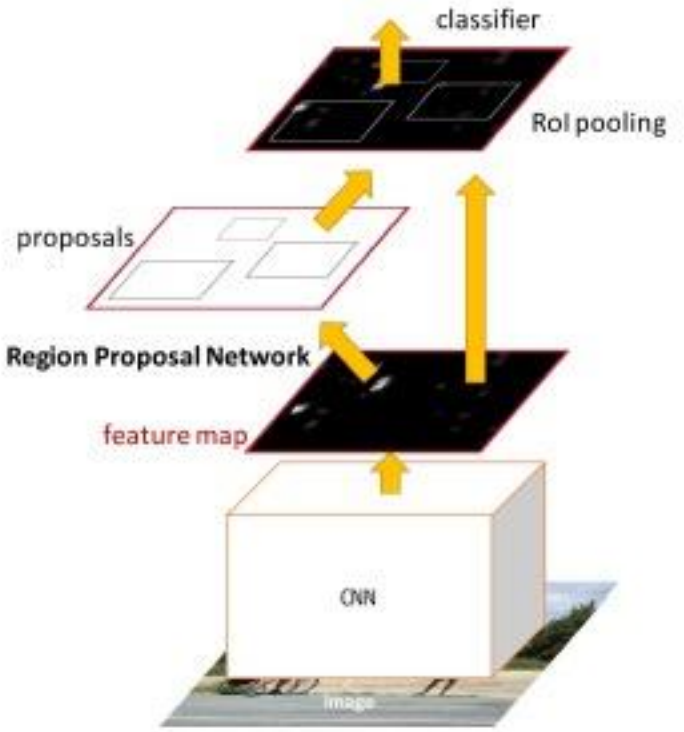
Semantic Segmentation



Semantic Segmentation



Faster R-CNN:



Insert a **Region Proposal Network (RPN)** after the last convolutional layer

RPN trained to produce region proposals directly; no need for external region proposals!

After RPN, use RoI Pooling and an upstream classifier and bbox regressor just like Fast R-CNN

Ren et al, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", NIPS 2015

Slide credit: Ross Girschick

R-CNN(Rは、RecurrentのRではなく、
Region Proposal NetworkのRである）
これ以上のSegmentationの紹介は、割愛する。


Caption生成の試みとAttention
“A herd of giraffes walk down the street in the middle of some trees.”

誰がAttentionを考えたのか？

現代の大規模言語モデルのエンジンは、「意味の分散表現」と「Attention Mechanism」を中核とするTransformerなのですが、Attention Mechanism を提唱した最初の論文は、2016年のBahdanauらによる次の論文です。

"Neural machine translation by jointly learning to align and translate"

https://arxiv.org/pdf/1409.0473.pdf

Transformerの「祖型」

このアーキテクチャーは、Googleの「ニューラル機械翻訳システム」に直ちに取り入れられ、機械翻訳のブレークスルーを引き起こしました。

これらの二つのシステムは、いずれも、RNNをエンジンとするものでしたが、「意味の分散表現」と「Attention Mechanism」を技術的中核とする点では、現代のTransformer エンジンと大きな共通点があります。

この二つのシステムを、Transfomerの「祖型」と考えることができると僕は考えています。

Attention Mechanism

Neural machine translation by jointly learning to align and translate

Bahdanau, D., Cho, K., and Bengio, Y

https://arxiv.org/pdf/1409.0473.pdf

2016年

論文の概要

近年、ニューラル機械翻訳として提案されたモデルは、多くの場合、Encoder-Decoderのファミリーに属している。そこでは、ソースの文が固定長ベクトルにエンコードされ、そこからデコーダが 翻訳文を生成する。

この論文では、固定長ベクトルの使用が、この基本的なEncoder/Decoderアーキテクチャの性能を改善する上でのボトルネックになっていると推論し、モデルに自動的に、ターゲット・ワードを予測するのに重要なソース・文の一部分について、（ソフト）検索を可能とすることによって、これを拡張することを提案する。

その際、これらの部分を明示的にハードセグメントとして形成する必要はない。

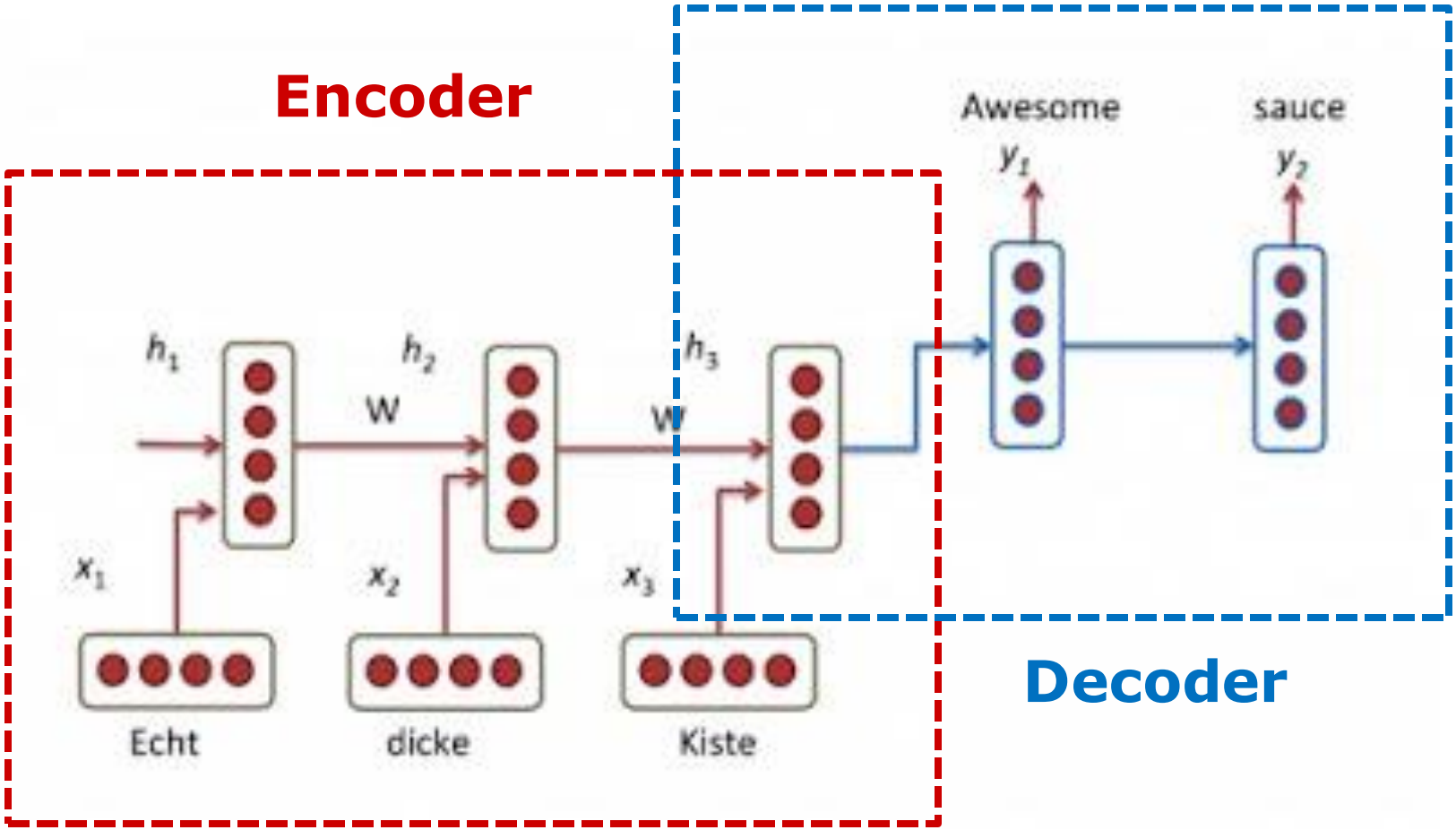
固定長ベクトルがボトルネック

先に見た、Ilya Sutskever らの翻訳システムでは、翻訳さるべき文は、Encoderで、一旦、ある決まった大きさの次元（例えば8000次元）を持つベクトルに変換される。このベクトルからDecoderが翻訳文を生成する。

入力された文が、長いものであっても短いものであっても、中間で生成され以降の翻訳プロセスすべての出発点となるこのベクトルの大きさは同じままだ。このシステムでは、長くても短くても入力された文全体が、一つの固定長のベクトルに変換されるのだ。

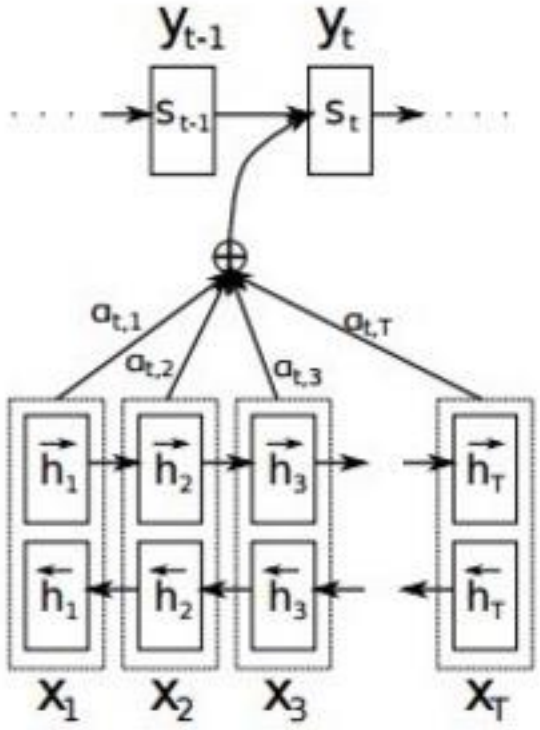
確かに、そこは翻訳の精度を上げる上でのボトルネックになりうる。事実、Ilya Sutskever らのシステムでは、文の長さが長くなるにつれて、翻訳の精度が低下されるのが観察されるという。

次の図(https://goo.gl/JGckBP から)は、こうしたメカニズムで、RNNが、独文の "Echt dicke Kiste" を英文の"Awesome sauce" に翻訳する様子を表している。(ここでは、文章の終わりを表す<EOS>は、省略されている)



この論文の基本的アイデア

文全体に一つの固定長のベクトルを割り当てるのではなく、翻訳時に、ソース・文の一部分を 改めて見直して、その部分から提供される情報を翻訳に生かそうということだ。



$a_{3,2}$が大きい場合、これは、Decorderがターゲット文の第3の単語を生成しながら、ソース文の第2の状態に多くの注意を払うことを意味する。

「ここで、yはデコーダによって生成された翻訳された単語であり、xは原文の単語である。上記の図は双方向のリカレント・ネットワークを使用しているが、それは重要ではない。逆方向は無視していい。

重要な部分は、各デコーダの出力するワード y_tが、Encoderの最後の状態だけでなく、すべての入力状態の重みづけられた結合に依存することである。

aは、出力ごとに、それぞれの入力状態をどの程度考慮されるべきかを定義する重みである。したがって、$a_{3,2}$が大きい場合、これは、Decorderがターゲット文の第3の単語を生成しながら、ソース文の第2の状態に多くの注意を払うことを意味する。

aは、通常、1に合計されるように正規化される（それらは、入力状態に対する確率分布である）。」

Google ニューラル機械翻訳

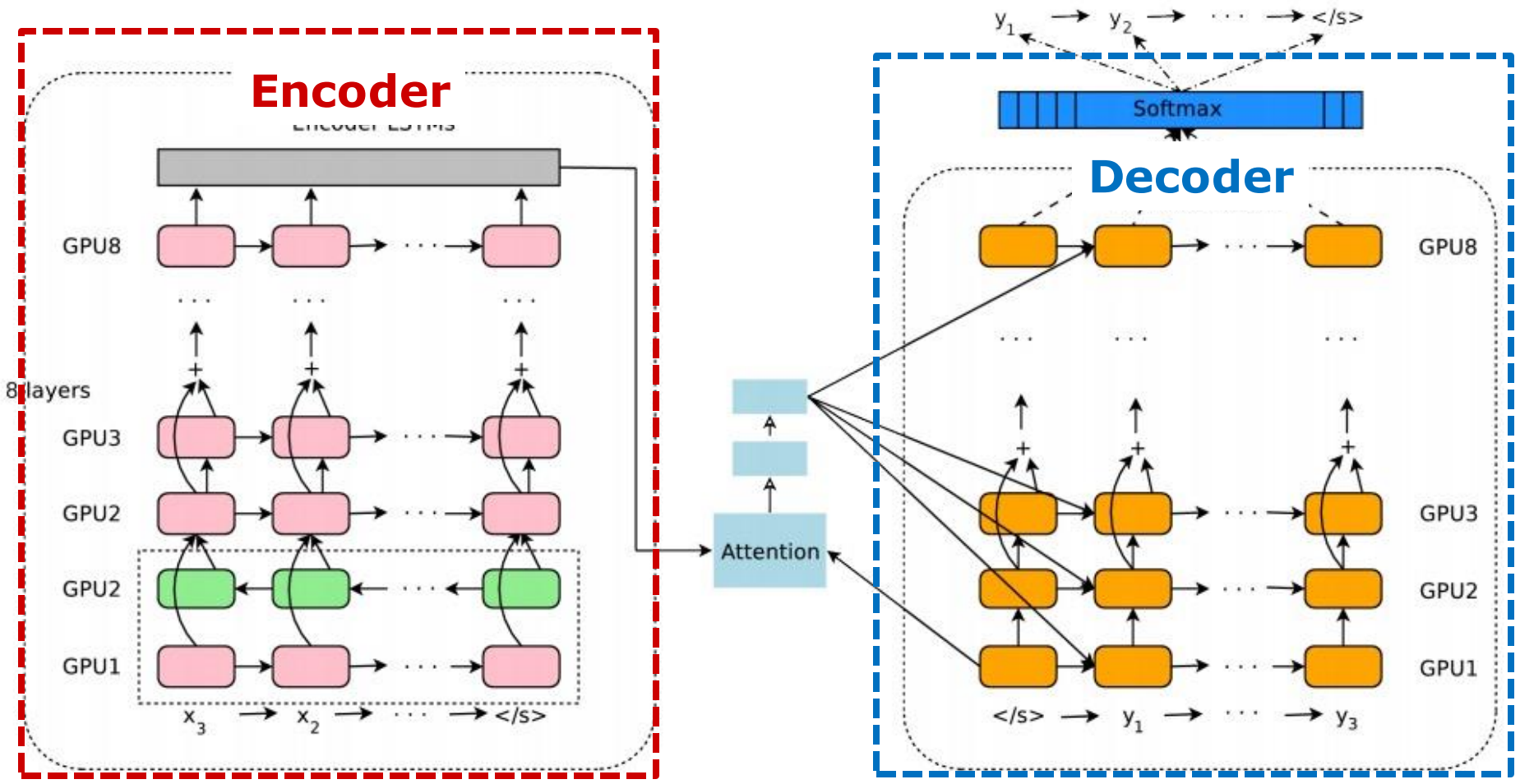
Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation

Yonghui Wu et al.

https://arxiv.org/pdf/1609.08144.pdf

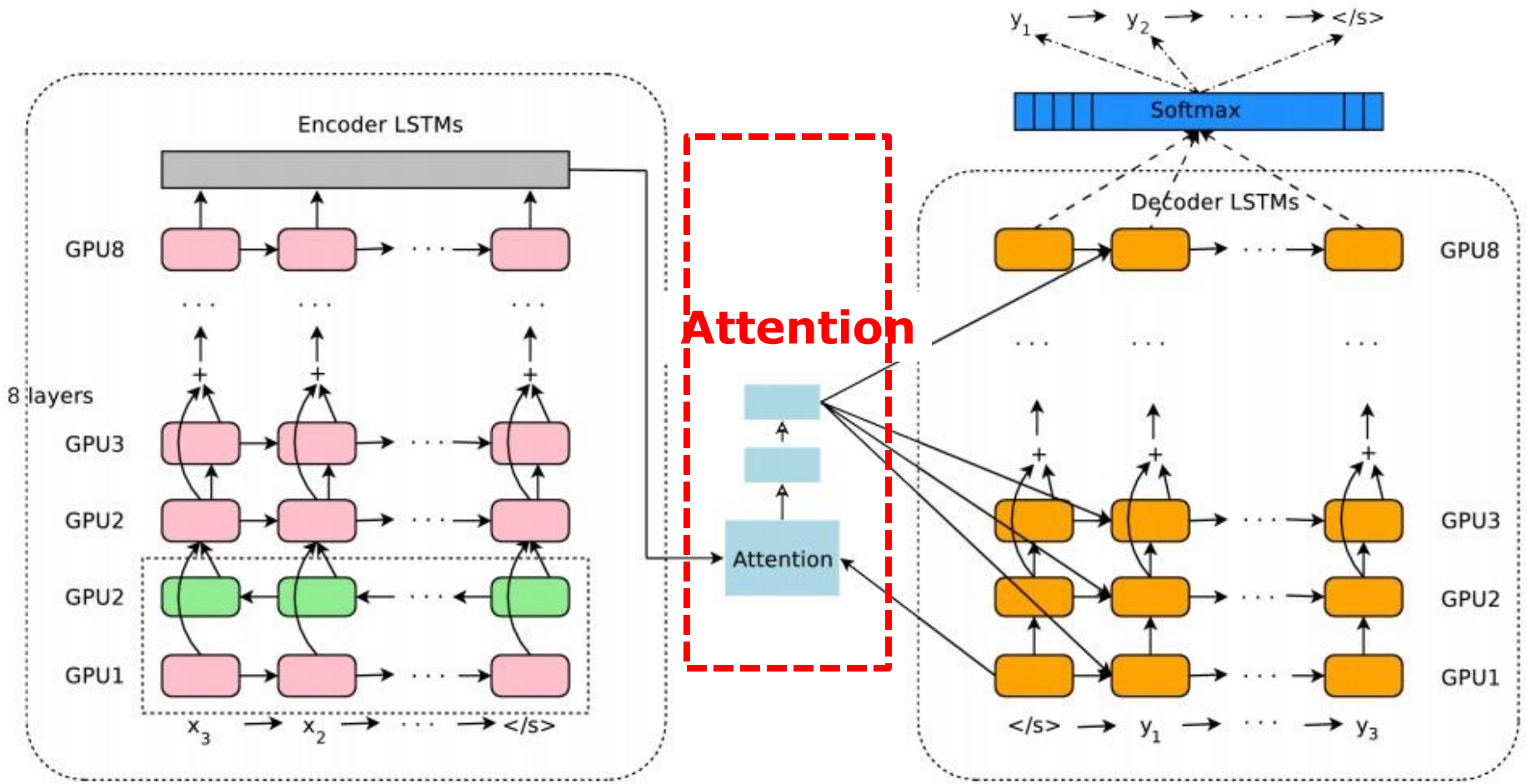
2016年

Encoder / Decoder



左側に、LSTMを8段重ねにした Encoder LSTMがあり、
右側には、同じくLSTMを8段重ねにした Decoder LSTMがある。

Attention Mechanism



EncoderとDecoderの中間に、Attentionと記された領域がある。
ここからの出力Attention Context は、Decoderのすべての
ノードに供給されている。

実は、まだ先があった

実は、2016年のBahdanauらの論文より前に、Attentionの重要性を指摘した論文があるのです。それは、2015年のKelvin Xu らの次の論文です。

"Show, Attend and Tell: Neural Image Caption Generation with Visual Attention"

http://arxiv.org/pdf/1502.03044v2.pdf

BahdanauもKelvin Xuも、Bengioの研究グループに属する人で、Attentionについてのこの二つの先駆的な論文には、いずれにも、Bengio が Last Author として名を連ねています。

画像に対するAttention

興味深いことは、ここで提唱されているのは、画像に対するAttentionを利用することで、画像からCaptionを生成することができるというシステムでした。

このシステムは、画像認識のエンジンにCNNを利用し、Caption生成にはRNNをエンジンとして利用するという混合エンジンのシステムでしたが、二つのエンジンの間をつなぐのが、Attentionでした。

簡単にいうと、画像のある部分にAttentionを固定した時（それは文字通りあるオブジェクトに「注意」を集中することです)、そのオブジェクトに対応する単語を生成するというものです。視点が移動するにつれて、Captionが生成されることになります。

Visual Attention

Show, Attend and Tell: Neural Image Caption Generation with Visual Attention

Kelvin Xu, Bengio et al.

http://arxiv.org/pdf/1502.03044v2.pdf

2015年

Abstract

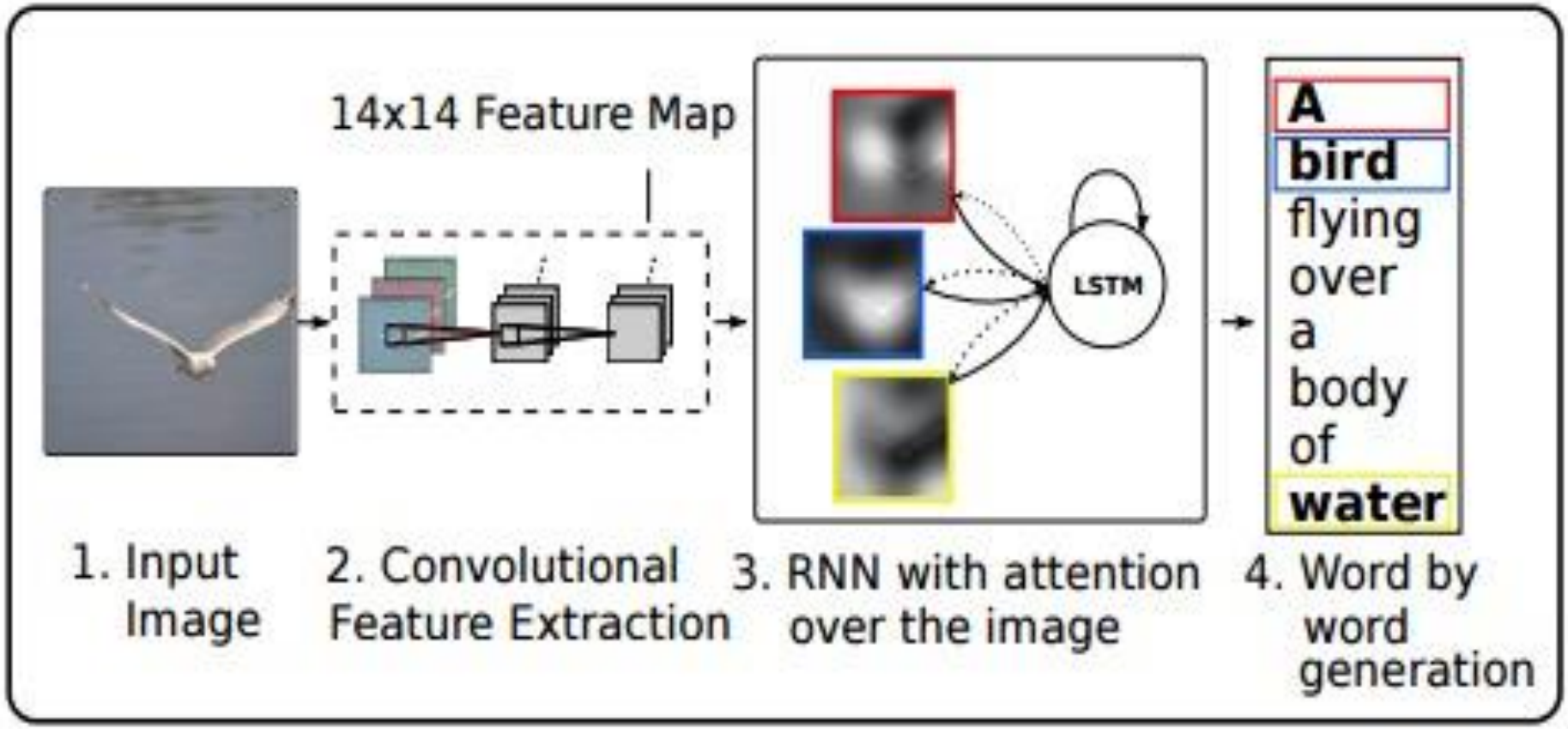
機械翻訳とオブジェクト検出における最近の研究に触発され、画像の内容を記述することを自動的に学習するAttentionベースのモデルを紹介する。

このモデルを、標準的なバックプロパゲーション技術を用いた決定論的方法と、変分下界を最大化することによる確率論的方法で学習する方法を説明する。

また、このモデルが、出力シーケンスにおいて対応する単語を生成しながら、顕著なオブジェクトに視線を固定することを自動的に学習できることを、視覚化を通して示す。

我々は、3つのベンチマークデータセットにおける最先端の性能により、Attentionの利用を検証する：Flickr8k、Flickr30k、MS COCOである。

システム概要


14x14 Feature Map
LSTM
A
bird
flying
over
a
body
of
water
1. Input Image
2. Convolutional Feature Extraction
3. RNN with attention over the image
4. Word by word generation

画像の特定の部分に注意を向ける

A woman is throwing a frisbee in a park.

A dog is standing on a hardwood floor.

A stop sign is on a road with a mountain in the background.

生成されたcaption

画像の特定の部分に注意を向ける

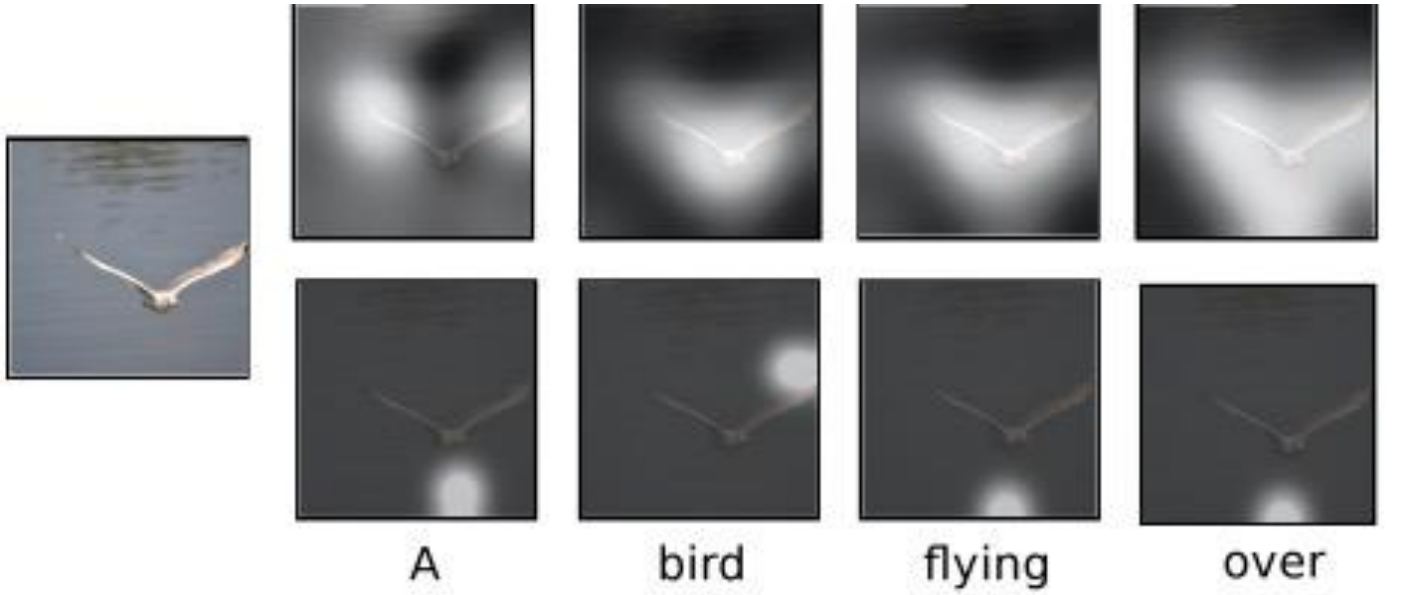
A little girl sitting on a bed with a teddy bear.

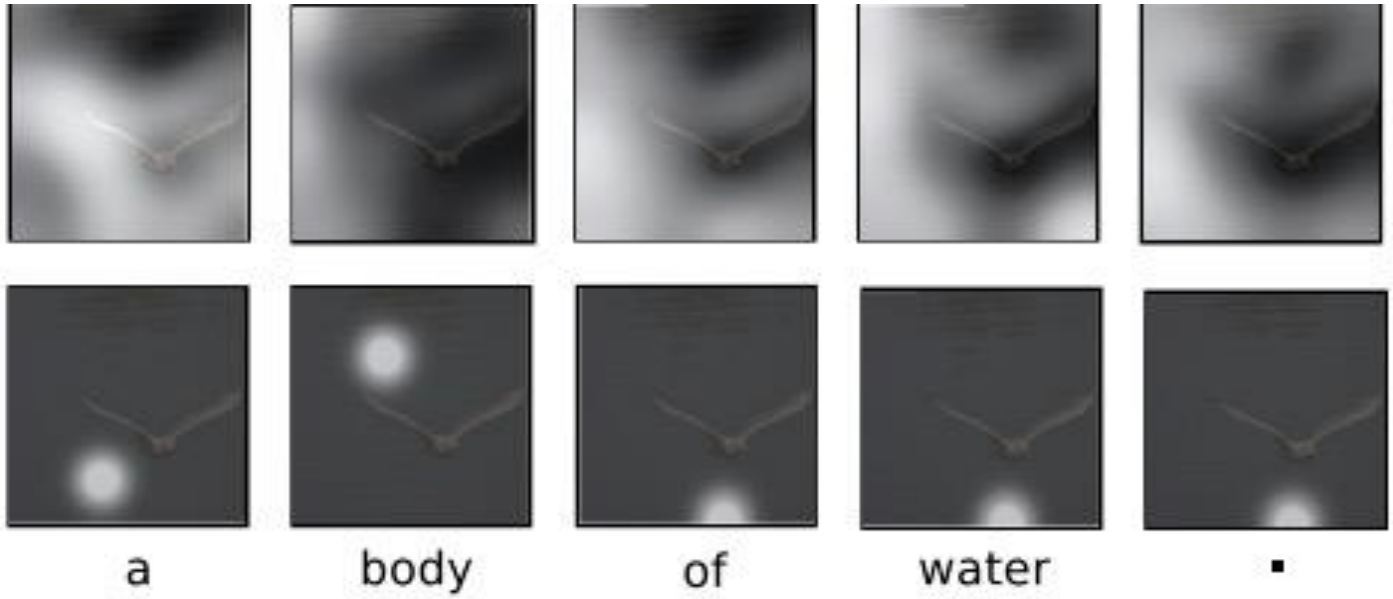
A group of people sitting on a boat in the water.

A giraffe standing in a forest with trees in the background.

生成されたcaption

注意点の移動とcaptionの生成





Captionの失敗例

A large white bird standing in a forest.

A woman holding a clock in her hand.

A man wearing a hat and
a hat on a skateboard.

Captionの失敗は、文法のレベルでなく、オブジェクトの認識の間違いで起きている

Captionの失敗例

A person is standing on a beach with a surfboard.

A woman is sitting at a table with a large pizza.

A man is talking on his cell phone while another man watches.

Captionの失敗は、文法のレベルでなく、オブジェクトの認識の間違いで起きている


Adobe Firefly


Part 2
Vision Transformer
Inductive Bias Free


Vision Transformer
画像embedding

Vision Transformer とは何か？

大規模言語モデルがMulti-Modal なAI に展開して上で、大きな役割を果たしたシステムがあります。それが、2021年に Google が発表した Vision Transformer です。

自然言語処理の世界では、Transformerベースの大規模言語モデルが大きな成功を収めていたのですが、画像情報処理の世界では、近年に至るまで CNN (Convolution Neural Network)が主流でした。

それに対して、GoogleのVision Transformer は、大規模な画像情報処理の世界でも、CNNを全く利用せずに、Transformerだけで最先端のCNNのシステムを上回る性能を発揮できることを示しました。

このことは、Transformerをエンジンとする一つのシステムで、自然言語処理と画像処理のタイプの異なる二つの処理が同時に可能になることを意味しています。

Vision Transformer が、Multi-ModalなAIへの突破口となったというのは、そういうことです。

Vision Transformer のアーキテクチャー

Vision Transformerが自然言語だけではなく、画像も処理できるのは、次のような手法を用いているからです。

「元の画像を小さな画像パッチに分割し、これらのパッチの線形なembeddingのシーケンスをTransformerへの入力として提供する。」

画像パッチは、自然言語処理アプリケーションにおけるトークン（単語）と同じように扱われ、教師あり方式で画像分類モデルを学習します。

論文タイトルの "An Image Is Worth 16x16 Words" というのは、このことを指しています。

注目すべきことは、この画像のembedding の方法を除いては、Vision Transformer は、元のTransformerの実装を、可能な限り修正しないようにしています。

ですから、もしも、自然言語処理での標準的なTransformerの実装を知っていれば、この画像のembedding の方法さえ理解すれば、ほとんど、Vision Transformer の振る舞いを理解できることになります。

このセッションでは、主要に、この画像のembeddingの手法をみていこうと思います。

AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE

Alexey Dosovitskiy, Neil Houlsby et al.

https://arxiv.org/pdf/2010.11929.pdf

2021年

論文の要旨

Transformerアーキテクチャは自然言語処理タスクのデファクトスタンダードとなっているが、コンピュータビジョンへの応用はまだ限られている。

視覚情報処理においては、AttentionはCNNと組み合わせて適用されるか、CNNの全体的な構造を維持したまま、CNNの特定のコンポーネントを置き換えるために使用される。

論文の要旨

我々は、このようなCNNへの依存は必要なく、画像パッチのシーケンスに直接適用される純粋なTransformerが、画像分類タスクにおいて非常に優れた性能を発揮できることを示す。

Vision Transformer（ViT）は、大量のデータで事前に訓練し、複数の中規模または小規模の画像認識ベンチマーク（ImageNet、CIFAR-100、VTABなど）に転送すると、訓練に必要な計算資源が大幅に少ない一方で、最先端のCNNと比較して優れた結果を達成する。

はじめに

Self Attentionに基づくアーキテクチャ、特にTransformers（Vaswani et al.、2017）は、自然言語処理（NLP）で選択されるモデルとなっている。広く用いられているアプローチは、大規模なテキストコーパスで事前学習し、その後、より小規模なタスク固有のデータセットでfine-tuningすることである（Devlin et al.）

Transformersの計算効率とスケーラビリティのおかげで、100Bを超えるパラメータを持つ前例のないサイズのモデルを訓練することが可能になった（Brown et al.） モデルとデータセットが増大する中、性能が飽和する兆候はまだない。

しかし、コンピュータビジョンでは、CNNアーキテクチャが依然として主流である（LeCun et al.）

NLPの成功に触発され、CNNのようなアーキテクチャとSelf Attentionを組み合わせた複数の研究が試みられており（Wangら、2018；Carionら、2020）、CNNを完全に置き換えたものもある（Ramachandranら、2019；Wangら、2020a）。後者のモデルは、理論的には効率的であるが、特殊なAttentionパターンを使用するため、最新のハードウェアアクセラレータではまだ効果的にスケールされていない。

したがって、大規模画像認識では、古典的なResNetライクアーキテクチャが依然として最先端である（Mahajan et al.）

NLPにおけるTransformerのスケーリングの成功に触発され、我々は標準的なTransformerを、可能な限り少ない修正で、画像に直接適用する実験を行う。

そのために、画像をパッチに分割し、これらのパッチの線形なembeddingのシーケンスをTransformerへの入力として提供する。

画像パッチは、自然言語処理アプリケーションにおけるトークン（単語）と同じように扱われる。教師あり方式で画像分類モデルを学習する。

ImageNetのような中規模のデータセットを強力な正規化なしで学習した場合、これらのモデルの精度は、同程度のサイズのResNetsを数%下回る。

この一見がっかりするような結果は予想通りかもしれない：TransformerはCNNに固有の帰納的バイアスのいくつかを欠いているため、十分な量のデータで訓練してもうまく汎化できない。

しかし、より大規模なデータセット（1,400万～3,000万画像）でモデルを学習させると、様相は一変する。我々は、大規模訓練が帰納的バイアスに勝ることを発見した。

我々のVision Transformer(ViT)は、十分なスケールで事前訓練され、より少ないデータポイントのタスクに転送された場合、優れた結果を達成する。

公的なImageNet-21kデータセットや社内JFT-300Mデータセットで事前訓練した場合、ViTは複数の画像認識ベンチマークで最先端技術に近づくか、凌駕する。特にImageNetでは88.55%、ImageNet-ReaLでは90.72%、CIFAR-100では94.55%、VTAB 19タスクでは77:63%の精度を達成した。

最先端のCNNとの比較

	Vision Transformer			CNN	
	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

表2：一般的な画像分類ベンチマークにおける最新技術との比較。
精度の平均と標準偏差を3回の微調整の平均値で示す。JFT-300Mデータセットで事前訓練されたVision Transformerモデルは、すべてのデータセットでResNetベースのベースラインを上回った。より小さなパブリックImageNet-21kデータセットで事前に訓練されたViTも良好な結果を示した。
Touvronら(2020)で報告された88.5%の結果をわずかに改善。

方法

モデル設計において、我々はオリジナルのTransformer（Vaswani et al. ）に可能な限り従った。

この意図的にシンプルなセットアップの利点は、スケーラブルなNLP Transformerアーキテクチャとその効率的な実装が、ほとんどそのまま使えることである。

モデルの概要を次に示す。

Visual Transformer のモデルの概要

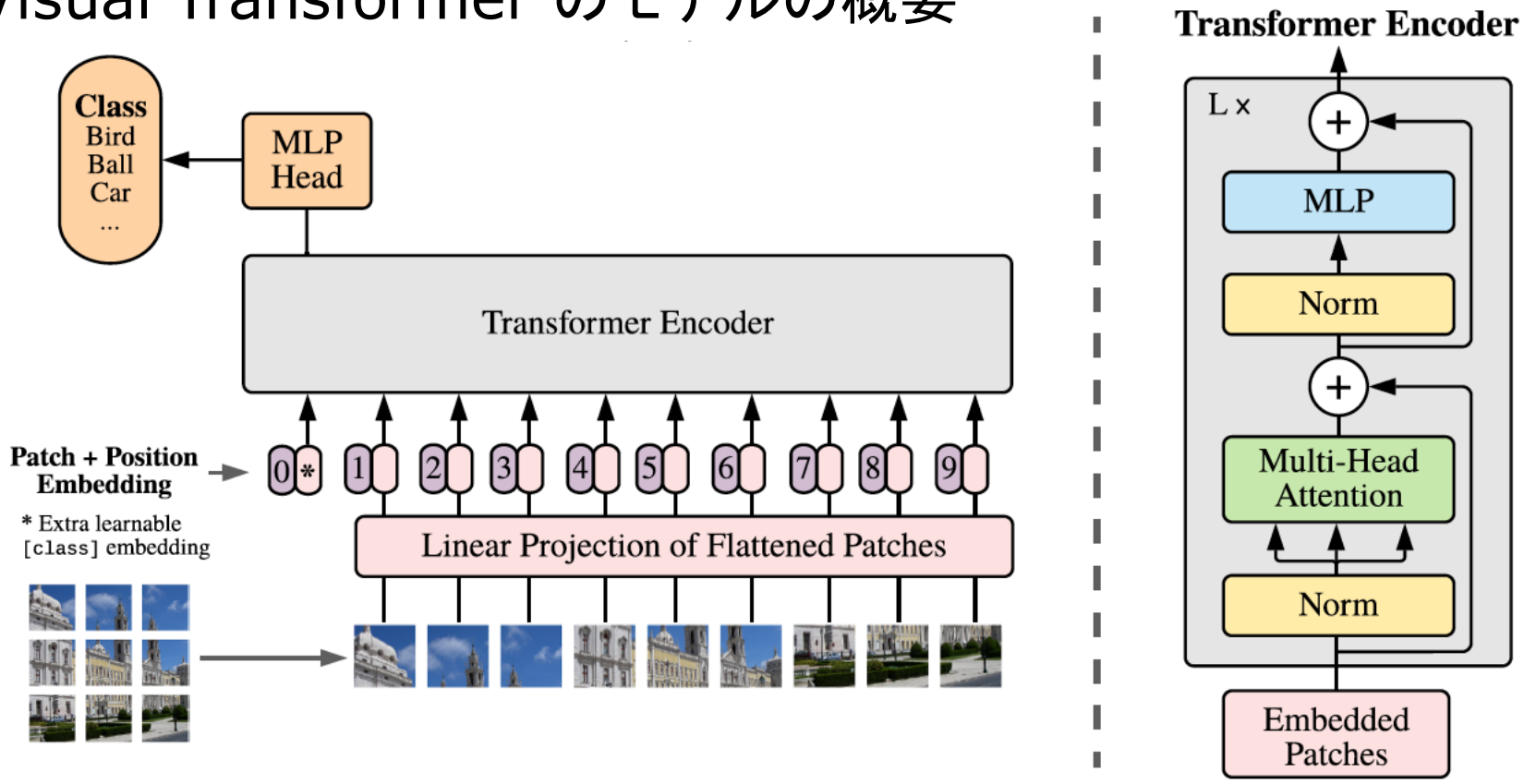


図1：モデルの概要。画像を固定サイズのパッチに分割し、それぞれを線形に埋め込み、位置埋め込みを追加し、得られたベクトル列を標準的なTransformerエンコーダに与える。分類を行うために、学習可能な "分類トークン "をシーケンスに追加するという標準的なアプローチを用いる。Transformerエンコーダの図は、Vaswani et al.

Vision Transformer (VIT)

標準的なTransformerは、トークンembeddingの1次元シーケンスを入力として受け取る。

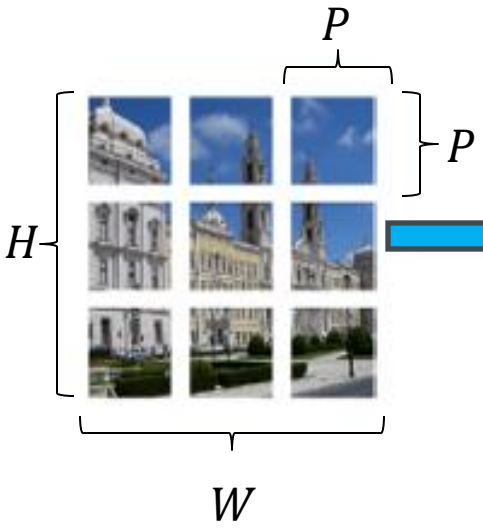
二次元の画像を扱うために、画像 $x \in \mathbb{R}^{H \times W \times C}$ を、平坦化された二次元のパッチのシーケンス $x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$ に変形する。

ここで、(H, W)は元画像の解像度、Cはチャンネル数、(P, P)は各画像パッチの解像度、$N = HW / P^2$は結果として得られるパッチの数であり、これはTransformerの有効な入力シーケンス長でもある。

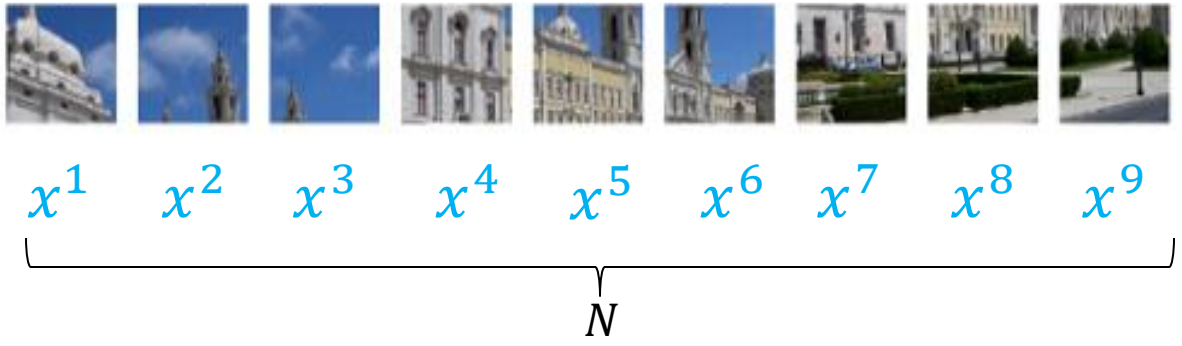
$$x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$$

画像 x をパッチx^iのシークエンスへ

画像 x



パッチx^iのシークエンス



$N = HWC \,/\, P^2\mathrm{C}$

$x \in \mathbb{R}^{H\times W\times C}$

Cは(R, G, B)等の
カラーチャンネル

$x_p \in \mathbb{R}^{N\times(P^2\cdot C)}$

ノテーションの一時変更

$x_p \in \mathbb{R}^{N\times(P^2\cdot C)}$は、小文字のpと大文字のPを使い分けているのだが、少し紛らわしい。

以下では、$x_p \in \mathbb{R}^{N\times(P^2\cdot C)}$ を$x_r \in \mathbb{R}^{N\times(P^2\cdot C)}$と表すことにする。

それでは、このx_rは何を表しているのだろう？

それは、N行× P^2C列の行列で、i行目の要素は、i番目のパッチx^iのP^2C 個のピクセルの要素の並びである。

$$x_r = \begin{bmatrix} x^1\text{の}P^2C\text{ 個のピクセルの要素の並び} \\ x^2\text{の}P^2C\text{ 個のピクセルの要素の並び} \\ \cdots \\ x^N\text{の}P^2C\text{ 個のピクセルの要素の並び} \end{bmatrix}$$

行列x_rのi行目を取り出す射影x_r^i

行列x_rのi行目を取り出す射影を、x_r^iと表そう。
先のように、行列x_rが定義されているとすると、

x_r^1 ＝ パッチ x^1のP^2C 個のピクセルからなる行ベクトル
x_r^2 ＝ パッチ x^2のP^2C 個のピクセルからなる行ベクトル
…
x_r^N ＝ パッチ x^NのP^2C 個のピクセルからなる行ベクトル

ということになる。

$x_p \in \mathbb{R}^{N\times(P^2\cdot C)}$の、小文字のpと大文字のPを使い分けに注意して
論文のノテーションに戻ることにしよう。

P^2C次元の行ベクトルのN個の並びへ

$x_p \in \mathbb{R}^{N\times(P^2\cdot C)}$ とすると、$x_p^i \in \mathbb{R}^{1\times(P^2\cdot C)}$は、パッチ$x^i$の$P^2C$個のピクセルの要素を並べた$P^2C$次元の行ベクトルとなる。

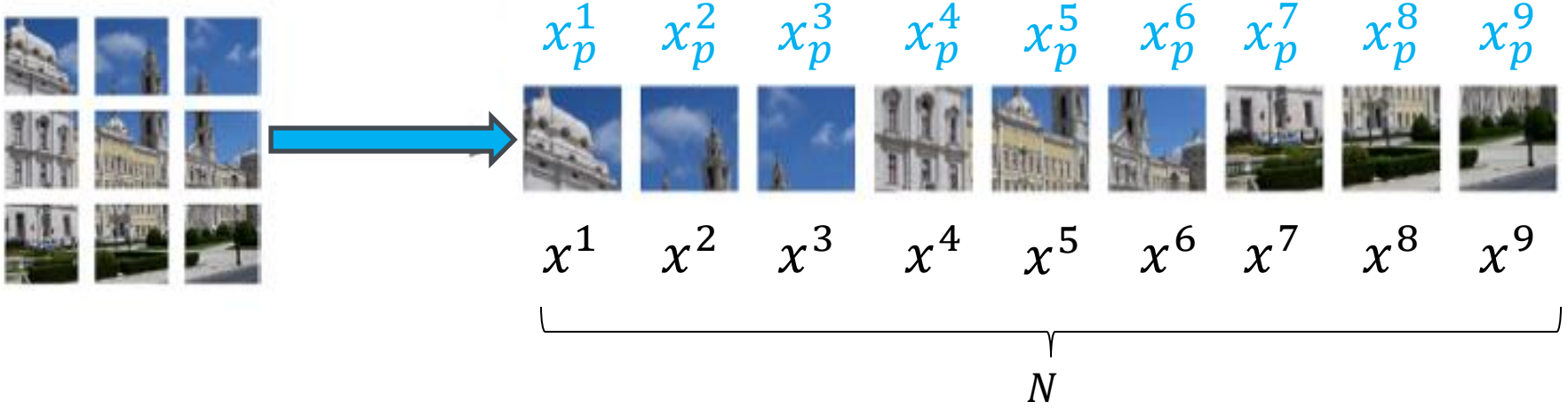
よって、並び $x_p^1, x_p^2, \cdots, x_p^N$ はP^2C次元の行ベクトルのN個の並びである。

P^2C次元の行ベクトルのN個の並びへ

$x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$

$x_p^i \in \mathbb{R}^{1 \times (P^2 \cdot C)}$

パッチ x^iのP^2C 個のピクセルからなる
行ベクトルx_p^iのN個の並び



画像 x

パッチx^iのN個のシークエンス

Patch Embedding

Transformerはすべての層で一定のベクトルサイズDを使用するので、パッチを平坦化し線形射影 E を用いてD次元に写像する。この射影の出力をパッチembeddingと呼ぶ。

$$x_p \in \mathbb{R}^{N\times\left(P^2\cdot C\right)}$$

$$x_p^1 \in \mathbb{R}^{1\times\left(P^2\cdot C\right)}, x_p^2 \in \mathbb{R}^{1\times\left(P^2\cdot C\right)}, \cdots, x_p^N \in \mathbb{R}^{1\times\left(P^2\cdot C\right)}$$

$E \in \mathbb{R}^{\left(p^2C\right)\times D}$ であるEを選ぶと、
$x_p^i E$ は、$1\times p^2C$の行ベクトルと$p^2C\times D$ の行列 の積なので
$\longrightarrow x_p^i E$ は $1\times D$ でD次元の行ベクトルになる。

$\left[x_p^1 E;\ x_p^2 E;\cdots;\ x_p^N E\right]$は、
N個のD次元行ベクトルの並びである。

D次元のembeddingを構成する

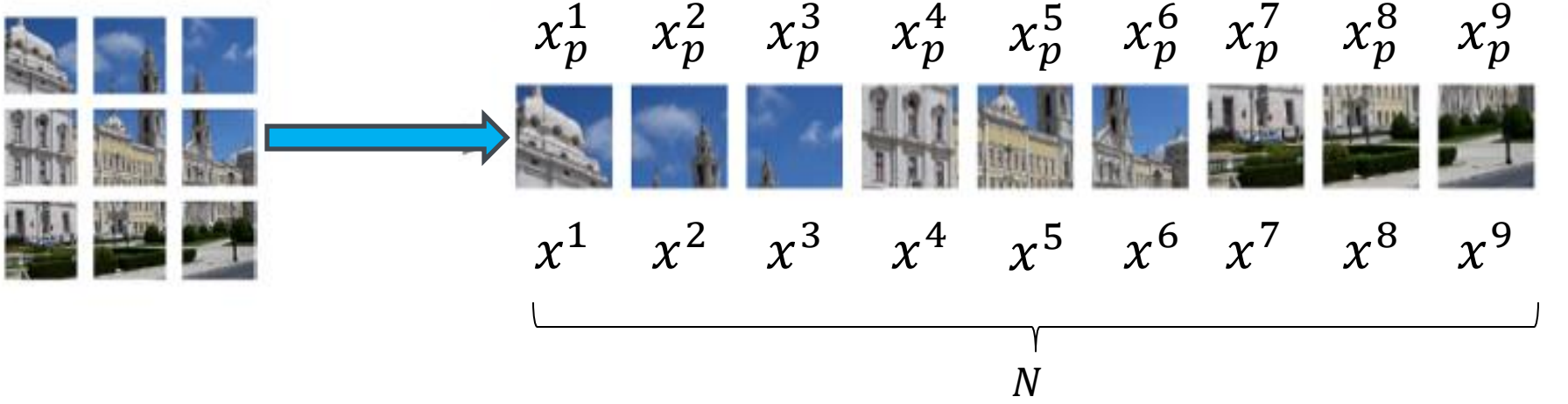
Linear Projection of Flattened Patches

$x_p^i \in \mathbb{R}^{1\times(P^2\cdot C)}$

$E \in \mathbb{R}^{(P^2C)\times D}$

$x_p^i E \in \mathbb{R}^{1\times D}$

D次元の行ベクトルのN個の並び



画像 x

パッチx^iのN個のシークエンス

[class]トークンの追加と位置埋め込み

BERTの[class]トークンと同様に、学習可能なembeddingをパッチembeddingのシーケンス（$z_0^0 = x_{class}$）に付加し、Transformerエンコーダの出力におけるその状態（z_L^0）が画像表現yとして機能する（式4）。

位置埋め込みは、位置情報を保持するためにパッチembeddingに追加される。我々は標準的な学習可能な1次元の位置embeddingを使用する（付録D.4）。結果として得られる埋め込みベクトルのシーケンスはエンコーダの入力となる。

$$E \in \mathbb{R}^{(p^2C)\times D},\ E_{pos} \in \mathbb{R}^{(N+1)\times D}$$

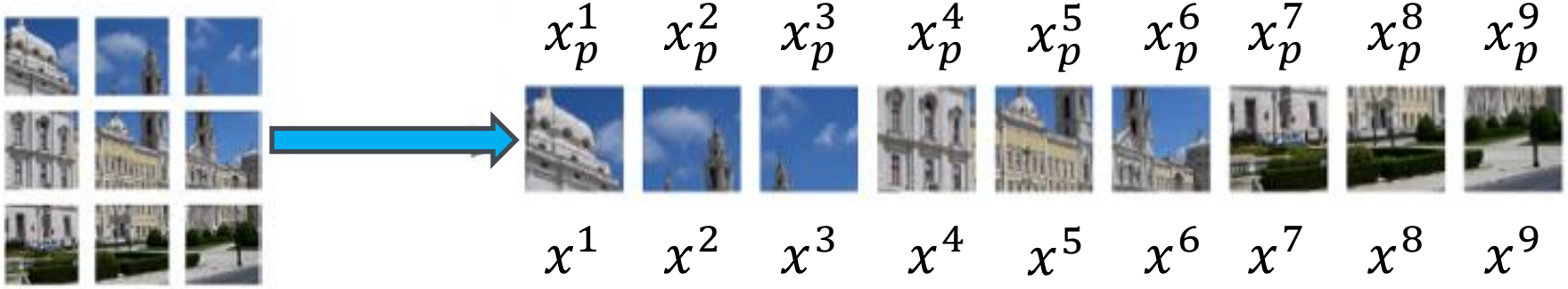
$$z_0 = \left[x_{class};\ x_p^1 E;\ x_p^2 E;\cdots;\ x_p^N E\right] + E_{pos}$$

学習可能な [class]トークンの追加

$$[\, x_{class}; x_p^1 E; x_p^2 E; x_p^3 E; x_p^4 E; x_p^5 E; x_p^6 E; x_p^7 E; x_p^8 E; x_p^9 E \,]$$

$x_p^1 E$ $x_p^2 E$ $x_p^3 E$ $x_p^4 E$ $x_p^5 E$ $x_p^6 E$ $x_p^7 E$ $x_p^8 E$ $x_p^9 E$

Linear Projection of Flattened Patches



画像 x

パッチx^iのN個のシークエンス

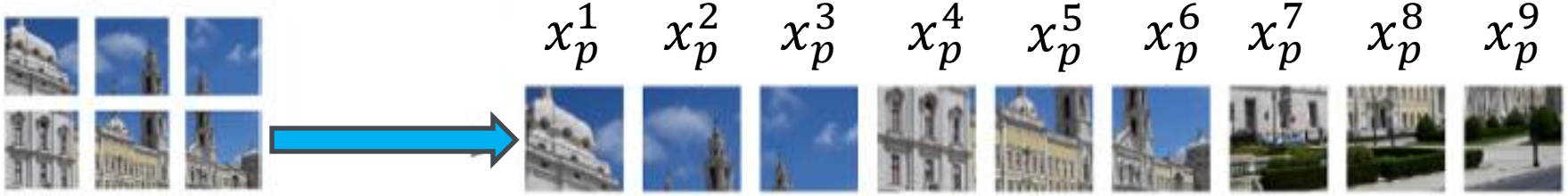
位置埋め込みの追加

$$[\, x_{class};\ x_p^1E; x_p^2E; x_p^3E;\ x_p^4E; x_p^5E; x_p^6E; x_p^7E; x_p^8E;\ x_p^9E \,] + x_{pos}$$

$$[\, x_{class}; x_p^1E; x_p^2E; x_p^3E;\ x_p^4E; x_p^5E; x_p^6E; x_p^7E; x_p^8E;\ x_p^9E \,]$$

x_p^1E x_p^2E x_p^3E x_p^4E x_p^5E x_p^6E x_p^7E x_p^8E x_p^9E

Linear Projection of Flattened Patches



x_p^1 x_p^2 x_p^3 x_p^4 x_p^5 x_p^6 x_p^7 x_p^8 x_p^9

x^1 x^2 x^3 x^4 x^5 x^6 x^7 x^8 x^9

N

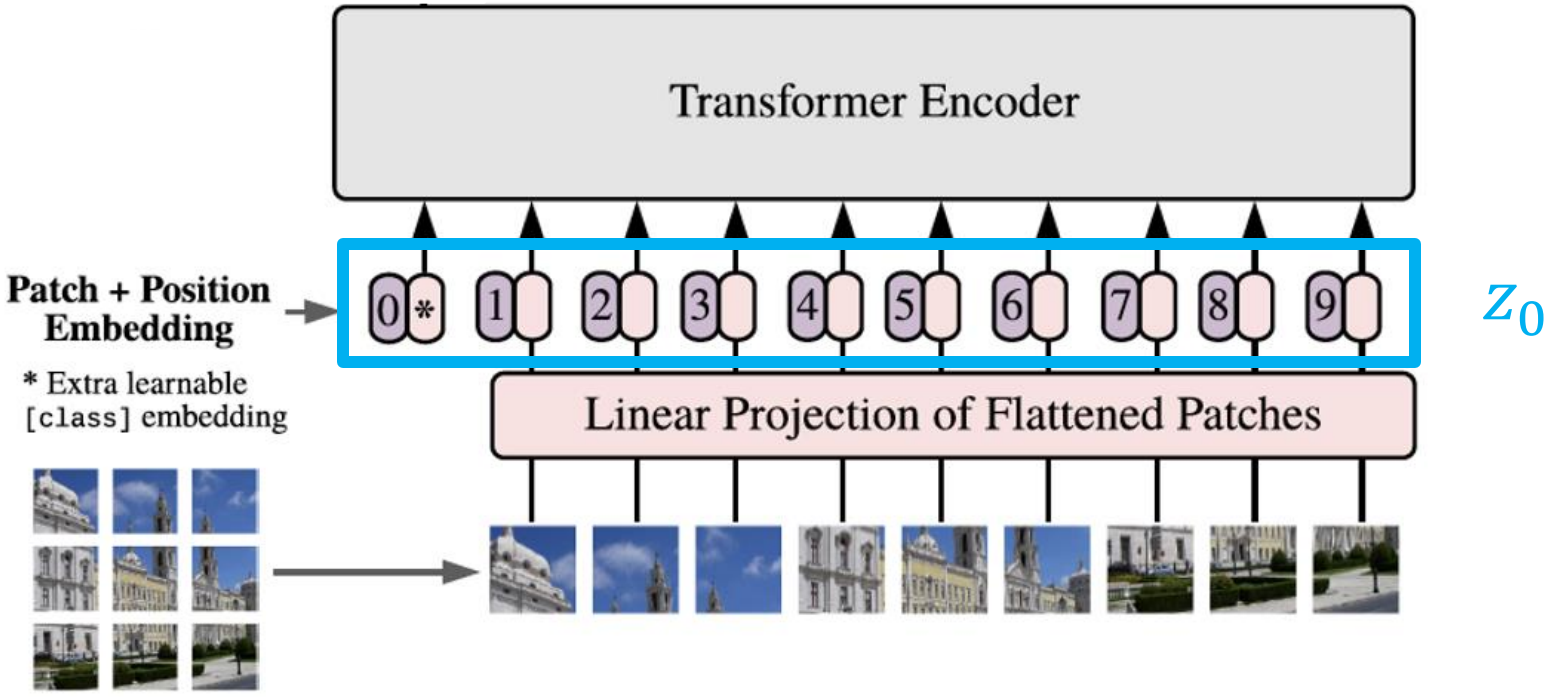
画像 x

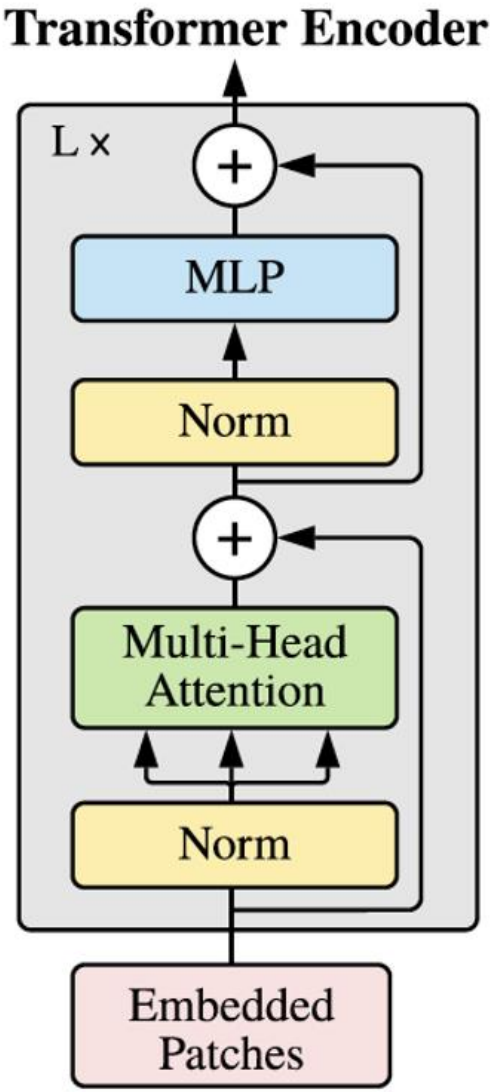
パッチx^iのN個のシークエンス

エンコーダへの入力 z_0

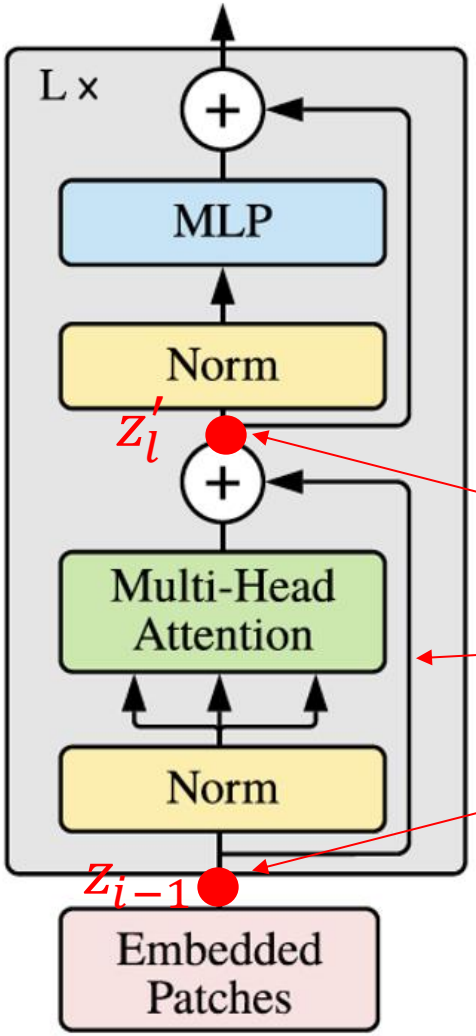
$$z_0 = [\, x_{class}; x_p^1 E; x_p^2 E; x_p^3 E; x_p^4 E; x_p^5 E; x_p^6 E; x_p^7 E; x_p^8 E; x_p^9 E \,] + x_{pos}$$

一般にエンコーダへの入力　$z_0 = [\, x_{class}; x_p^1 E; x_p^2 E; \cdots; x_p^N E \,] + x_{pos}$

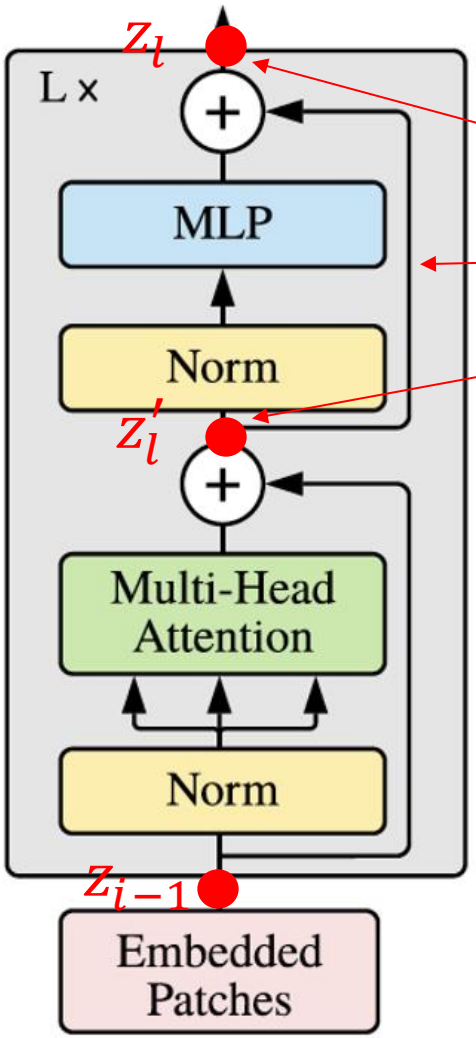




Transformerエンコーダは、Multi-head Self Attention（MSA)とMLP(Multi Layer Perceptron)ブロック（の交互の層で構成される。各ブロックの前にはLayer-norm（LN）が適用され、各ブロックの後には残差接続が適用される。MLPにはGELU非線形性を持つ2つの層が含まれる。



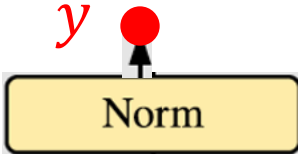
$$z_l' = MSA\left(LN\left(z_{l-1}\right)\right) + z_{t-1}$$



$$z_l = MLP\left(LN(z'_l)\right) + z'_l$$

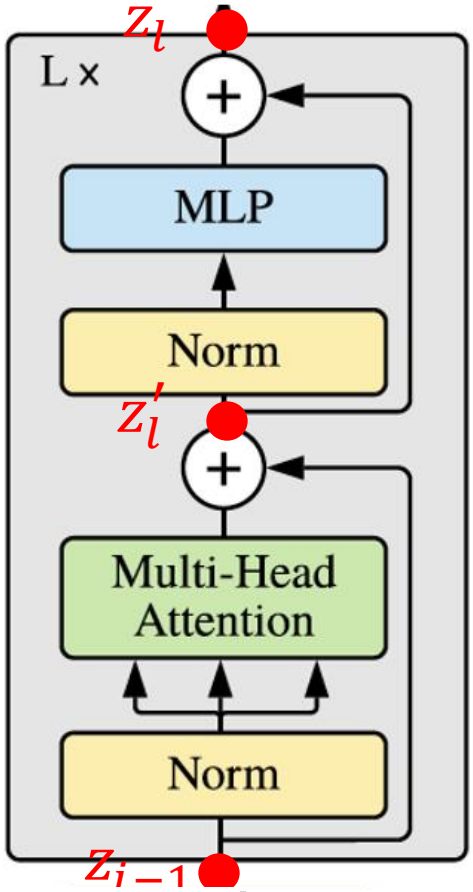
$$z'_l = MSA\left(LN\left(z_{l-1}\right)\right) + z_{l-1}$$

$l = 1 \cdots L$
Lは処理の多重度



y

$z_1\ z_2 \cdots z_l \cdots z_L$



z_l

z_l'

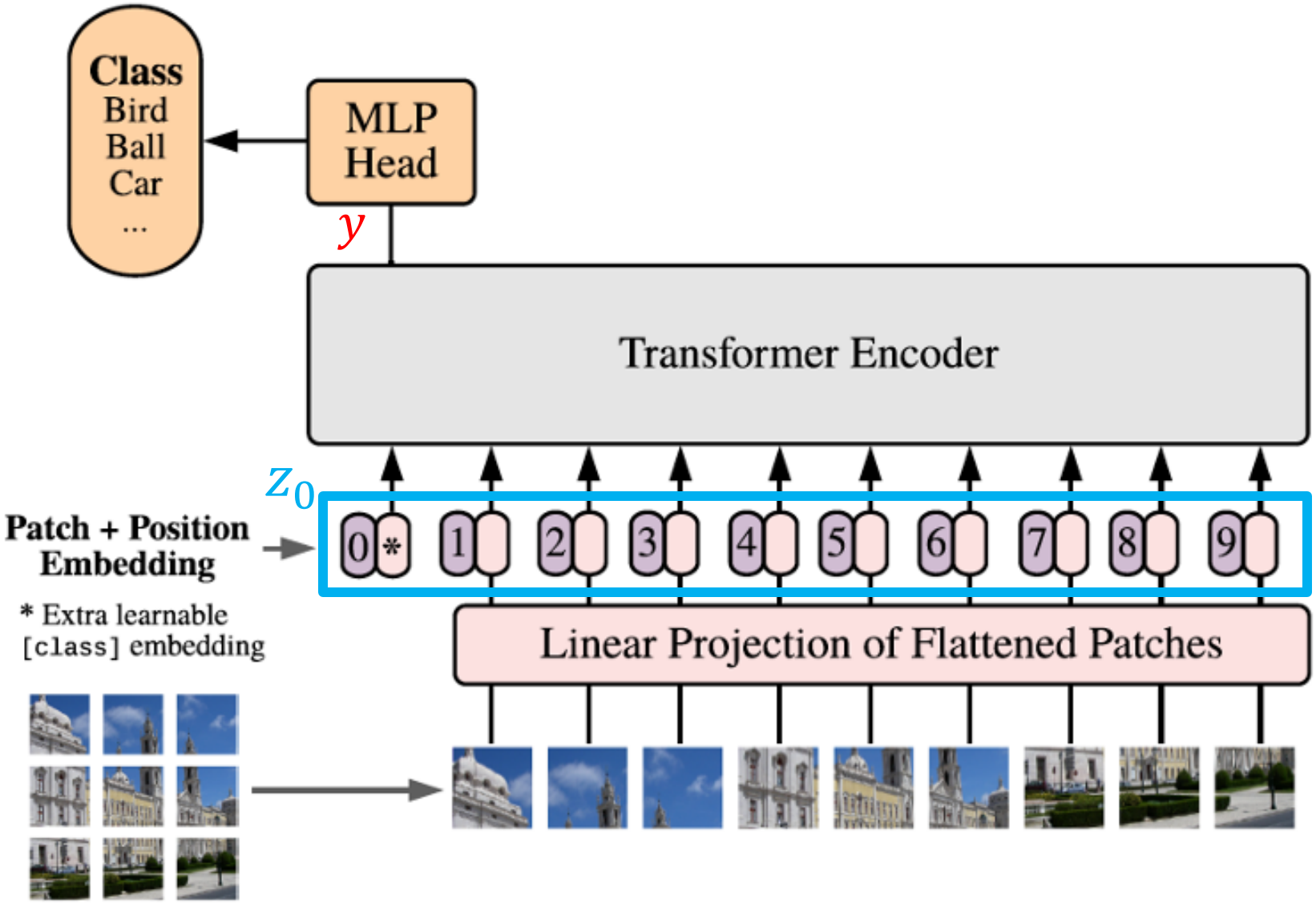
z_{l-1}

$$y = LN(z_L^0)$$

$$z_l = MLP\left(LN(z_l')\right) + z_l'$$

$$z_l' = MSA\left(LN\left(z_{l-1}\right)\right) + z_{l-1}$$

$$z_0 = [\, x_{class};\ x_p^1 E;\ x_p^2 E;\ x_p^3 E;\ x_p^4 E;\ x_p^5 E;\ x_p^6 E;\ x_p^7 E\ \ x_p^8 E;\ x_p^9 E\,] + x_{pos}$$


Class
Bird
Ball
Car
...
MLP Head
y
Transformer Encoder
z_0
Patch + Position Embedding
* Extra learnable [class] embedding
0 * 1 2 3 4 5 6 7 8 9
Linear Projection of Flattened Patches

Vision Transformer

内部表現の分析

Vision Transformer と CNN

Vision Transformerは “Inductive Bias Free”

今回のセッションの隠れたテーマの一つは、前回も触れた Vision Transformer の「CNNのInductive Biasの排除」という設計デザインについてです。

「Vision TransformerはCNNに比べ、画像固有の帰納的バイアスがはるかに少ない。」

「CNNでは、局所性、2次元近傍構造、並進等価性がモデル全体を通して各層に焼き付けられている。」

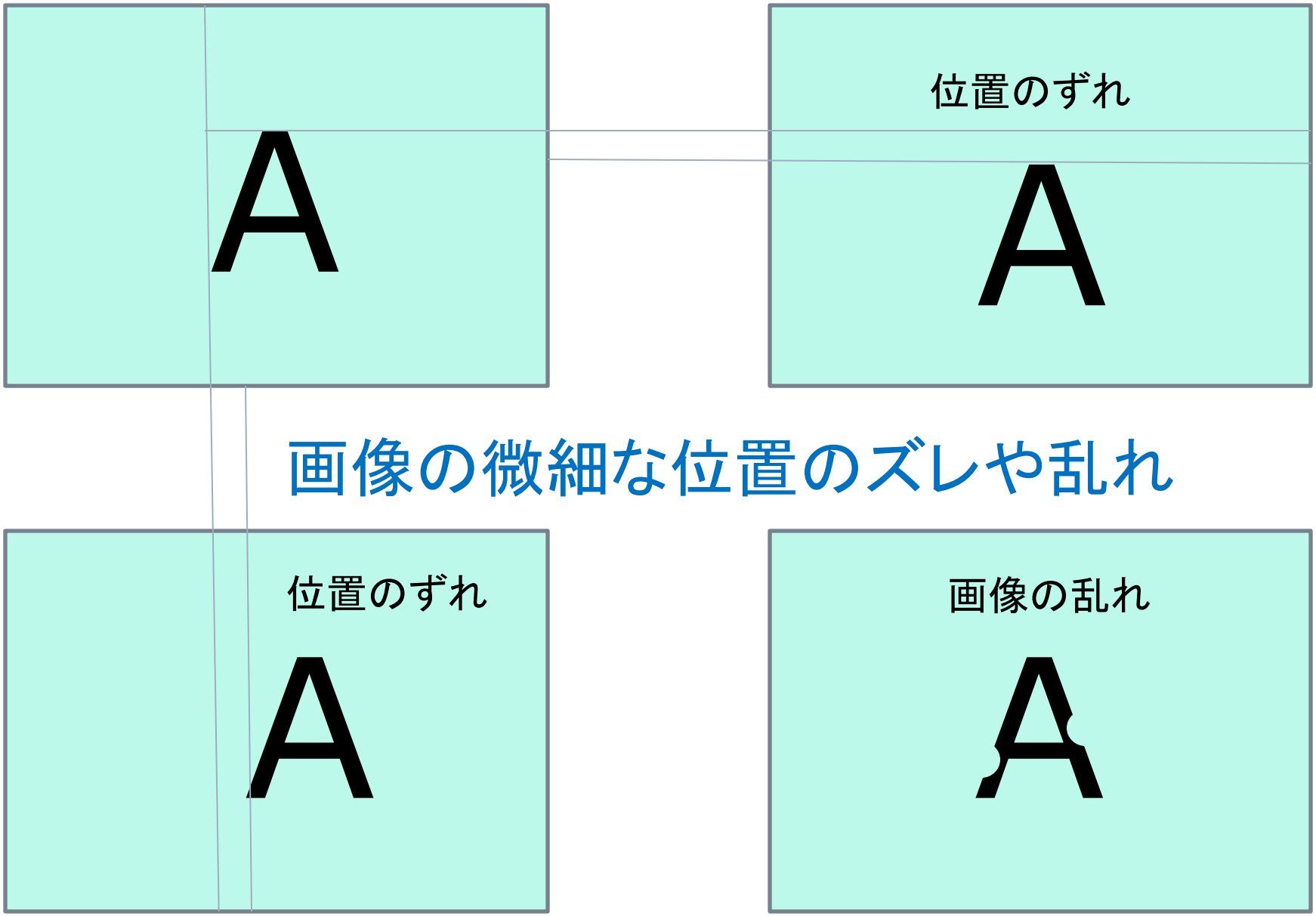
最初に、CNNが画像認識の上で、画像データの特徴をどのように捉えているかを、簡単に振り返ろうと思います。

CNNがとらえる画像データの特徴

- Full Connectなネットワーク (MLP)では、入力層の全てのノードが、次の層の全てのノードと接続しているので、入力層上でのデータの並びは、大きな意味を持たない。

- ただ、画像データでは、一つ一つのピクセルの位置情報とその並びは、意味を持つ。同時に、画像認識という視点から考えれば、ピクセルの微細な位置のずれや画像の乱れは、画像の認識にとっては、大きな意味を持たない。こうした、相反する要求を、同時に満たすことが、求められることになる。

- また、RGBで表された三枚の画像のデータには、対応する箇所で、強い相関がある。同様に、画像上の特徴的なパターンは、データ内部の局所的な強い相関として捉えることができる。

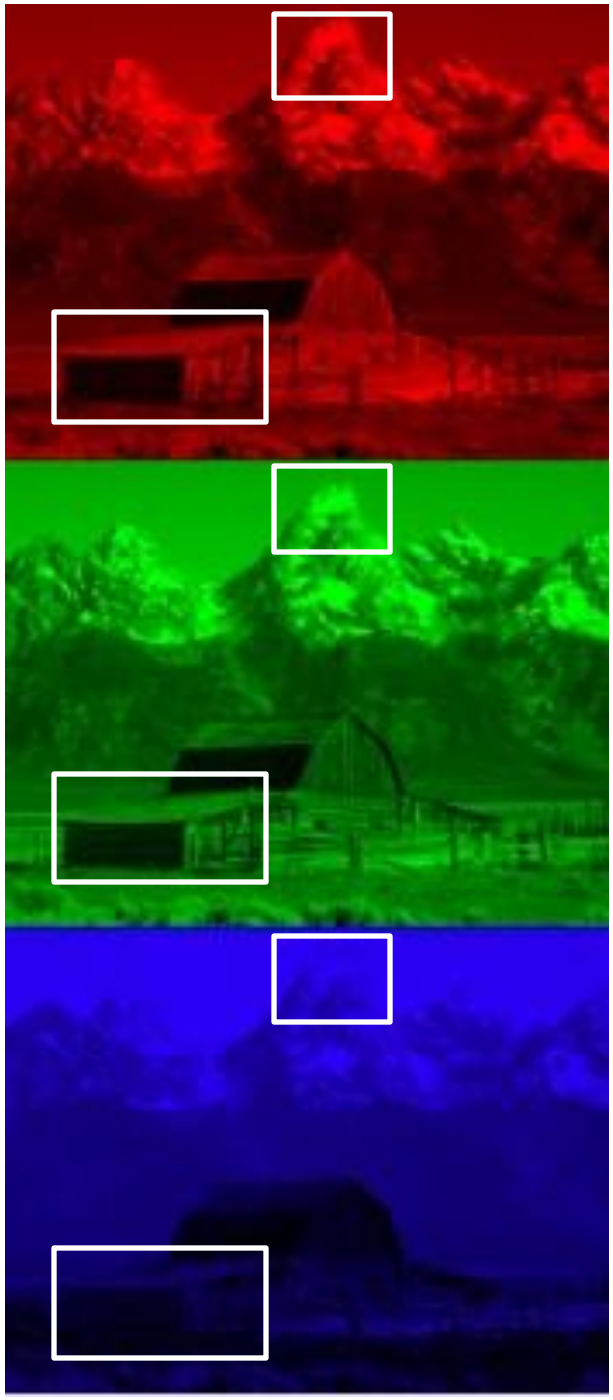

A
位置のずれ
A
画像の微細な位置のズレや乱れ
位置のずれ
A
画像の乱れ
A

カラー画像のRGB表現

R

G

B



三つの画像のピクセルには相関がある

Vision Transformerの画像認識上の特徴

ViTではMLP層のみが局所的で並進等価であり、Self Attention層は大域的である。

2次元の近傍構造の利用は、画像をパッチに分割することによるモデルの開始時と、異なる解像度の画像に対する位置埋め込みを調整するためのfine-tuning時に、ごく僅かに使用されるのみである。

それ以外は、初期化時の位置埋め込みはパッチの2次元位置に関する情報を持たず、パッチ間の空間関係はすべてゼロから学習する必要がある。

Vision Transformerの内部表現の分析

「Vision Transformerがどのように画像データを処理するかを理解するために、その内部表現を分析する。」

内部表現の分析

「 内部表現の分析 」というのは、システムの内部で、入力に与えられたデータが出力に至るまでどのように変化するのかを追いかける分析です。

ディープラーニングのニューラル・ネットワークは、膨大な量のデータが、多数の「層」を通り抜け、かつそれらが相互に作用するので、アーキテクチャーの構成をみただけでは「なぜ、このシステムで、こういう働きが可能になるの？」という疑問の答えは得られません。

一つ一つのデータの動きを追いかけても（実際には、それは無理なのですが）、それぞれの層でのデータを表現する数式を眺めても（それは抽象的すぎます）、システムの「ふるまい」は、さっぱり見えてきません。

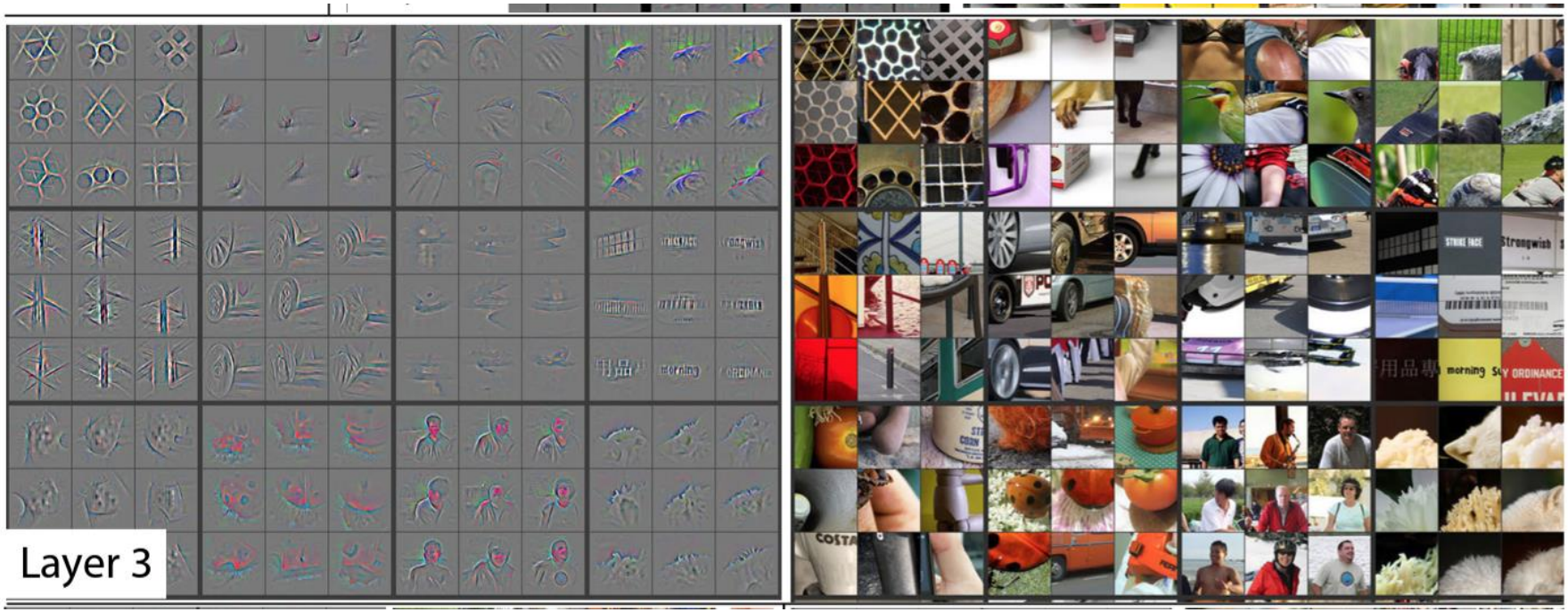
システムの可視化

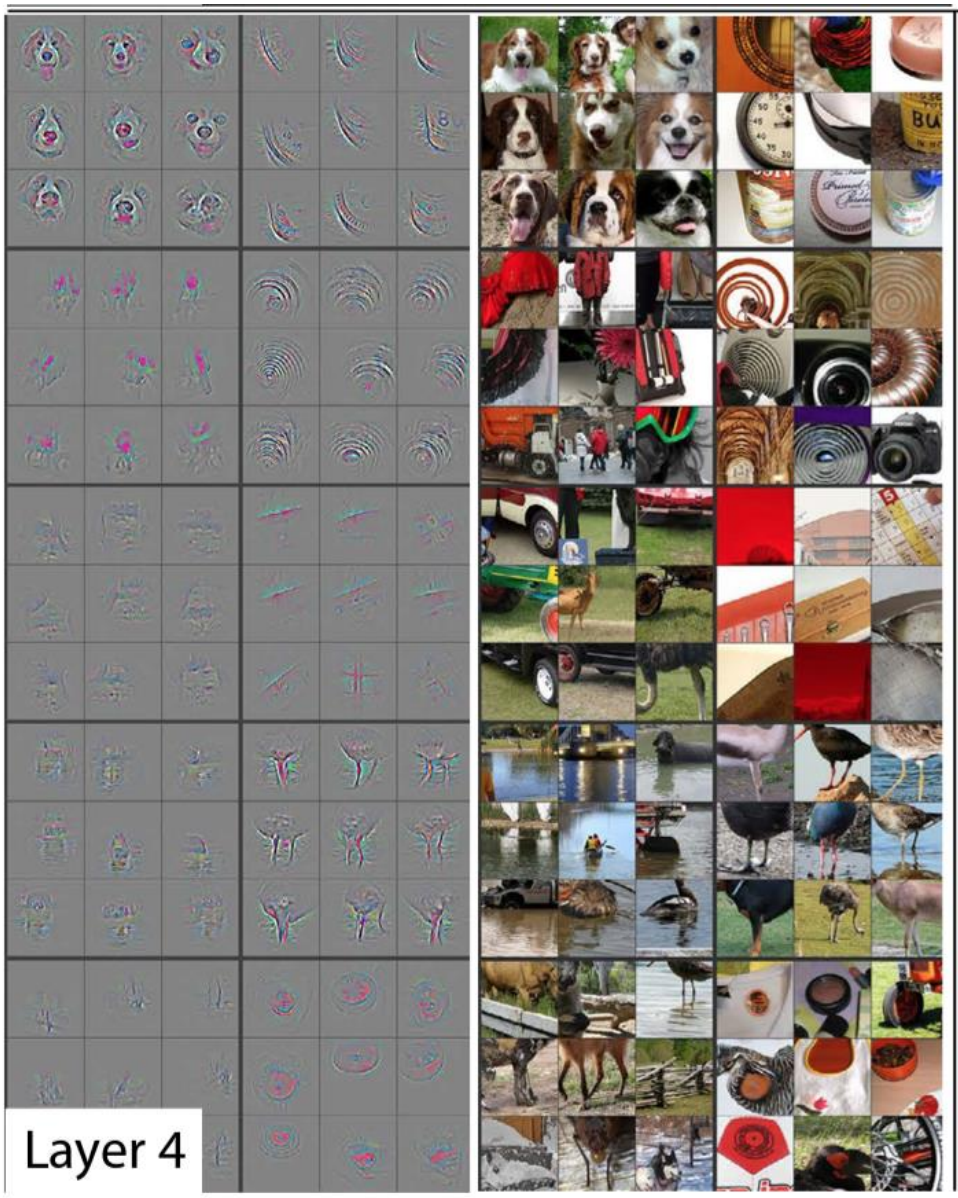
でも、いい方法があるのです。それは、複雑なものを一瞬・一瞥で理解・把握する人間の視覚の力を利用することです。

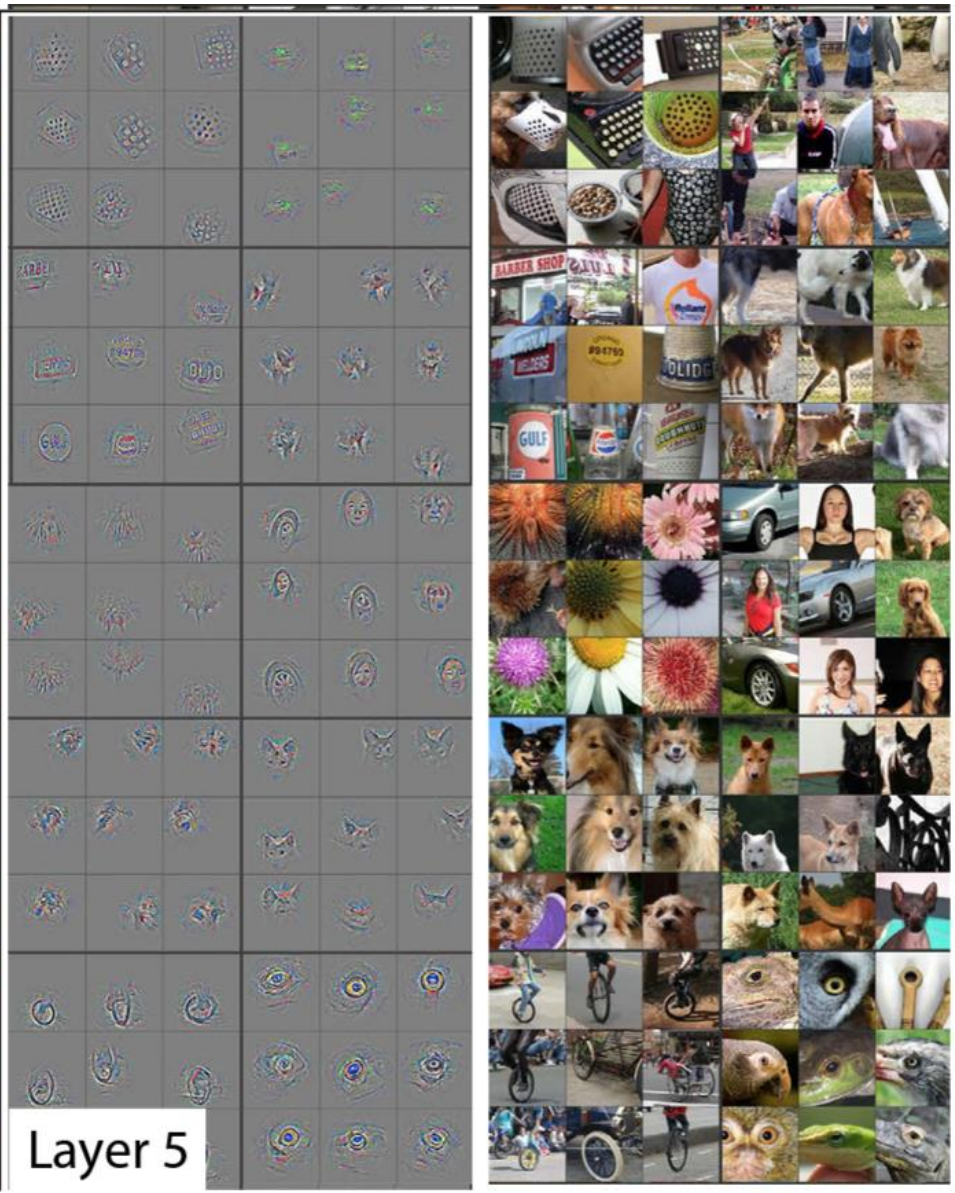
データではなく各層でもっとも活発に呼び出されている「ふるまい」に注目します。そのふるまいが各層でデータをどのように変えているかを、そのふるまいの結果の全部をデータの画像として表示します。

そうすれば、その層のふるまい全部ではないけど、もっとも頻繁に呼び出されているふるまいの結果を画像として可視化することができます。これが、重要な情報を与えてくれることになります。

2013年の Zeiler , Fergusの "Visualizing and Understanding Convolutional Networks" https://arxiv.org/pdf/1311.2901.pdf は、この分野を切り拓いた画期的な論文です。お時間があったら、ぜひ、お読みください。画像を見るだけでも楽しいです。




Layer 4


Layer 5

Attentionの働きが分析の焦点の一つ

「 Vision Transformerの 内部表現の分析 」では、Vision Transformer での Attentionの働きが分析の焦点になります。

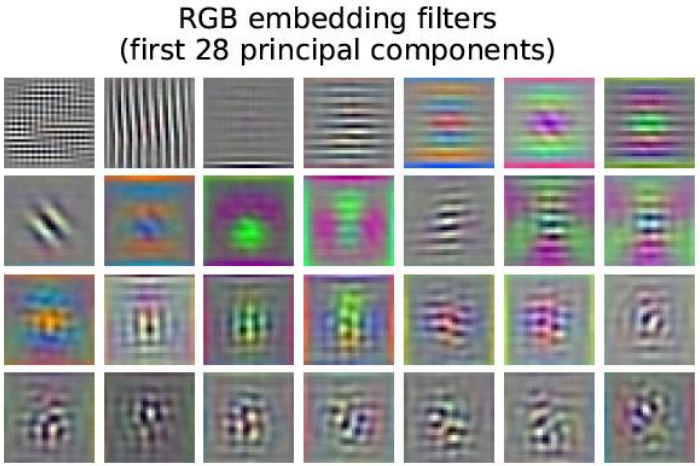
なぜなら、Transformer ＝ 分散表現＋Attentionと考えていいので、このアーキテクチャーで CNNと同じような画像認識の機能を発揮することができるのは何故かという疑問が出てくるのは当然ですから。

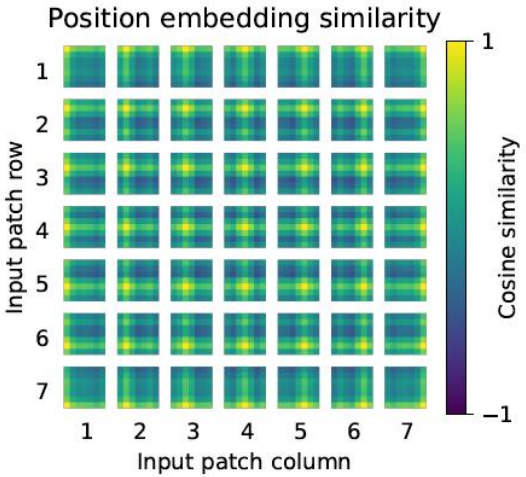
その答えの一つは、Vision Transformer も、CNNと同じふるまいを行うことがあることを示すことです。

Vision Transformerの内部表現の分析

この論文では、見える化によって次のようなことが示唆されています。

- Vision Transformer の最初の層での画像embedding は、CNNの最下層のフィルターの働きとほとんど同じように見えること。
- Vision Transformer では、元の画像は異なる位置情報を持つ多数の画像パッチにばらばらにされるのだが、学習の結果、それらの本来の位置関係は学習されること。
- Vision Transformer での距離の近い画像へのAttentionの集中は、CNNのConvolutionと同じ効果を持つらしいこと。
- Vision Transformer の「Attentionの距離」は、CNNの「局所的受容野」のサイズに相当するらしいこと。





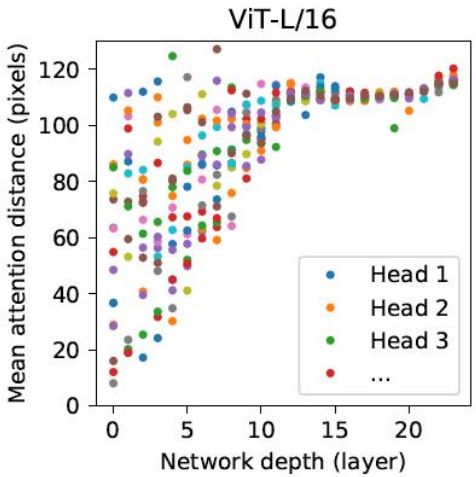


図7:

- 左:ViT-L／32のRGB値の低層での線形埋め込みフィルタ。
- 中央: ViT-L／32の位置埋め込みの類似度。タイルは、指定された行と列を持つパッチの位置埋め込みと、他のすべてのパッチの位置埋め込みとの間のコサイン類似度を示す。
- 右: ヘッドとネットワークの深さによるAttention領域の大きさ。各ドットは1つのレイヤーにおける16個のヘッドの1つについて、画像全体の平均Attention距離を示す。詳細は付録D.7参照。

Vision Transformerの第一層

Vision Transformerの第1層は、平坦化されたパッチを低次元空間に線形射影する。

$$z_0 = \left[x_{class} ; x_p^1 E ; x_p^2 E ; \cdots ; x_p^N E \right] + x_{pos}$$

$$x_p^i \in \mathbb{R}^{1\times(p^2 C)}, E \in \mathbb{R}^{(p^2 C)\times D} \rightarrow x_p^i E \in \mathbb{R}^{1\times D}$$

$$x_{pos} \in \mathbb{R}^{(N+1)\times D}$$

この成分は、各パッチ内の微細構造を低次元で表現するための、CNNの基本的なフィルターの成分に似ている。

学習された埋め込みフィルタの上位主成分

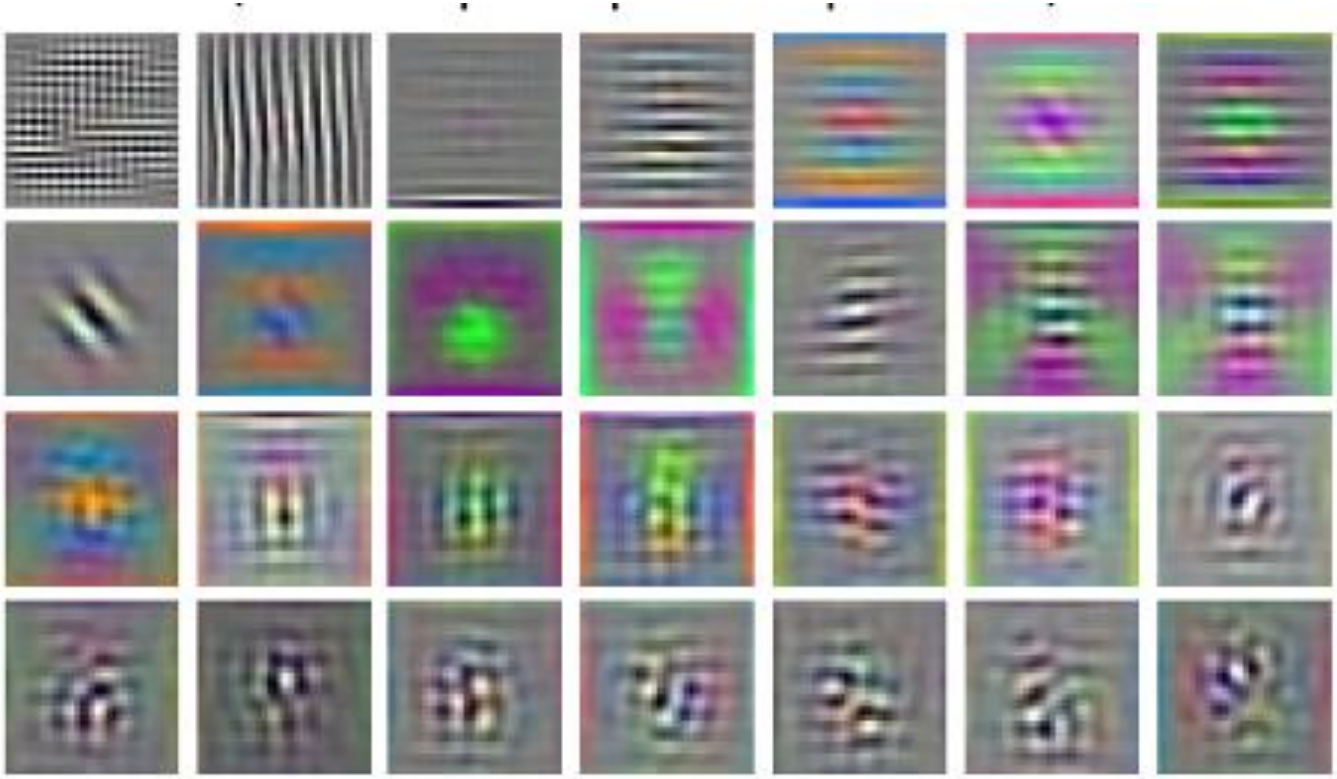
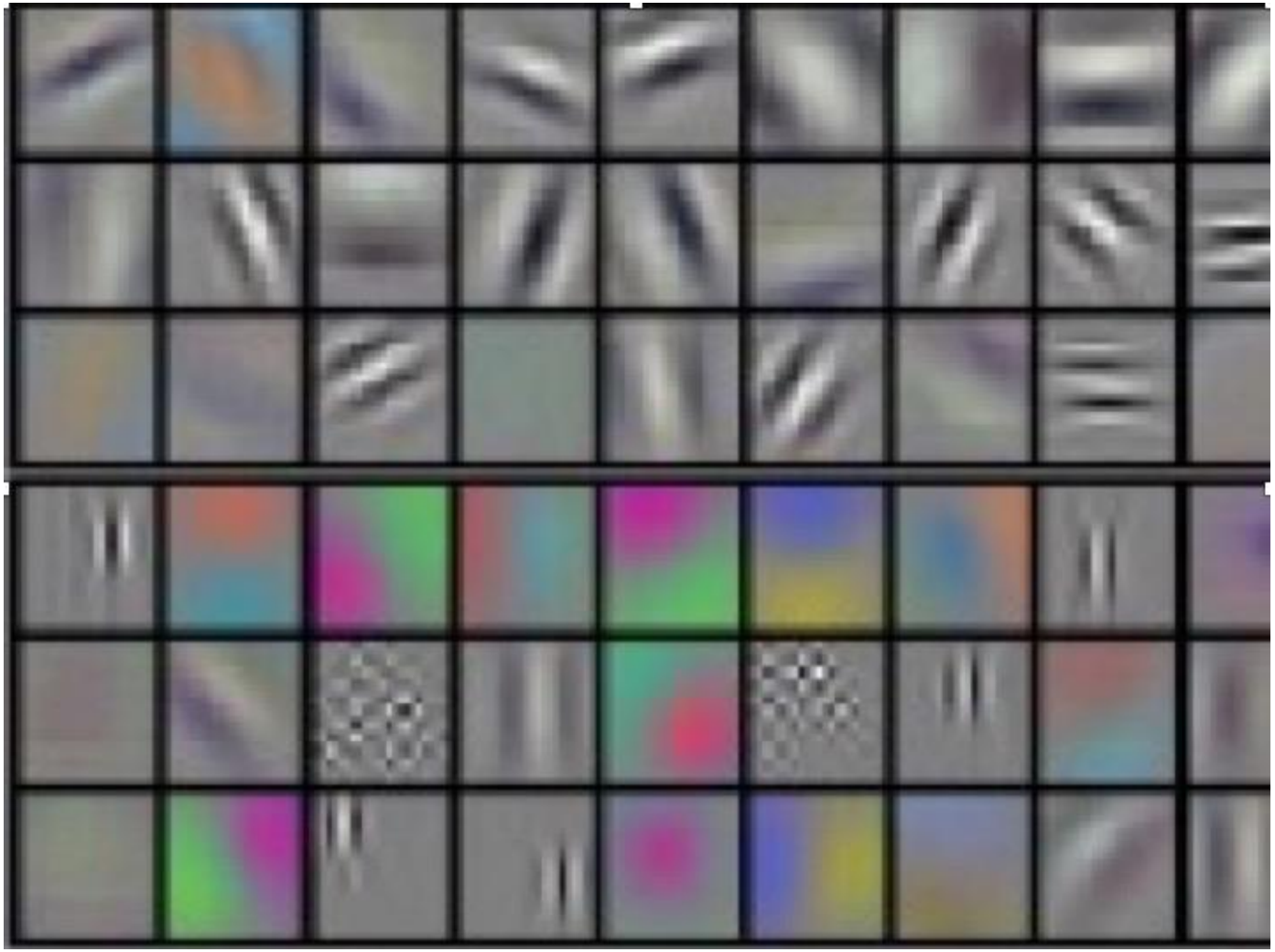


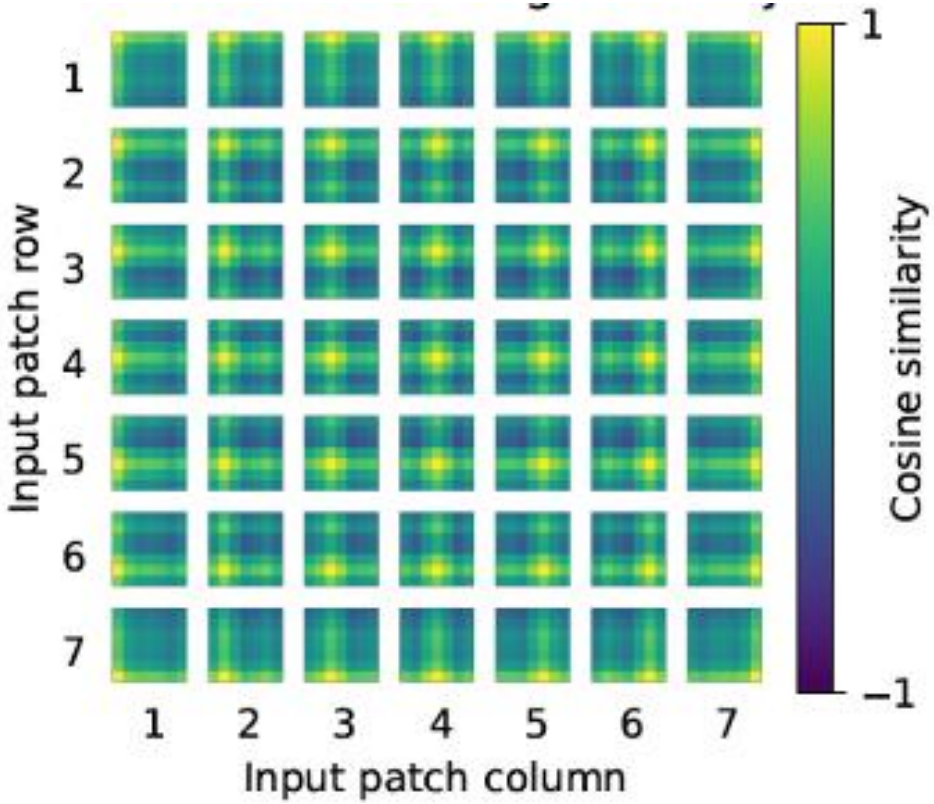
図7：左：ViT-L／32のRGB値の初期線形埋め込みフィルタ

CNNで学習されたフィルター



位置埋め込みの類似性

射影の後、学習された位置埋め込みがパッチ表現に追加される。図7（中央）は、このモデルが画像内の距離を位置埋込みの類似度で符号化することを学習していることを示している、すなわち、より近いパッチはより類似した位置埋込みを持つ傾向がある。



さらに、行-列構造が現れ、同じ行／列のパッチは類似した埋め込みを持つ。

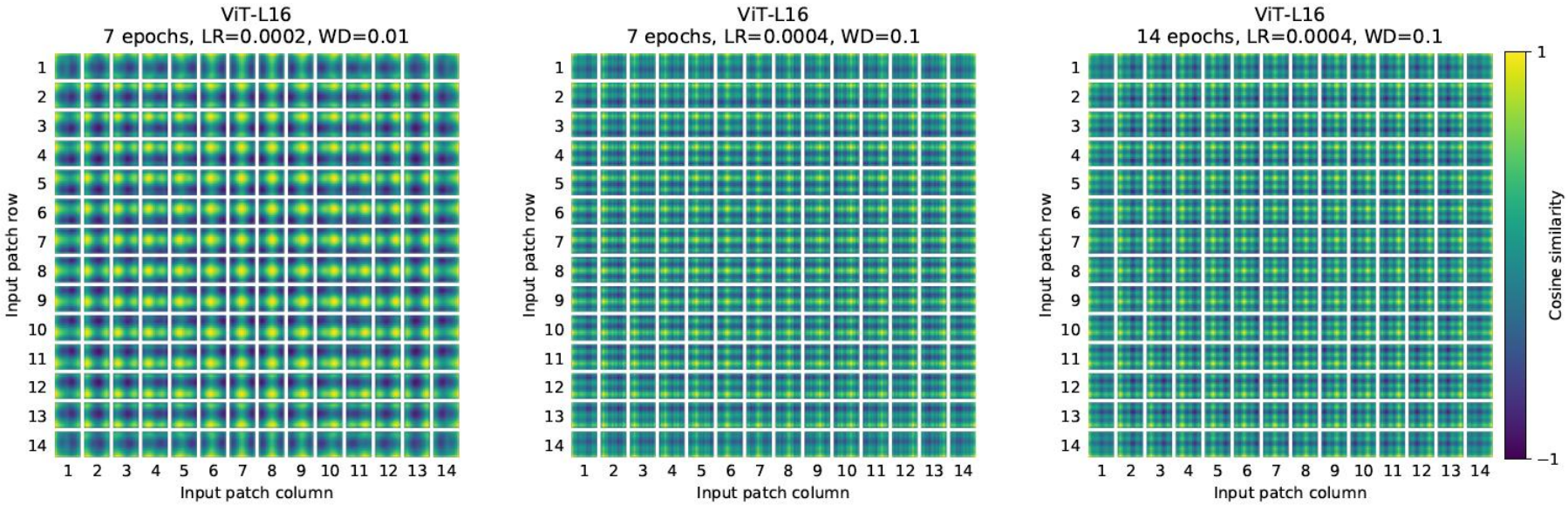


図10：異なるハイパーパラメータで学習したモデルの位置埋め込み。

Attention 距離

ViTがどのようにSelf Attentionを用いて画像全体の情報を統合するかを理解するために、異なるレイヤーにおけるAttentionの重みがまたがる平均距離を分析した（図11）。

あるヘッドは画像の大部分に注意を向け、他のヘッドはクエリー位置やその近くの小さな領域に注意を向ける。

深度が深くなるにつれて、Attention距離はすべてのヘッドで増加する。ネットワークの後半では、ほとんどのヘッドが広くトークンをまたいでAttendした。

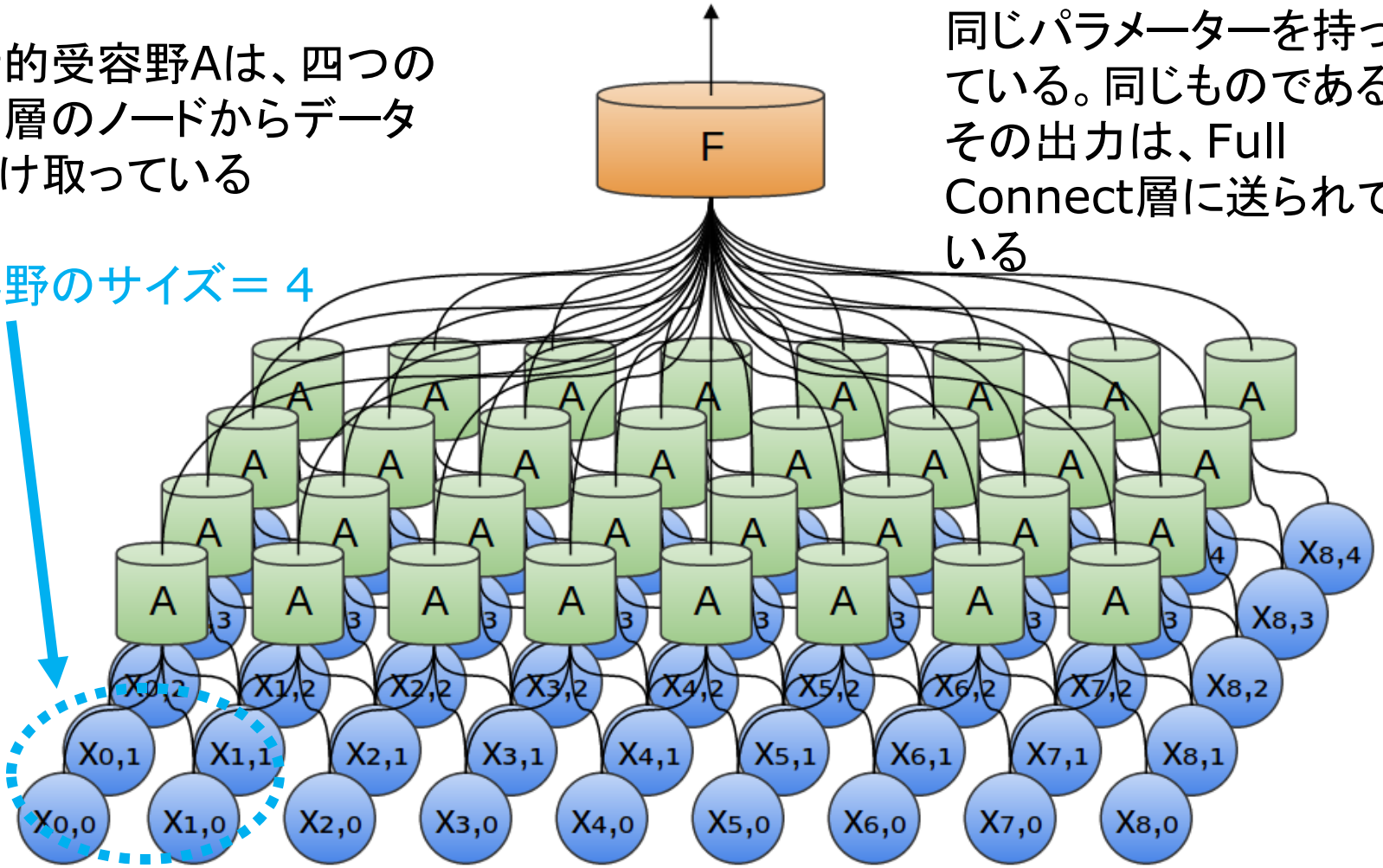
この「Attention距離」はCNNにおける受容野のサイズに類似している。

CNNの局所的受容野

局所的受容野Aは、四つの入力層のノードからデータを受け取っている

局所的受容野Aは、全て同じパラメーターを持っている。同じものである。その出力は、Full Connect層に送られている

受容野のサイズ＝４



その結果、いくつかのヘッドはすでに最下層で画像のほとんどに注目していることがわかり、情報を大域的に統合する能力がモデルによって実際に使われていることが示された。

他のAttentionのヘッドは、低層でのAttentionの距離が一貫して小さい。この高度に局在化したAttentionは、Transformerの前にResNetを適用したハイブリッドモデルではあまり顕著ではなく、CNNの初期の畳み込み層と同様の機能を果たす可能性を示唆している。

さらに、Attention距離はネットワークの深さとともに増加する。

Mean attention distance (pixels)とNetwork depth (layer)

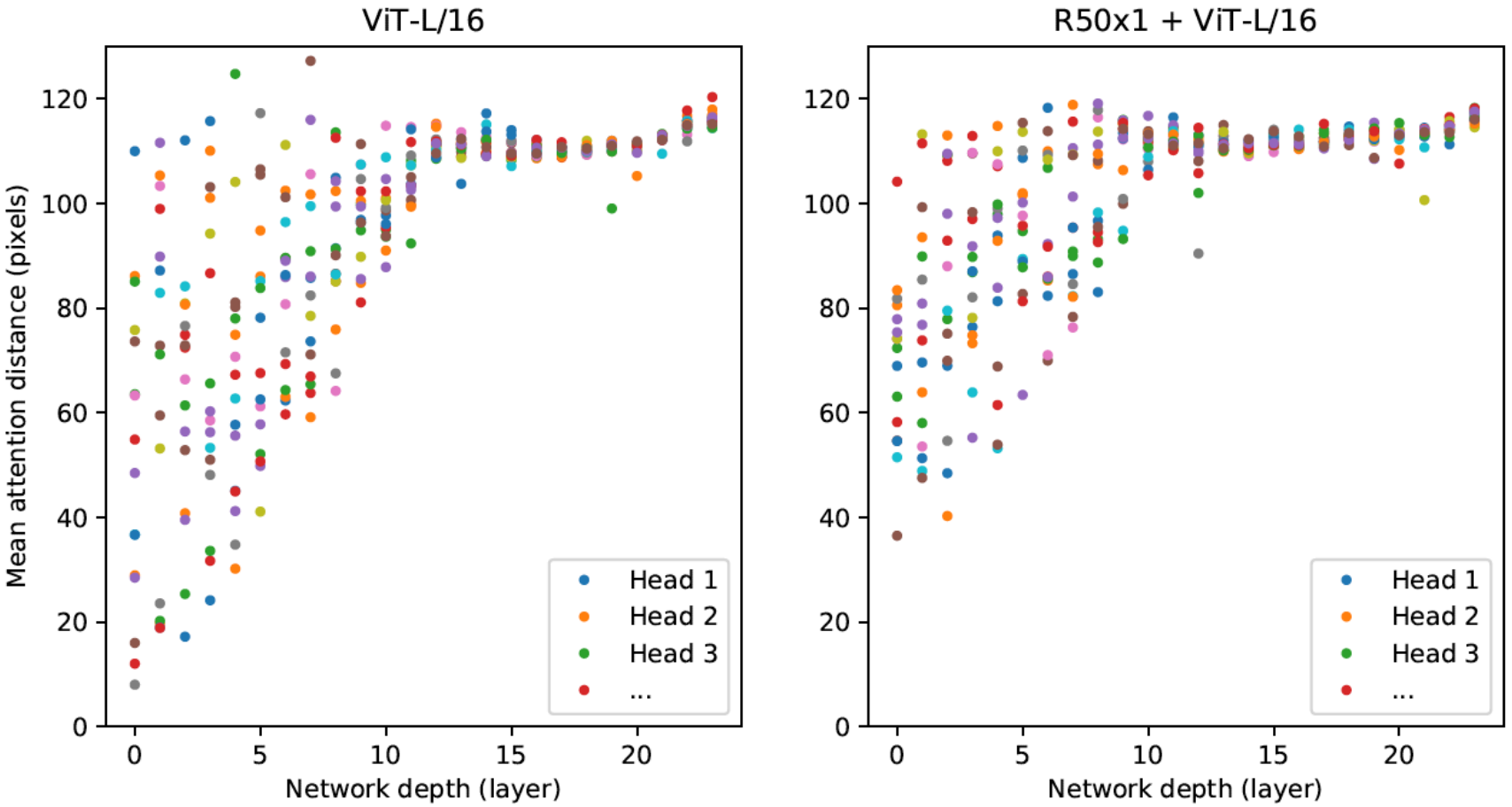


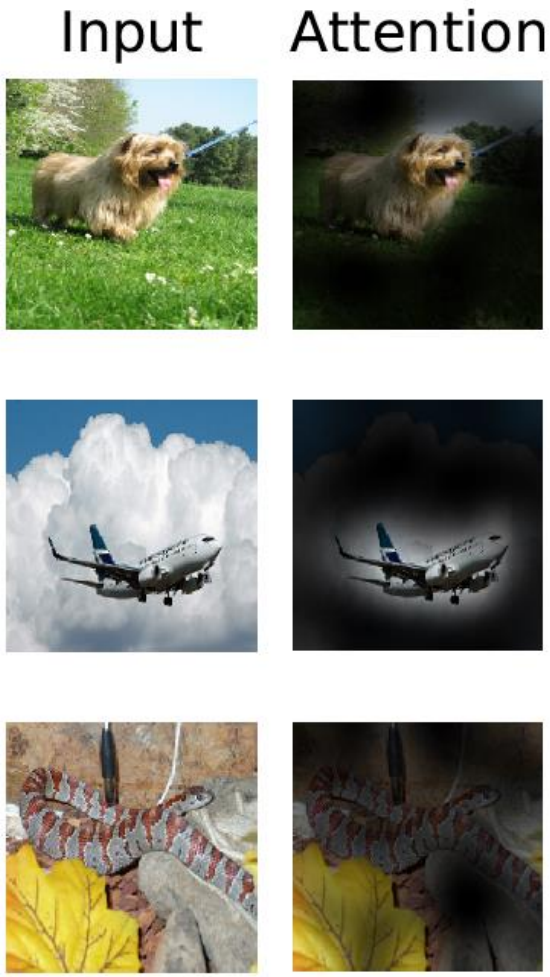
図 11: ヘッドとネットワークの深さによるAttention領域の大きさ。各ドットは1つのレイヤーにおける16個のヘッドの1つに対する画像全体の平均Attention距離を示す。画像幅は224ピクセル。

Attentionが注目する領域

Attentionが注目する領域

全体的に、このモデルは分類に意味的に関連する画像領域に注目することが分かる。

図6：出力トークンから入力空間へのAttentionの代表例。詳細は付録D.7参照。



Attention Map

出力トークンから入力空間へのAttentionのマップを計算するために（図6と14）、Attention Rollout (Abnar & Zuidema, 2020)を使用した。

簡単に説明すると、ViTL/ 16の注意の重みを全ヘッドで平均し、すべての層の重み行列を再帰的に乗算した。これにより、全レイヤーを通してトークン間のアテンションが混在することを考慮する。

図14：図6と同様のアテンション・マップのさらなる例（ランダム選択）。

3

4

11

12

19

20

5

6

13

14

21

22

7

8

15

16

23

24

CNN+Attention
caption生成でのAttentionの利用

A woman is throwing a frisbee in a park.

A dog is standing on a hardwood floor.

A stop sign is on a road with a mountain in the background.

A giraffe standing in a forest with trees in the background.

この論文の結論

成果

我々はTransformersを画像認識に直接応用することを探求してきた。

コンピュータビジョンにおけるSelf Attentionを用いた先行研究とは異なり、我々は最初のパッチ抽出ステップとは別に、画像特有の帰納的バイアスをアーキテクチャに導入しない。

その代わりに、我々は画像をパッチのシーケンスとして解釈し、NLPで用いられるような標準的なTransformerエンコーダによって処理する。

このシンプルかつスケーラブルな戦略は、大規模なデータセットでの事前学習と組み合わせることで、驚くほどうまく機能する。

このように、Vision Transformerは多くの画像分類データセットにおいて、比較的安価に事前学習が可能でありながら、最先端技術に匹敵するか、それ以上の性能を発揮する。

課題

こうした初期の成果は心強いものだが、多くの課題が残されている。

その1つは、検出やセグメンテーションなど、他のコンピュータビジョンタスクにViTを適用することである。

我々の結果は、Carionら（2020）の結果と相まって、このアプローチが有望であることを示している。

End-to-End Object Detection with Transformers
Carion et al. 2020
https://arxiv.org/abs/2005.12872

我々は、オブジェクト検出を直接的な集合予測問題として捉える新しい手法を提示する。我々のアプローチは検出パイプラインを効率化し、タスクに関する事前知識を明示的にエンコードする非最大抑制手順やアンカー生成のような、手作業で設計された多くのコンポーネントの必要性を効果的に取り除く。
DEtection TRansformer（DETR）と呼ばれる新しいフレームワークの主な構成要素は、二分割マッチングによって一意な予測を強制する集合ベースの大域的損失と、変換エンコーダ・デコーダアーキテクチャである。学習されたオブジェクトクエリの固定された小さな集合が与えられたとき、DETRはオブジェクトの関係とグローバルな画像コンテキストを推論し、最終的な予測集合を並列に直接出力する。この新しいモデルは概念的に単純であり、他の多くの最新の検出器とは異なり、特別なライブラリを必要としない。DETRは、難易度の高いCOCO物体検出データセットにおいて、確立され高度に最適化されたFaster RCNNベースラインと同等の精度と実行時間性能を示す。さらに、DETRは総括的なセグメンテーションを統一的な方法で生成するために容易に一般化できる。我々は、DETRが競合ベースラインを大幅に上回ることを示す。

課題

もう一つの課題は、自己教師付き事前学習法の探求を続けることである。

我々の最初の実験では、教師あり事前学習による改善が見られたが、教師あり事前学習と大規模教師あり事前学習との間には、まだ大きな隔たりがある。

最後に、ViTのさらなるスケーリングが性能向上につながる可能性が高い。


Adobe Firefly


Part 3
CLIP
Connecting text and images

CLIP
アプローチ

CLIPとは何か？

CLIP (Contrastive Language–Image Pre-training) は、テキストとイメージを結合することを目指したOpenAIのプロジェクトです。

CLIPは、大規模言語モデルをマルチモーダルな人工知能に展開する上での、OpenAIの中心的なプロジェクトと考えていいと思います。

CLIP: Connecting text and images

OpenAIは、CLIPを次のように紹介しています。

「CLIPと呼ばれるニューラルネットワークを導入し、natural language supervision から視覚概念を効率的に学習する。CLIPは、GPT-2やGPT-3の「ゼロショット」機能と同様に、認識すべき視覚カテゴリの名前を与えるだけで、あらゆる視覚分類ベンチマークに適用できる。」

ここでのキー・コンセプトは、"natural language supervision" です。その意味は、すぐ後で説明します。先の文は、それを視覚概念の学習に活かすと言っています。

https://openai.com/research/clip

CLIP登場の背景

“natural language supervision” の説明の前に、CLIPの登場の背景を見ておきましょう。その背景を、OpenAIは、とても率直に語っています。

「ディープラーニングはコンピュータ・ビジョンに革命をもたらしたが、現在のアプローチにはいくつかの大きな問題がある。」

最大のものは、データセットの問題だとOpenAIは言います。

「また、ベンチマークでは優れた性能を発揮するモデルも、ストレス・テストでは失望するほど低い性能しか発揮できず、コンピュータ・ビジョンへのディープラーニング・アプローチ全体に疑問を投げかけている。」

データセットの問題

データセットには、どういう問題があるのでしょう？

先に見た Vision Transformer は、”Inductive Bias Free”なシンプルなアーキテクチャーでも、データセットの規模を拡大すると、画像認識の性能を上げられることを強調し、「大規模訓練が帰納的バイアスに勝ることを発見した。」と豪語していたのですが、OpenAIのCLIPのアプローチは、すこし違ったものです。

「典型的なビジョン・データセットは、作成に労力とコストがかかる一方で、狭い範囲の視覚概念しか教えない。標準的なビジョン・モデルは、1つのタスクと1つのタスクにしか向いておらず、新しいタスクに適応させるためには多大な労力を必要とする。」

例えば、代表的なビジョン・データセットであるImageNetは、一つの画像には一つのカテゴリー（ラベル）が割り当てられています。ただ、象の画像には「象」というラベルがつけられているだけです。

このようなデータセットで訓練されたシステムは、提示された画像が「なんであるか」という質問には答えられるかもしれませんが、それ以外のタスクには、対応できません。

しかし、画像認識には多様なタスクが存在します。

「そこはどこ？」

「何をしているの？」

“natural language supervision”とは何か？

「我々はこのような問題を解決することを目的としたニューラルネットワークを発表する。」

それがCLIPだといいます。

「それは、インターネット上に豊富に存在する多種多様なnatural language supervisionを用いて、多種多様な画像で学習される。これは重要な変更点である。」

例えば、象がいる画像に「象」というラベルを人手で一つつけるのではなく、その画像に「動物園で象がリンゴを食べている」というテキストを対応させ、この画像とテキストのペアを画像データセットに加えるのです。こうしたデータは、インターネット上にはたくさん存在します。

“natural language supervision” というのは、自然言語で書かれたテキストの意味を抽出して、その管理下で画像処理のタスク処理を行うということです。それは、画像の「意味」を対応するテキストが与えると考えることです。

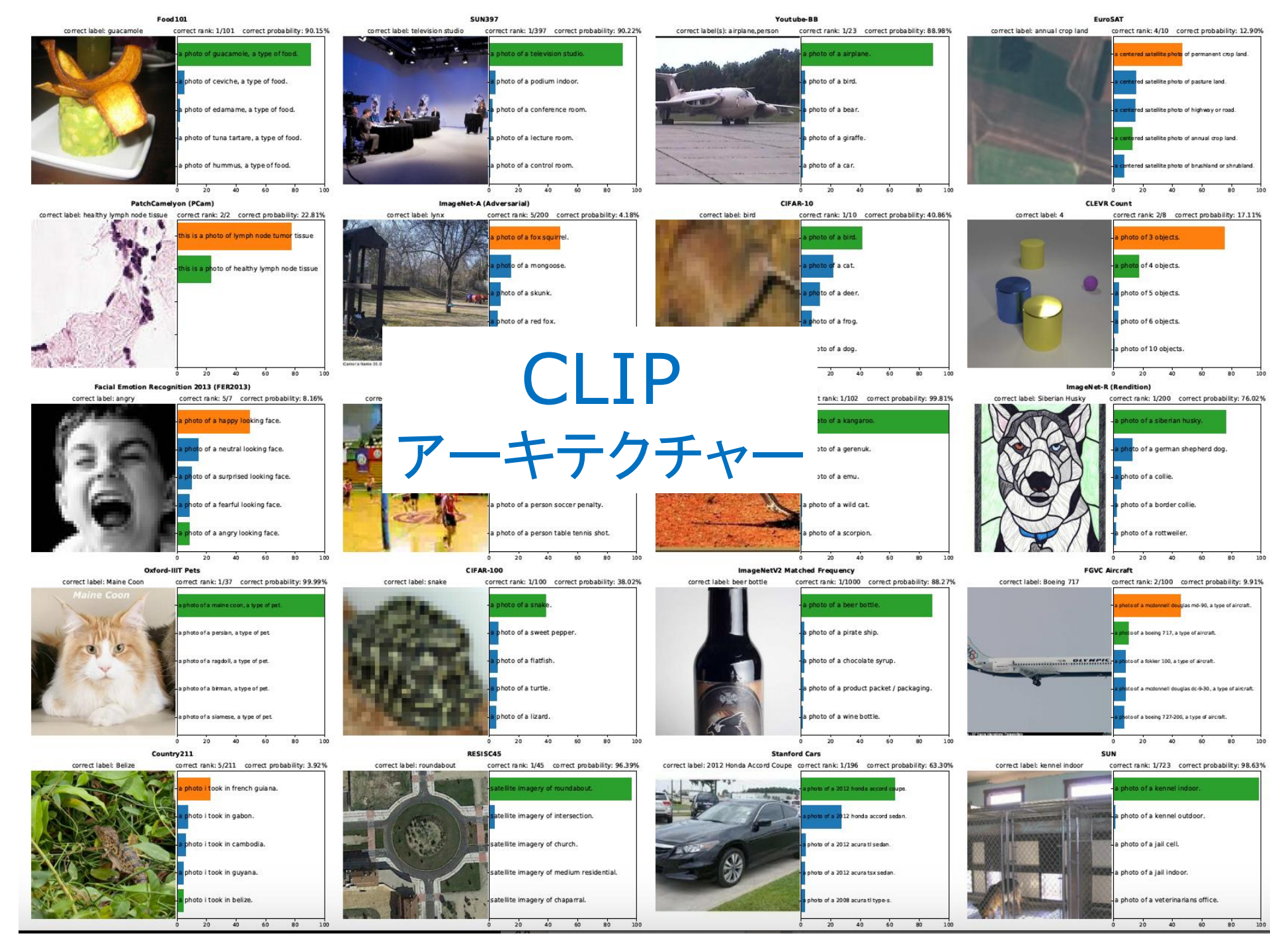
（どう日本語に訳すのがいいか難しいですが、「自然言語監督」でもいいかなとも思っています。）

従来のImageNETのようなデータセットでは、「この画像は何？」という画像認識タスクに対して「象」と答えるしかなかったのですが、この新しいデータセットでは、「ここはどこ？」 -> 「動物園」、「何をしているの」 -> 「リンゴを食べている」のように、複数の画像認識タスクに対応できます。

細かいことですが、画像と対応付けられたテキストは、もはや単なる分類カテゴリー名のラベルではないので、単純なデフォールトでは「〇〇の画像」あるいは「〇〇の写真」と記述した方がいいことになります。これはこれで、ベタですが画像の意味記述になっています。

Vision Transformer は、大規模言語モデルのTransformerという処理スタイルの形式を画像処理に転用したのですが、CLIPは、もう少し深いところで、大規模言語モデルの自然言語の「意味理解」の能力を画像認識に活かそうとしているように見えます。

実際は、もう少し複雑なのですが（というか、いろいろな単純化がなされているのですが）、CLIPのアーキテクチャーを論文にそくして見ていきたいと思います。



CLIP
アーキテクチャー

Learning Transferable Visual Models From Natural Language Supervision

AlecRadford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, Ilya Sutskever,

https://arxiv.org/pdf/2103.00020v1.pdf
2021年

Abstract

最先端のコンピュータビジョンシステムは、あらかじめ決められたオブジェクトカテゴリの固定セットを予測するように訓練されている。

この制限された監視形式は、他の視覚的概念を指定するために追加のラベル付きデータが必要であるため、その汎用性と使いやすさを制限している。

画像に関する生のテキストから直接学習することは、より広範な監督源を活用する有望な代替手段である。

我々は、どの画像にどのキャプションが合うかを予測するという単純な事前学習タスクが、インターネットから収集した4億の（画像とテキストの）ペアのデータセットに対して、最先端の画像表現をゼロから学習する効率的でスケーラブルな方法であることを実証する。

事前学習後、自然言語を用いて学習した視覚的概念を参照する（あるいは新しい概念を記述する）ことで、モデルを下流のタスクにゼロショットで転送することが可能となる。

はじめに
自然言語処理の発展

生のテキストから直接学習する事前学習法は、ここ数年で自然言語処理に革命をもたらした。

自己回帰モデリングやマスク言語モデリングなどのタスクにとらわれない手法は、計算量、モデル容量、データの何桁にもわたってスケールし、着実に能力を向上させてきた。

標準化された入出力インターフェースとしての「テキストからテキストへ」の手法の発展により、タスクにとらわれないアーキテクチャは、特殊な出力ヘッドやデータセット固有のカスタマイズの必要性を排除し、下流のデータセットへのゼロショット転送が可能になった。

GPT-3 (Brown et al., 2020)のようなフラッグシップ・システムは、データセット固有の学習データをほとんど必要としない一方で、特注モデルによる多くのタスクで競争力を持つようになった。

これらの結果は、ウェブスケールのテキストコレクション内で最新の事前学習手法にアクセス可能な総監視量は、高品質のクラウドラベル付き自然言語処理データセットのそれを上回ることを示唆している。

はじめに
画像処理処理の現状

しかし、コンピュータビジョンのような分野では、ImageNetのようなクラウドラベル付きデータセットでモデルを事前学習するのが標準的なやり方である。

ウェブテキストから直接学習するスケーラブルな事前学習法は、コンピュータビジョンにおいても同様のブレークスルーをもたらすのだろうか？

さまざまな、有望な先行研究がある。

はじめに
本研究の課題

本研究では、大規模なnatural language supervisionで訓練された画像分類器の振る舞いを研究する。

インターネット上で公開されている大量のこの形式のデータを利用し、4億の（画像とテキストの）ペアからなる新しいデータセットを作成し、ゼロから学習したCLIP（Contrastive Language-Image Pre-training）が、natural language supervisionから学習する効率的な手法であることを実証する。

我々は、ほぼ2桁の計算量に及ぶ一連の8つのモデルを訓練することによってCLIPのスケーラビリティを研究し、転送性能が計算量の滑らかに予測可能な関数であることを観察する。

はじめに
成果

- 我々は、CLIPがOCR、ジオロカライゼーション、行動認識、その他多くのタスクを含む幅広いタスクを事前学習中に学習することを発見した。

- 先行するタスク固有の教師ありモデルと競合できることを発見した。CLIPが計算効率に優れながら、公開されている最良のImageNetモデルを上回ることを示す。

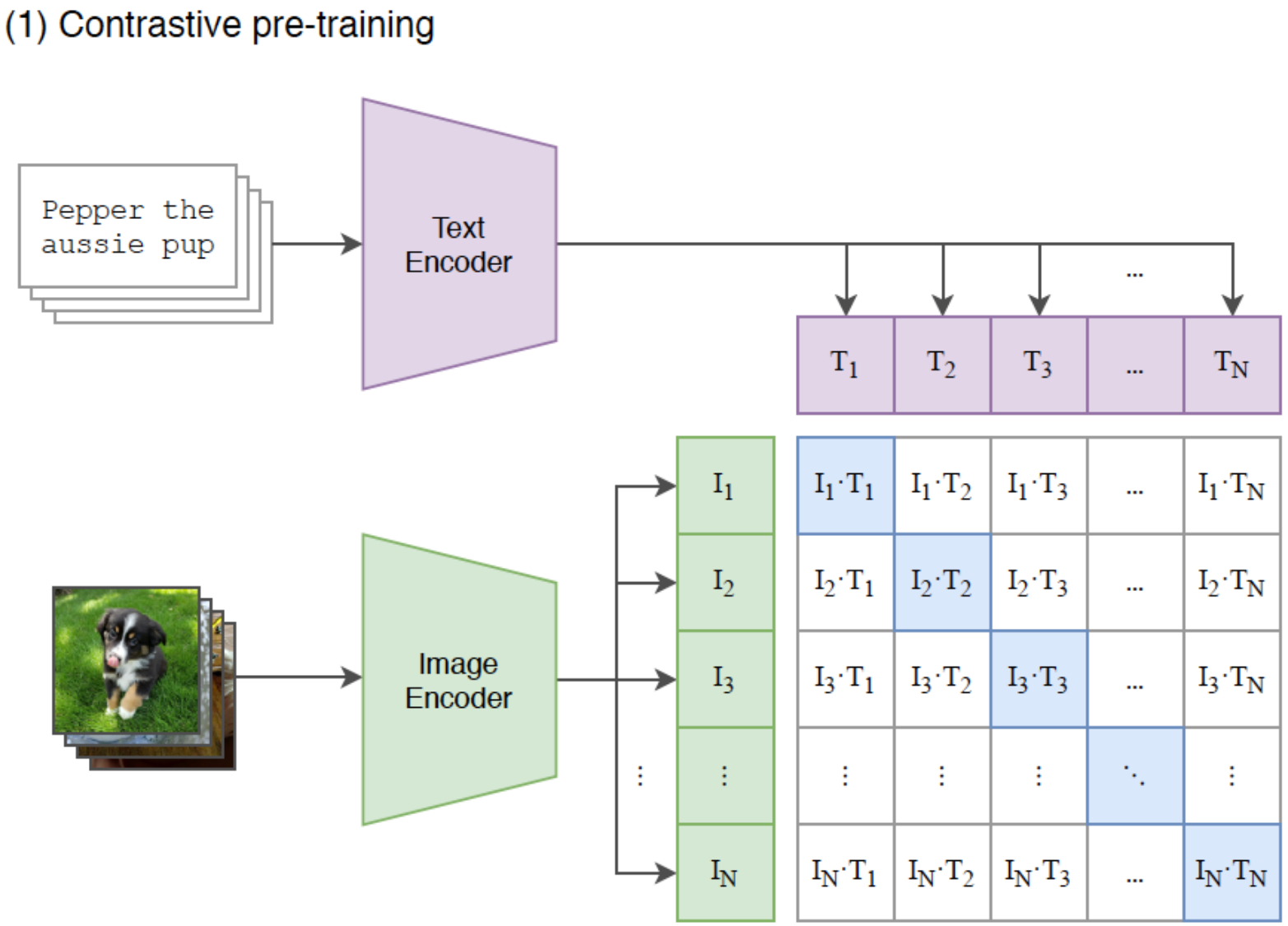
- さらに、ゼロショットCLIPモデルは、同等の精度を持つ教師ありImageNetモデルよりもはるかにロバストであることを発見した。

我々のアプローチの概要

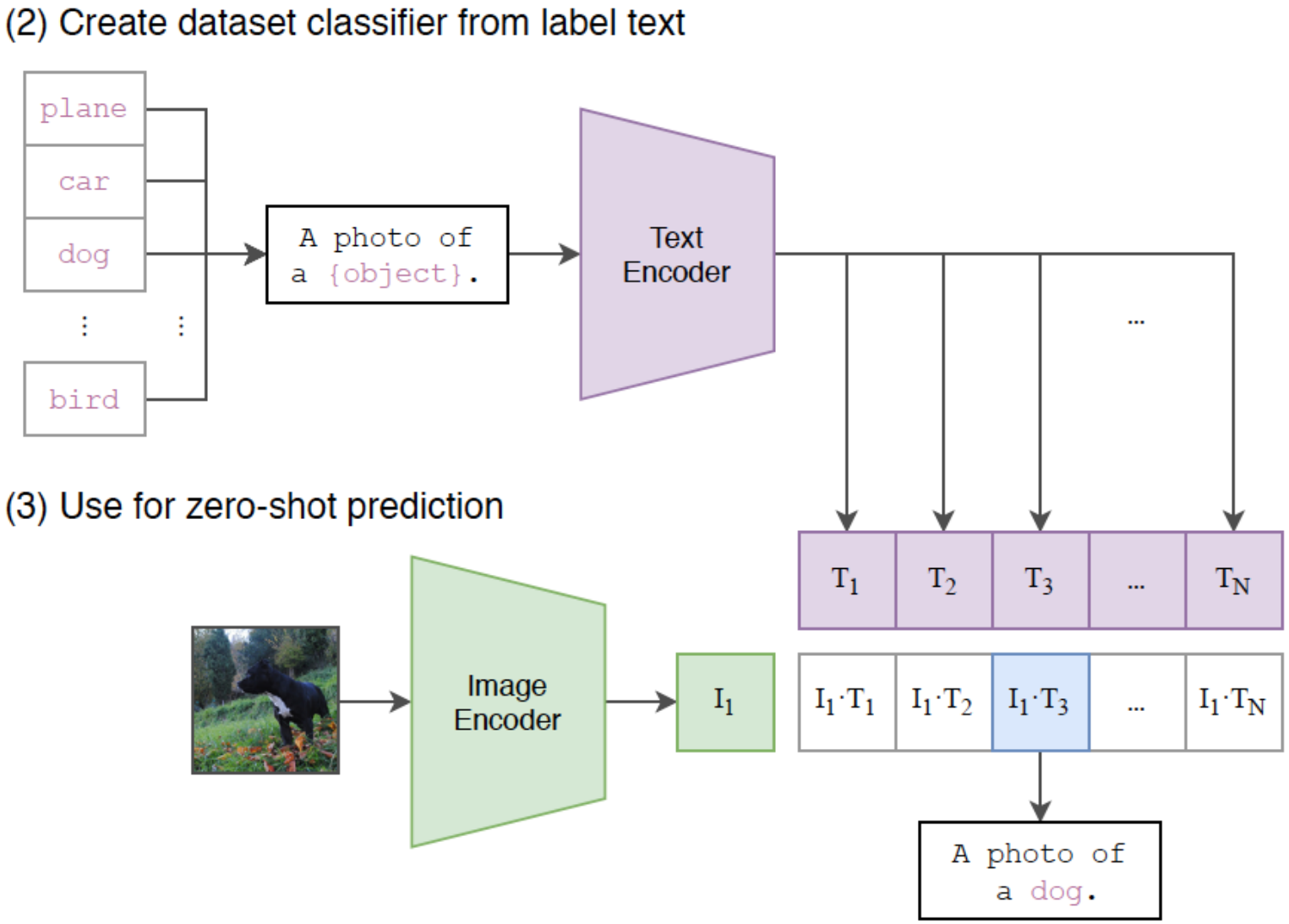
CLIPは画像エンコーダとテキストエンコーダを共同で学習し、（画像とテキストの）バッチ学習例の正しいペアリングを予測する。

テスト時に、学習されたテキストエンコーダは、ターゲットデータセットのクラスの名前や説明を埋め込むことで、ゼロショットの線形分類器を合成する。

(1) Contrastive pre-training



(2) Create dataset classifier from label text
(3) Use for zero-shot prediction



アプローチ
Natural Language Supervision

我々のアプローチの核心は、自然言語に含まれる監督情報から知覚を学習するという考え方である。

これは全く新しいアイデアではないが、この領域での研究を説明するために使用される用語は様々で、一見矛盾しているようにさえ見え、述べられた動機も多様である。

Zhangら(2020)、Gomezら(2017)、Joulinら(2016)、Desai & Johnson(2020)はいずれも、画像と対になったテキストから視覚表現を学習する手法を紹介しているが、それぞれのアプローチを教師なし、自己教師あり、弱教師あり、教師ありと表現している。

私たちは、この一連の研究に共通しているのは、使用されている特定の手法の詳細ではなく、自然言語を学習の信号として評価していることであることを強調する。

これらのアプローチはすべて、Natural Language Supervisionから学習している。初期の研究では、トピックモデルとn-gram表現を使用する場合、自然言語の複雑さと格闘していたが、深い文脈表現学習の改善は、我々は現在、この豊富な監督ソースを効果的に活用するためのツールを持っていることを示唆している（McCannら、2017）。

自然言語からの学習には、他の学習方法と比較していくつかの潜在的な強みがある。画像分類のための標準的なクラウドソースラベリングと比較して、自然言語監視のスケールははるかに簡単です。なぜなら、正統的な1対Nの多数決「ゴールドラベル」のような古典的な「機械学習互換フォーマット」である注釈を必要としないからです。

その代わりに、自然言語で動作するメソッドは、インターネット上の膨大な量のテキストに含まれる監視から受動的に学習することができる。

また、自然言語からの学習は、「単に」表現を学習するだけでなく、その表現を言語と結びつけることで、柔軟なゼロショット転送を可能にするという点で、ほとんどの教師なし学習や自己教師あり学習アプローチよりも重要な利点がある。

CLIP
データセットと予測サンプル

大規模データセットの構築と CLIPの性能を見る

このセッションでは、CLIPがどのようなデータセットを訓練用データを構築したのか、また、CLIPがどのような性能を持つかを見ていこうと思います。

CLIPの基本的なアイデアの一つは、さまざまな画像認識タスクをを訓練する大規模なデータセットを、インターネット上に大量に存在するテキストと画像のペアから構築しようということです。

そこでは、natural language supervision によるテキストが与える画像の解釈が重要な役割を果たします。

CLIPの訓練用データセット

従来の研究で利用されたデータセット

既存のコンピュータビジョンの研究では、主にMS-COCO、Visual Genome、YFCC100M の3つのデータセットを使用している。

MS-COCOとVisual Genomeは高品質のクラウドラベル付きデータセットであるが、それぞれ約10万枚のトレーニング写真と現代の基準からすると小規模である。

YFCC100Mは1億枚の写真からなる。ただ、各画像のメタデータはまばらで、品質も様々である。多くの画像は、20160716 113957.JPGのような自動生成されたファイル名を "タイトル "として使用していたり、カメラの設定の "説明 "を含んでいる。

英語の自然言語のタイトルや説明文を持つ画像だけを残すようにフィルタリングした結果、YFCC100Mデータセットは6分の1に縮小され、わずか1500万枚の写真になった。これはImageNetとほぼ同じサイズである。

その他のデータセット

ここにあげた27個のデータセットは、CLIPの性能評価に利用される

Dataset	Classes	Train size	Test size	Evaluation metric
Food-101	102	75,750	25,250	accuracy
CIFAR-10	10	50,000	10,000	accuracy
CIFAR-100	100	50,000	10,000	accuracy
Birdsnap	500	42,283	2,149	accuracy
SUN397	397	19,850	19,850	accuracy
Stanford Cars	196	8,144	8,041	accuracy
FGVC Aircraft	100	6,667	3,333	mean per class
Pascal VOC 2007 Classification	20	5,011	4,952	11-point mAP
Describable Textures	47	3,760	1,880	accuracy
Oxford-IIIT Pets	37	3,680	3,669	mean per class
Caltech-101	102	3,060	6,085	mean-per-class
Oxford Flowers 102	102	2,040	6,149	mean per class

ここにあげた27個のデータセットは、CLIPの性能評価に利用される

Dataset	Classes	Train size	Test size	Evaluation metric
MNIST	10	60,000	10,000	accuracy
Facial Emotion Recognition 2013	8	32,140	3,574	accuracy
STL-10	10	1000	8000	accuracy
EuroSAT	10	10,000	5,000	accuracy
RESISC45	45	3,150	25,200	accuracy
GTSRB	43	26,640	12,630	accuracy
KITTI	4	6,770	711	accuracy
Country211	211	43,200	21,100	accuracy
PatchCamelyon	2	294,912	32,768	accuracy
UCF101	101	9,537	1,794	accuracy
Kinetics700	700	494,801	31,669	mean(top1, top5)
CLEVR Counts	8	2,000	500	accuracy
Hateful Memes	2	8,500	500	ROC AUC
Rendered SST2	2	7,792	1,821	accuracy
ImageNet	1000	1,281,167	50,000	accuracy

CLIPはどのようなデータセットで訓練されたのか

natural language supervisionの主な動機は、インターネット上で公開されている大量のデータである。既存のデータセットはこの可能性を十分に反映していないため、それらのデータセットのみでの結果を考慮することは、この研究分野の可能性を過小評価することになる。

この問題に対処するため、我々はインターネット上の様々な公開ソースから収集した4億組の（画像、テキスト）データセットを新たに構築した。

可能な限り幅広い視覚的概念をカバーするため、構築プロセスの一環として、テキストが50万件のクエリのいずれかを含む（画像、テキスト）ペアを検索した。

クエリごとに最大2万件の（画像、テキスト）ペアを含めることで、結果をおおよそクラスバランスさせた。

得られたデータセットの総語数は、GPT-2の学習に使用したWebTextデータセットとほぼ同じである。

このデータセットをWebImageTextのWITと呼ぶ。

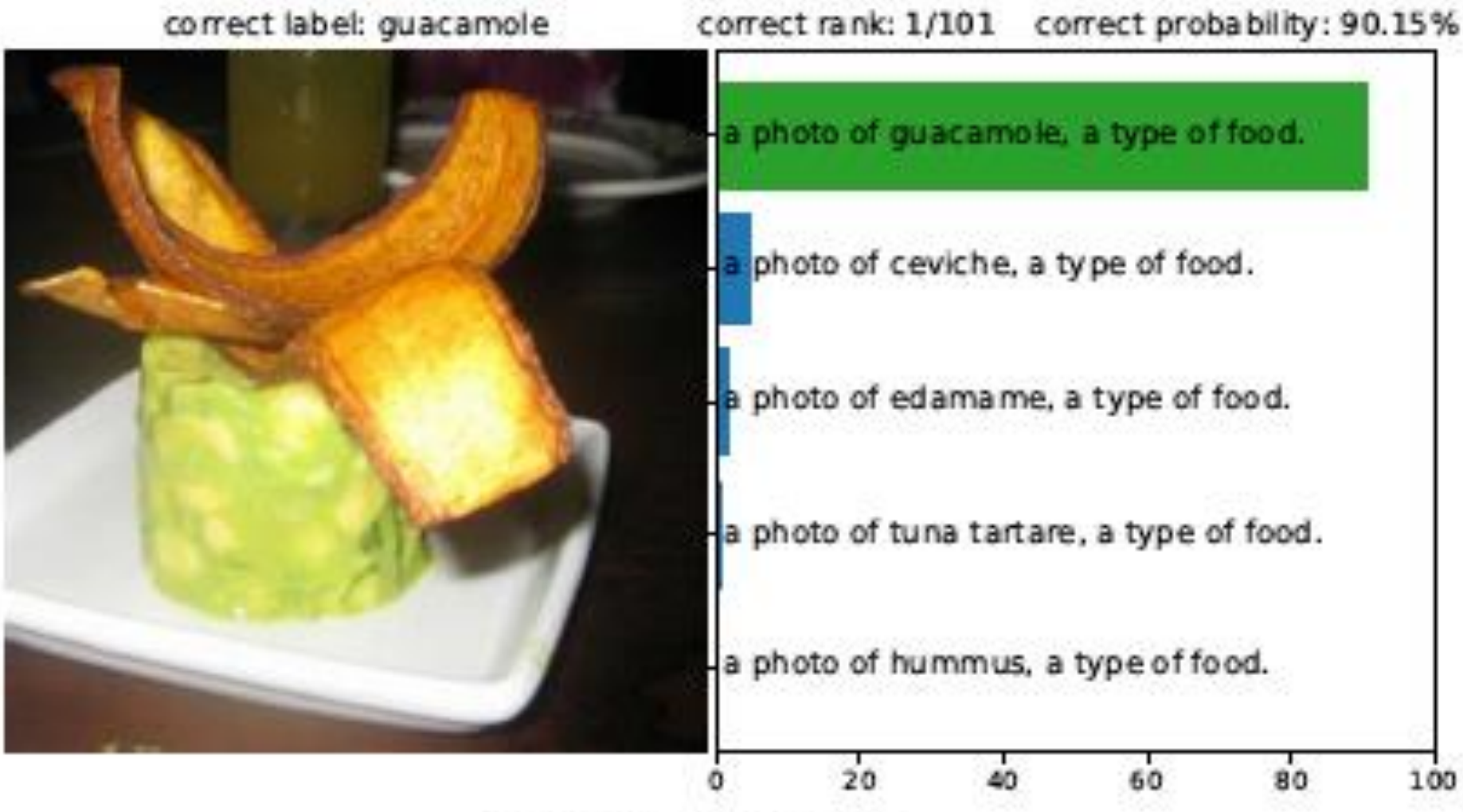
CLIPの予測の視覚化

上位 5 クラスの予測確率が、クラスを表現するために使用されたテキストとともに示される。

グランドトゥルースのラベルは緑色で表示され、間違った予測はオレンジ色で表示されている。

a photo of guacamole, a type of food.

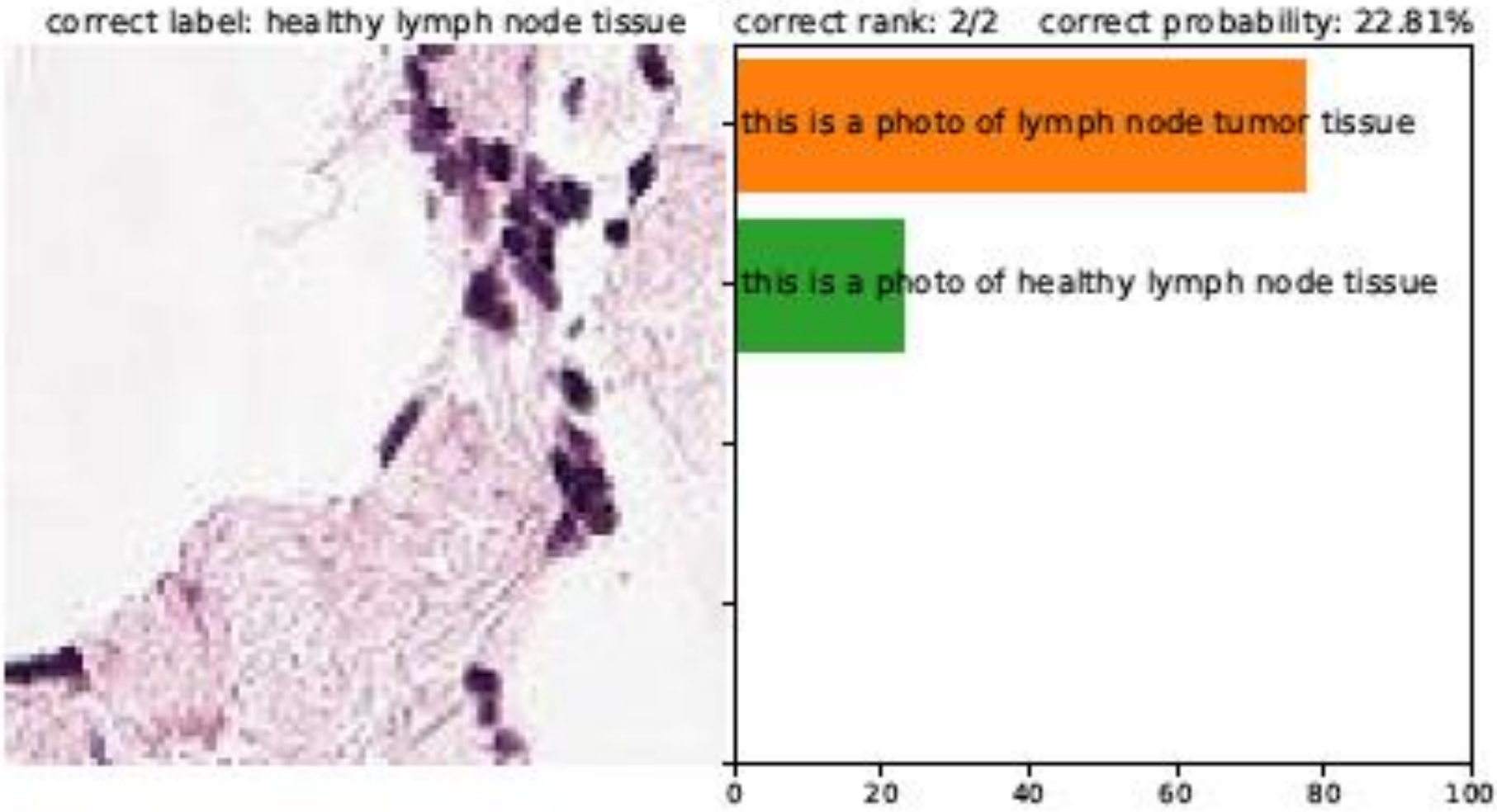
Food101



this is a photo of lymph node tumor tissue

this is a photo of healthy lymph node tissue

PatchCamelyon (Pcam)



a photo of a happy looking face.

photo of a angry looking face.

Facial Emotion Recognition 2013 (FER2013)

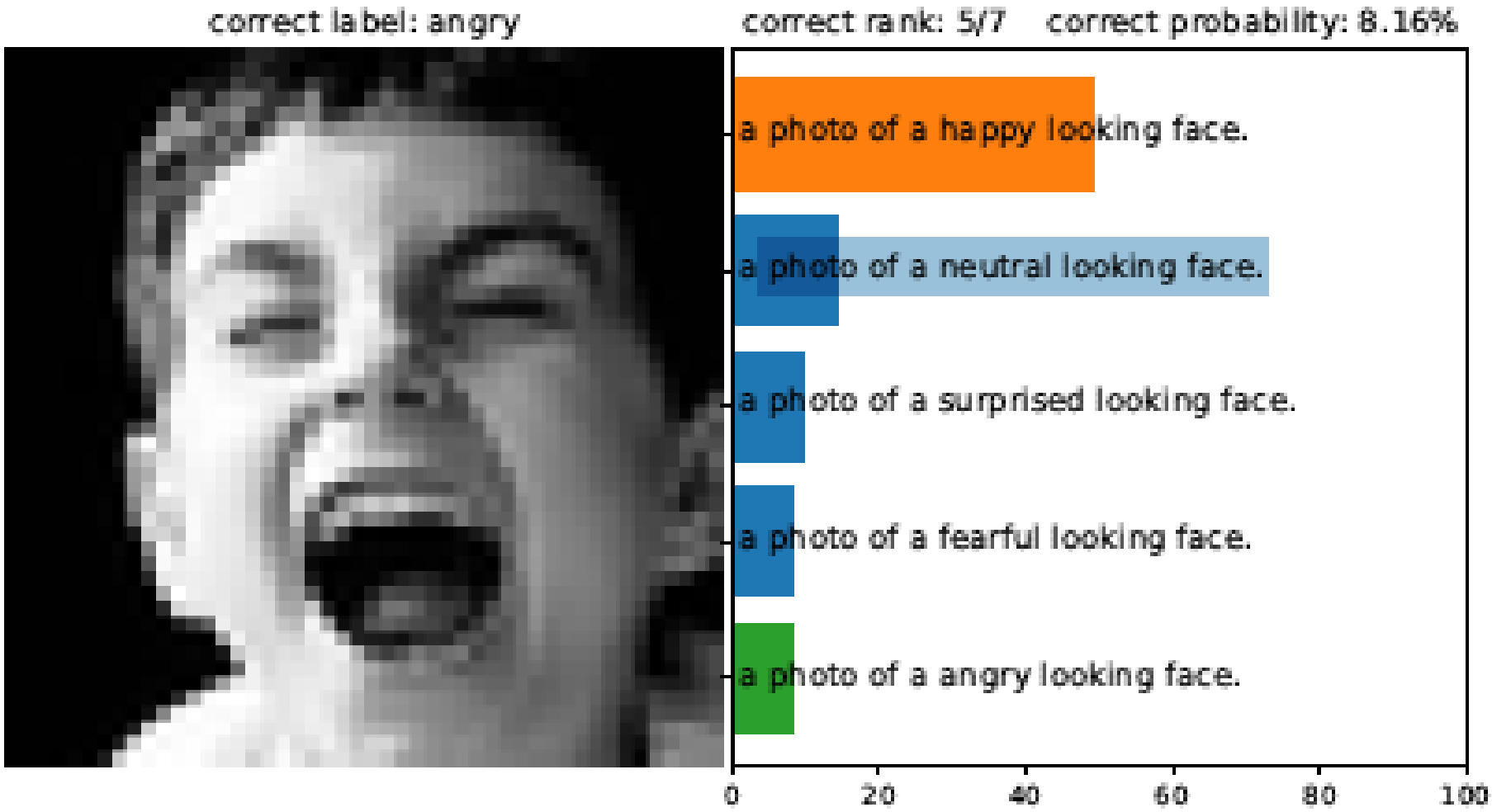
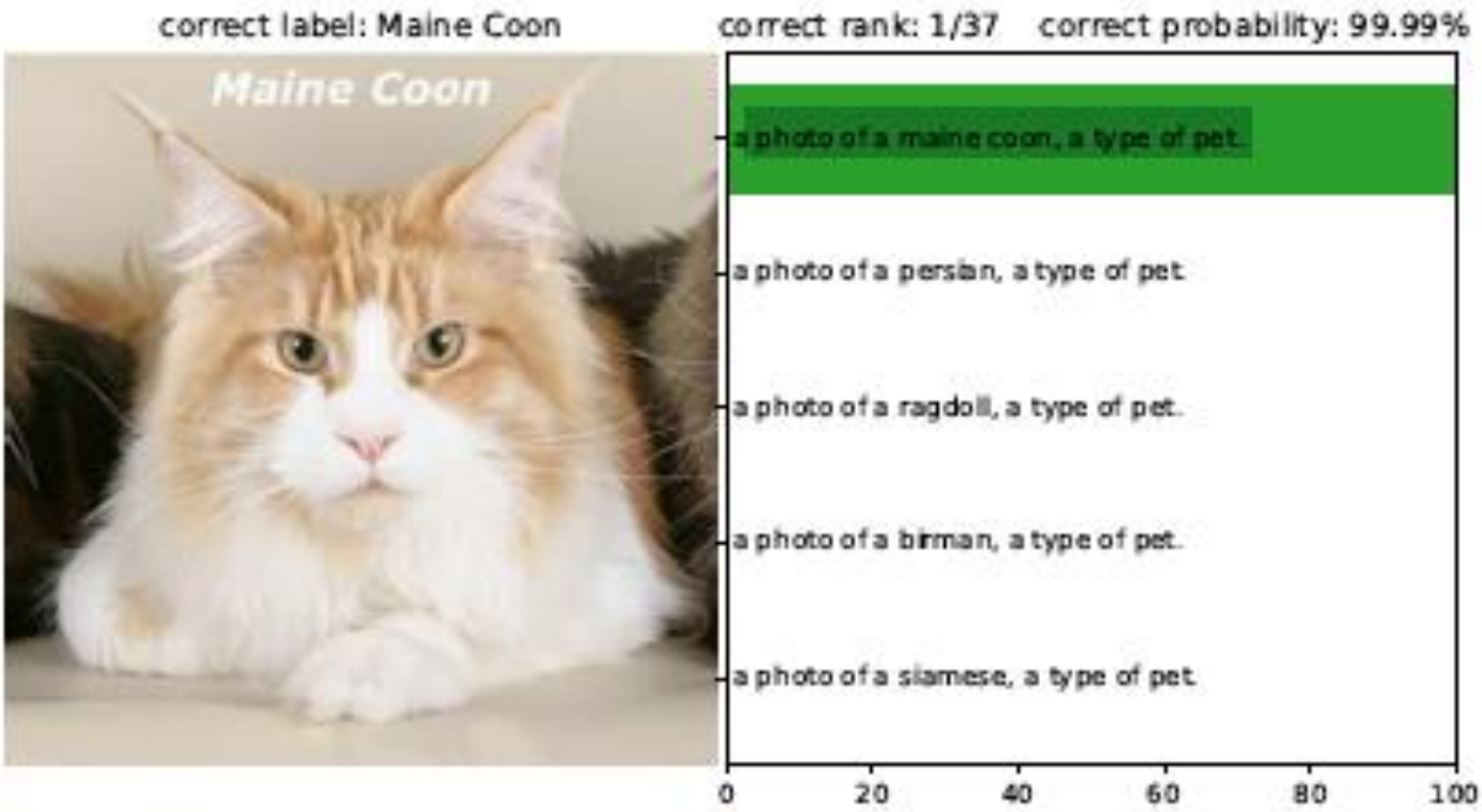


photo of a maine coon, a type of pet.

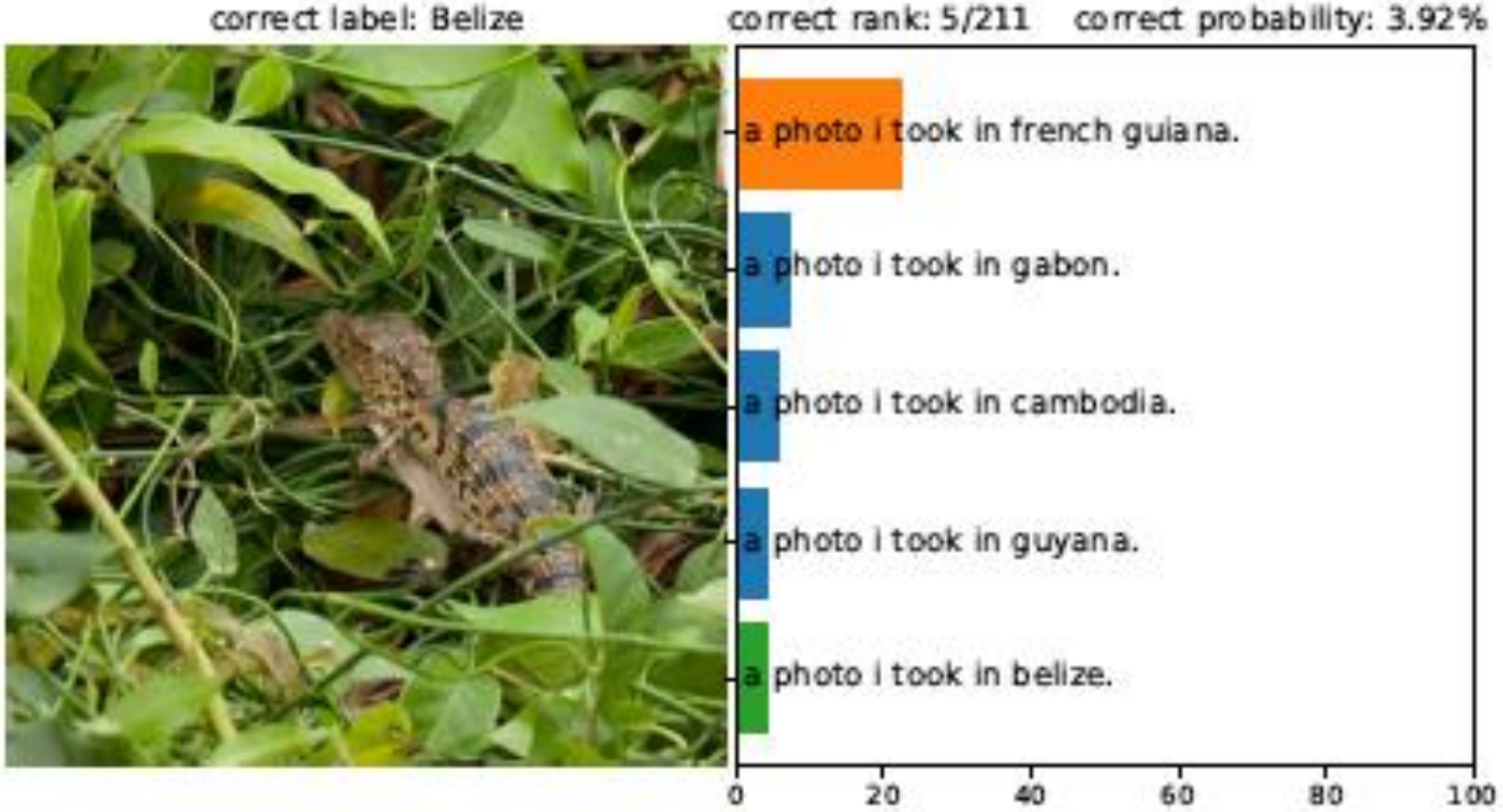
Oxford-IIIT Pets



a photo i took in french guiana.

a photo i took in belize.

Country211

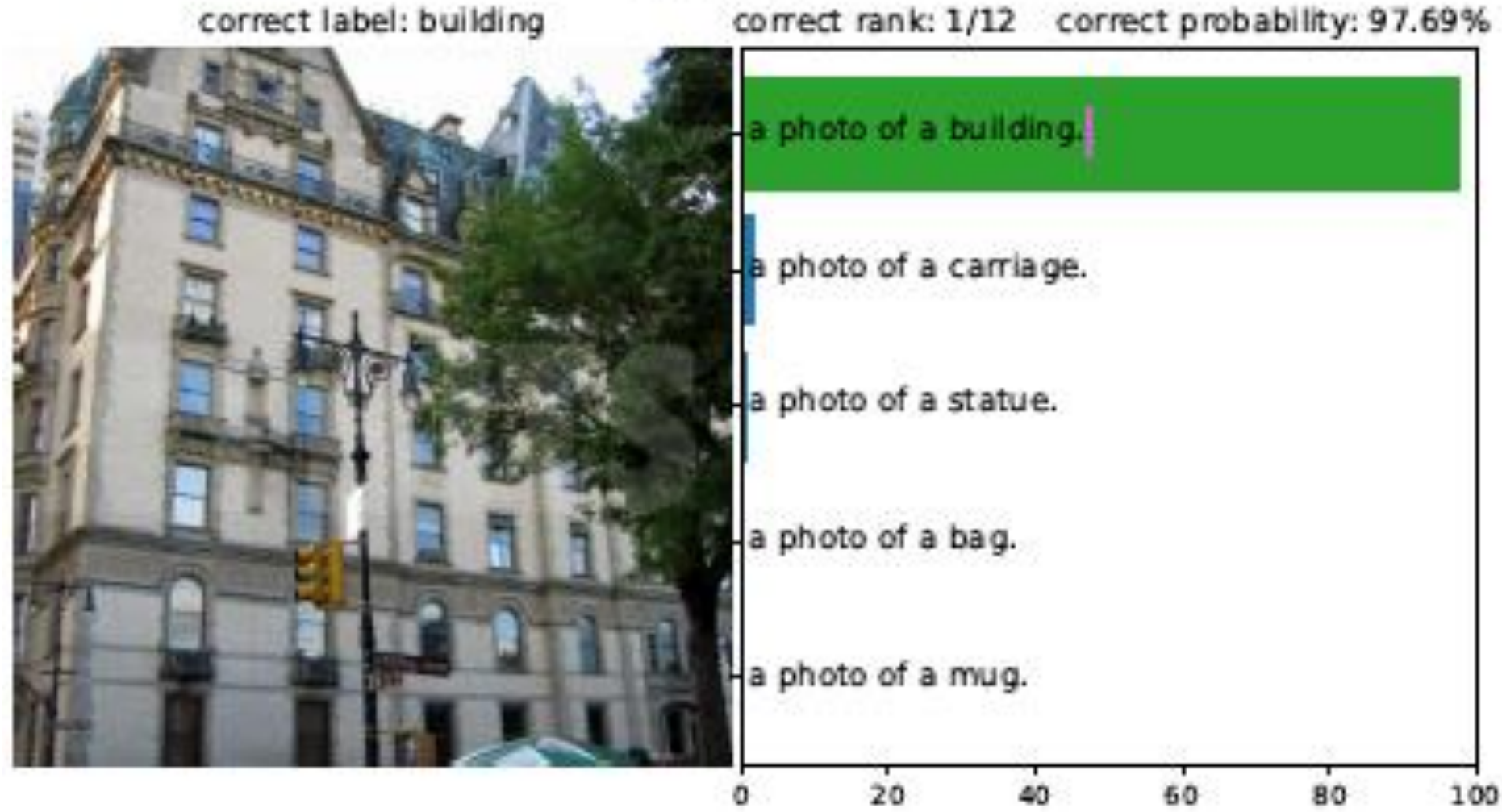


a photo of country line dancing.

Kinetics-700

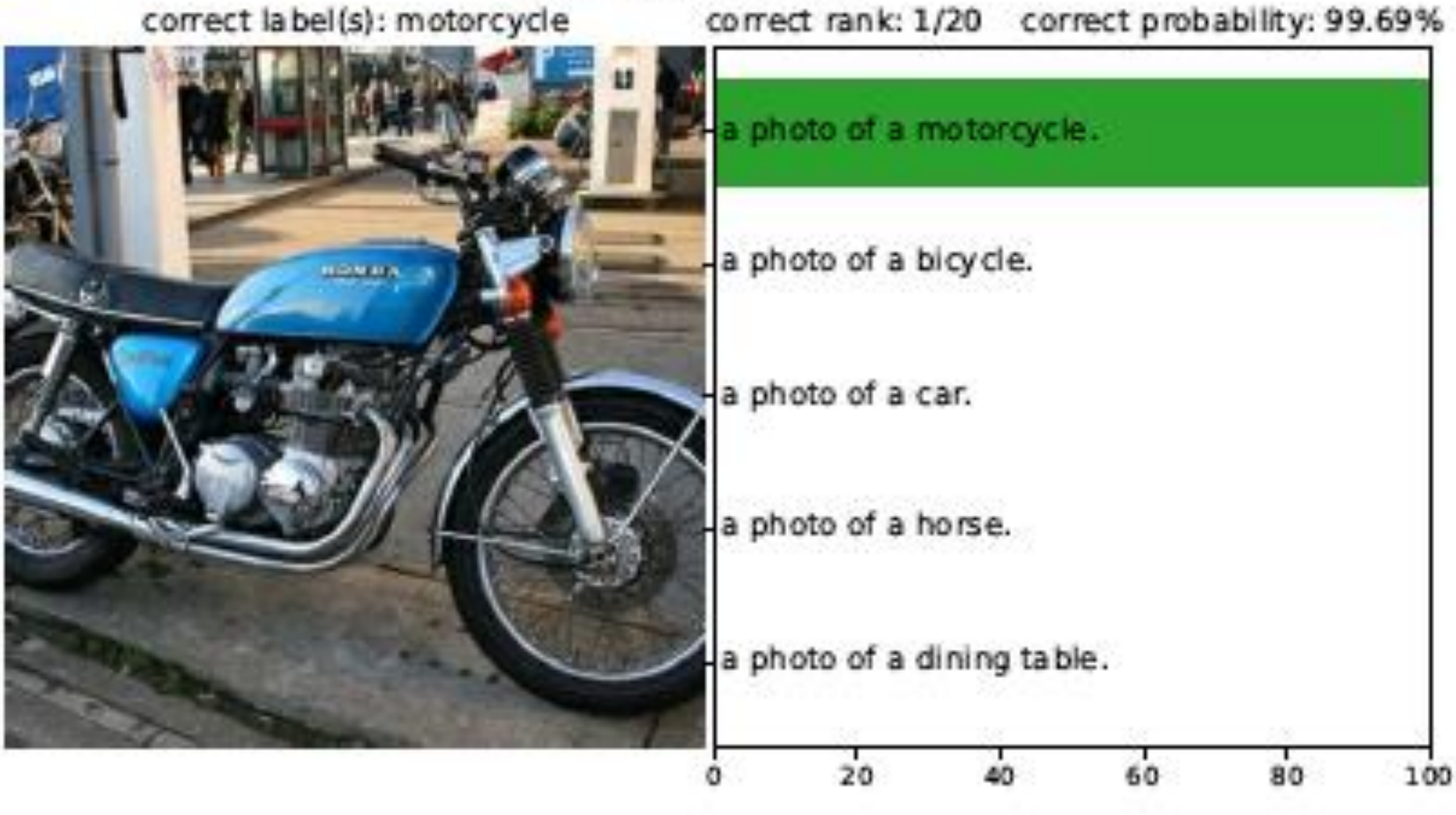
a photo of a building.

aYaho



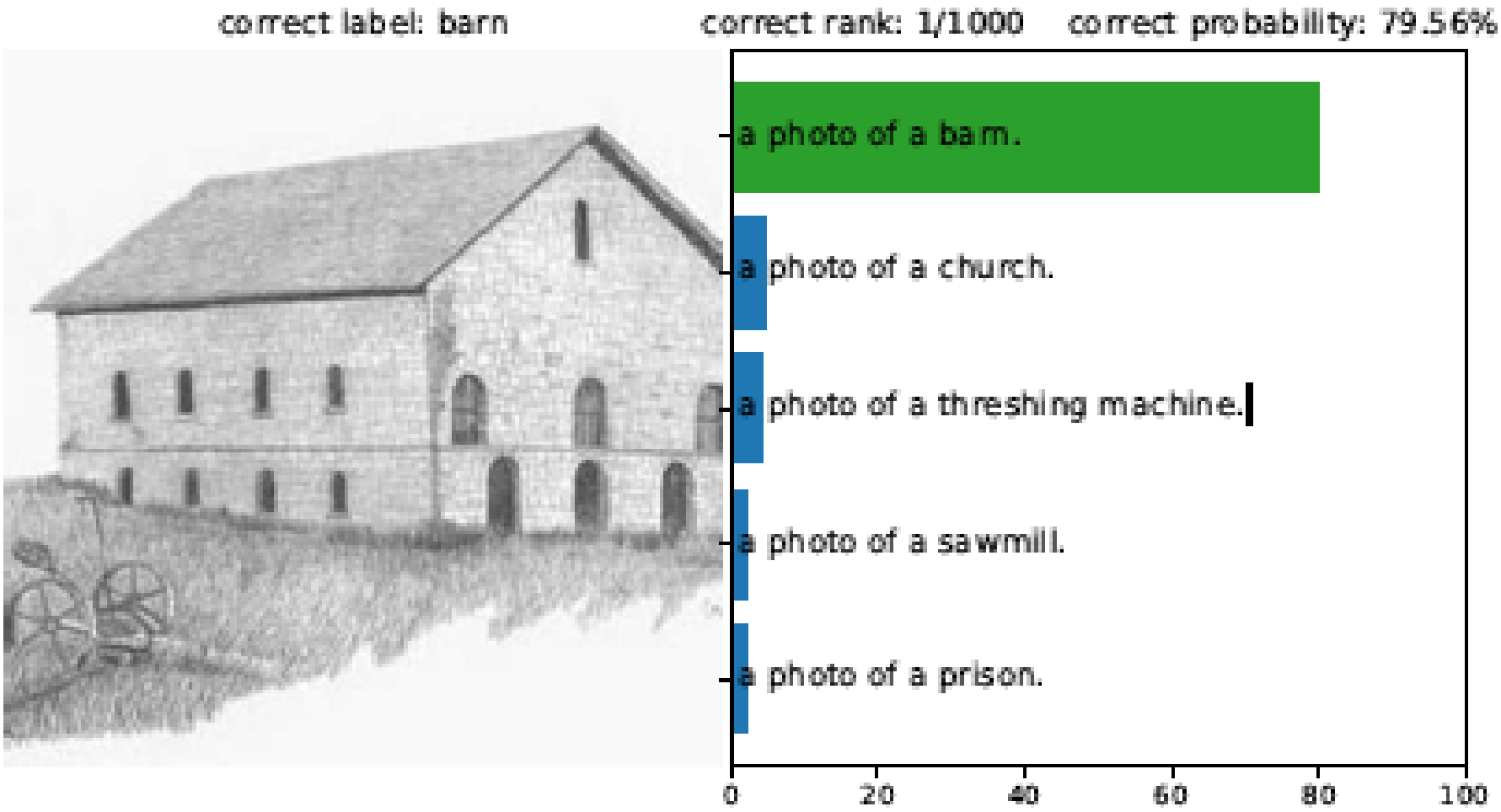
a photo of a motorcycle.

PASCAL VOC 2007



a photo of a barn.

ImageNet Sketch



a photo of a television studio.

SUN397

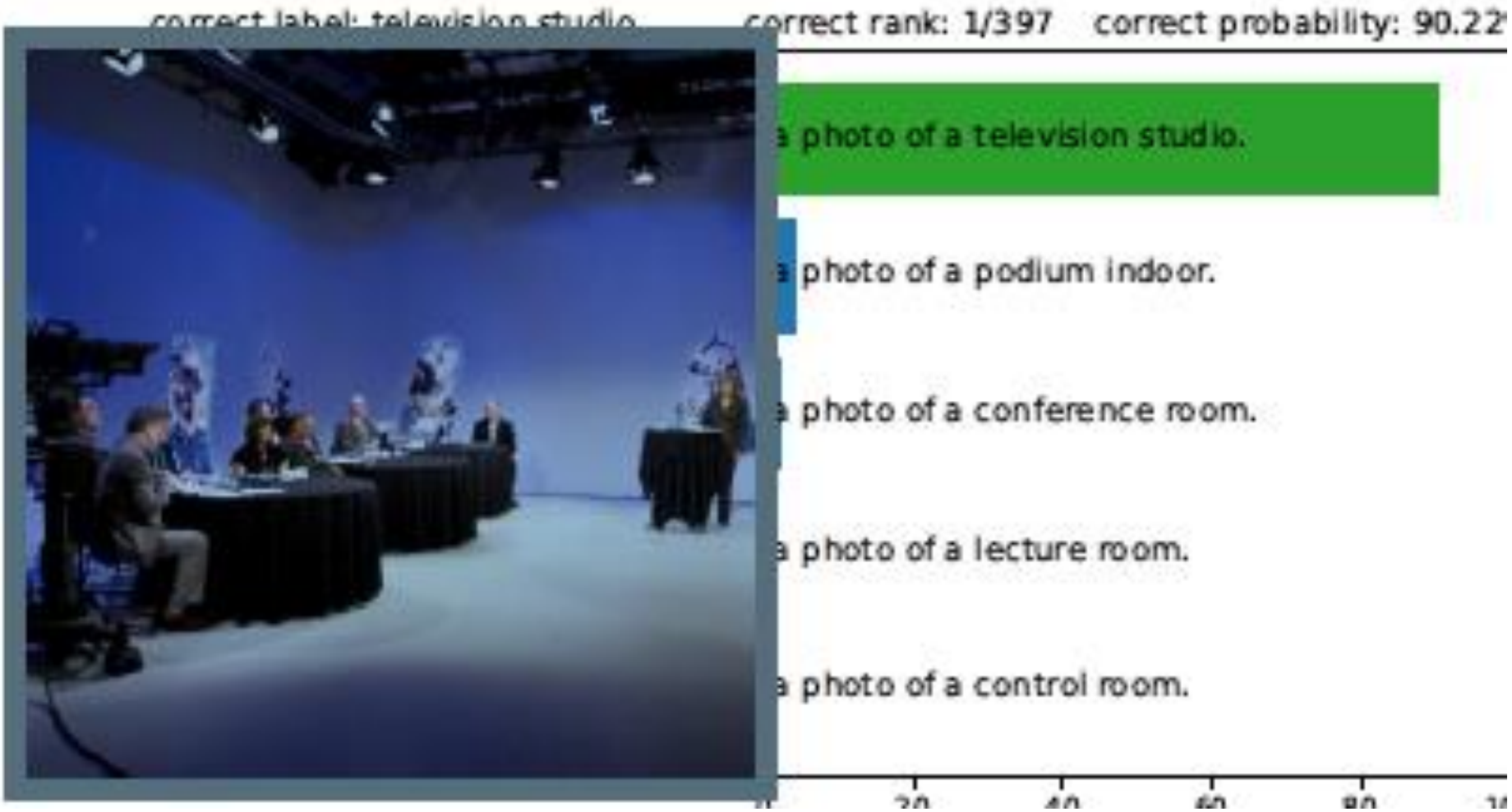


photo of a fox squirrel.
a photo of a lynx.

ImageNet-A (Adversarial)

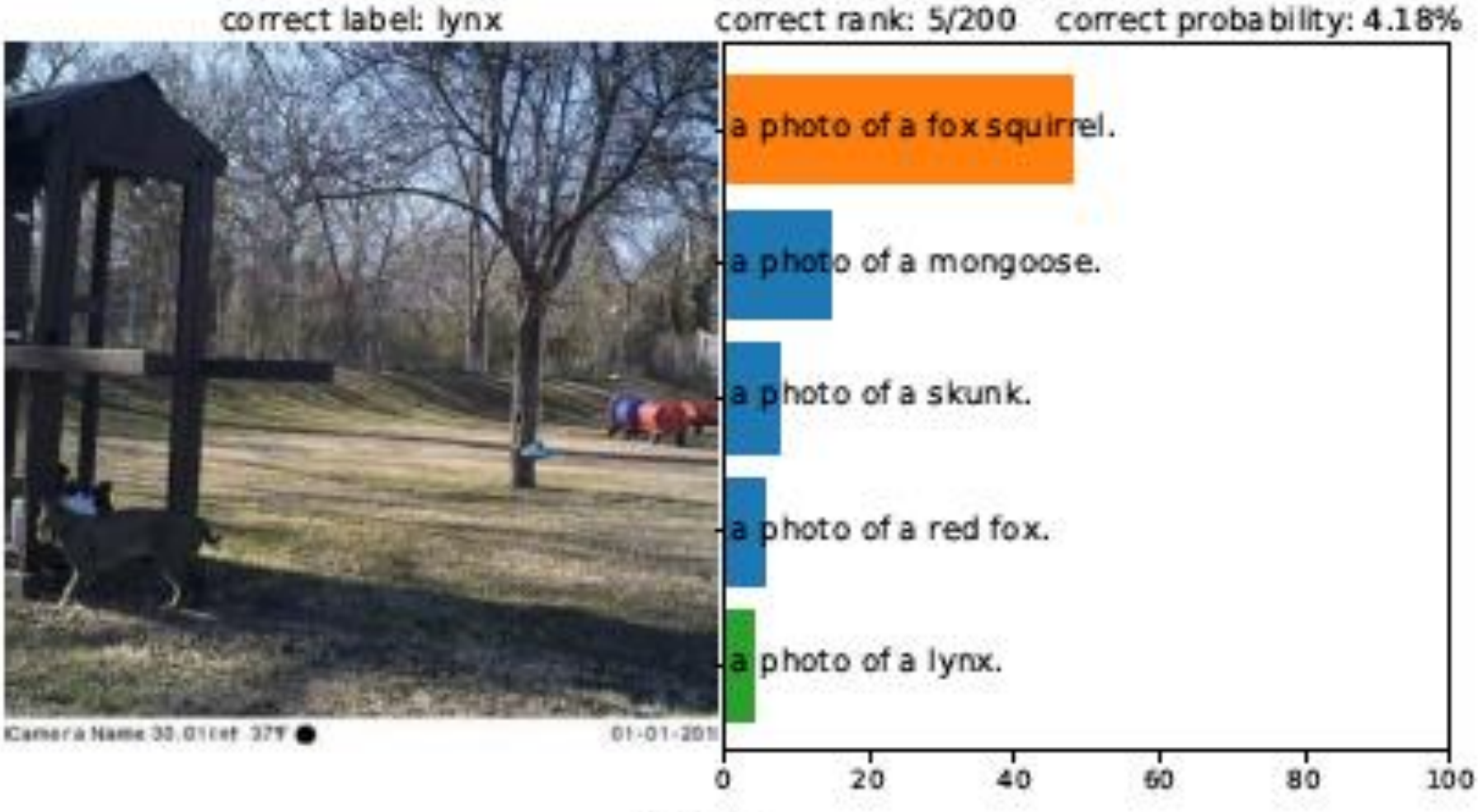
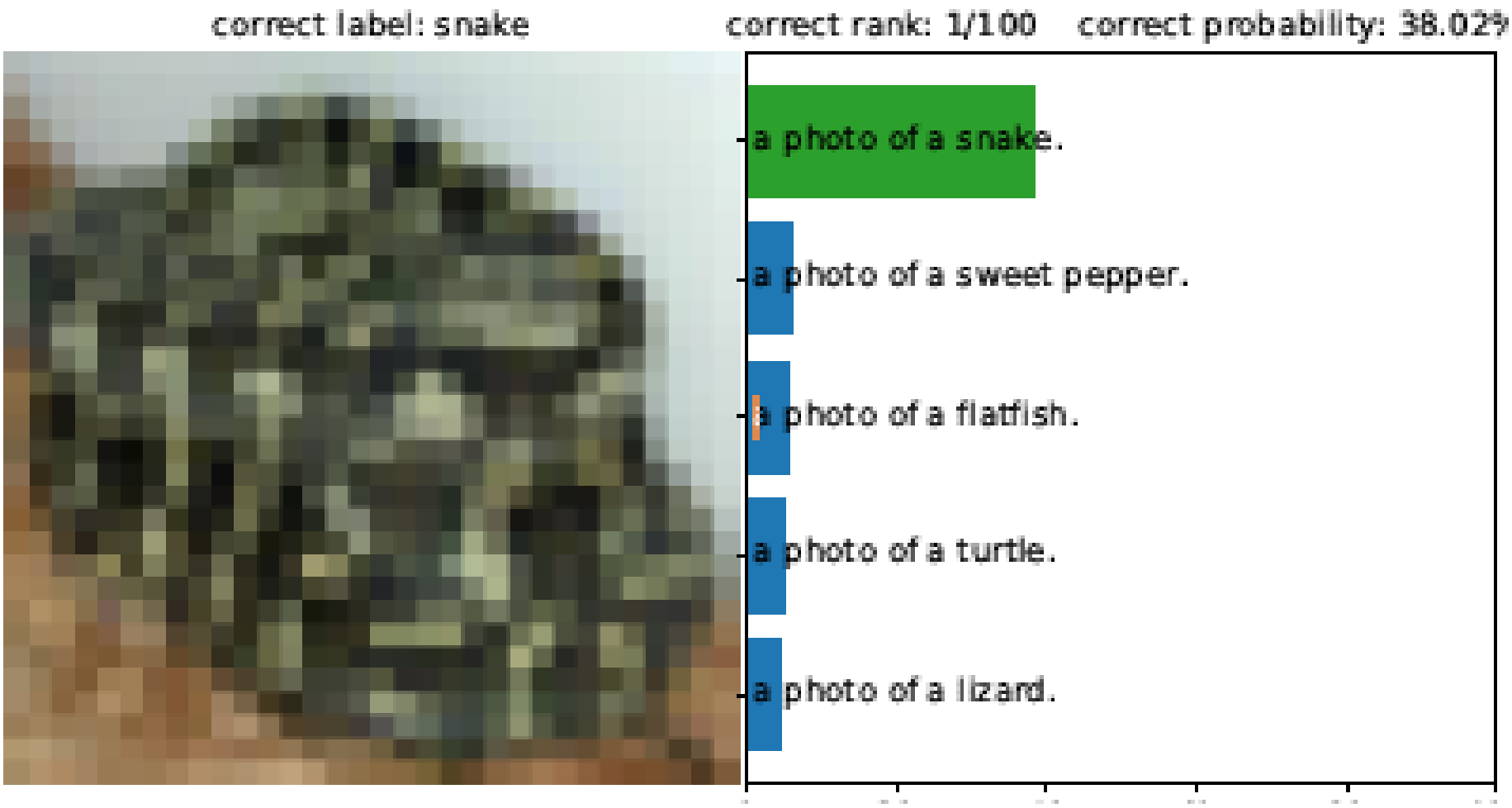


photo of a person volleyball spiking.

UCF101

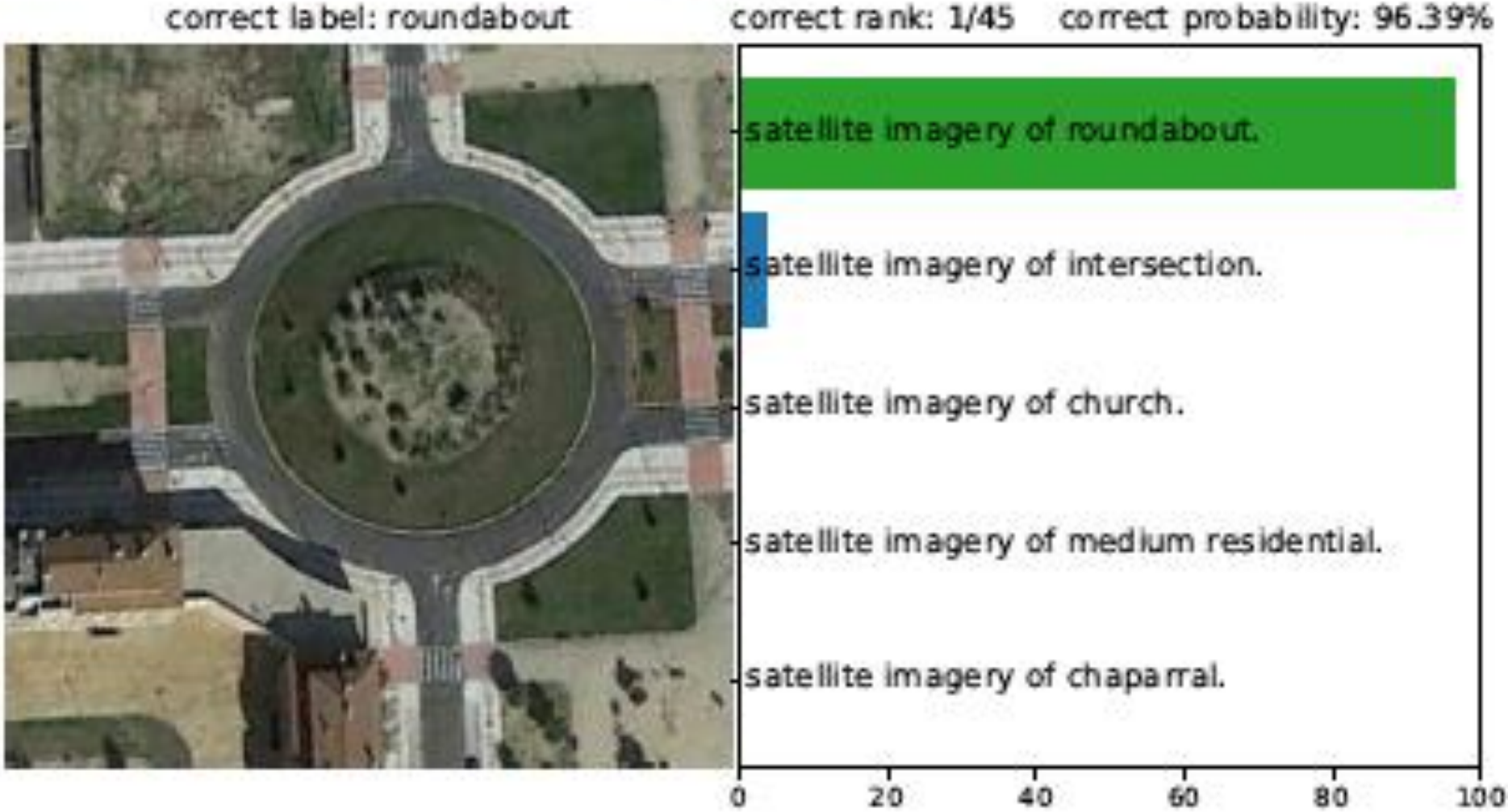
a photo of a snake.

CIFAR-100



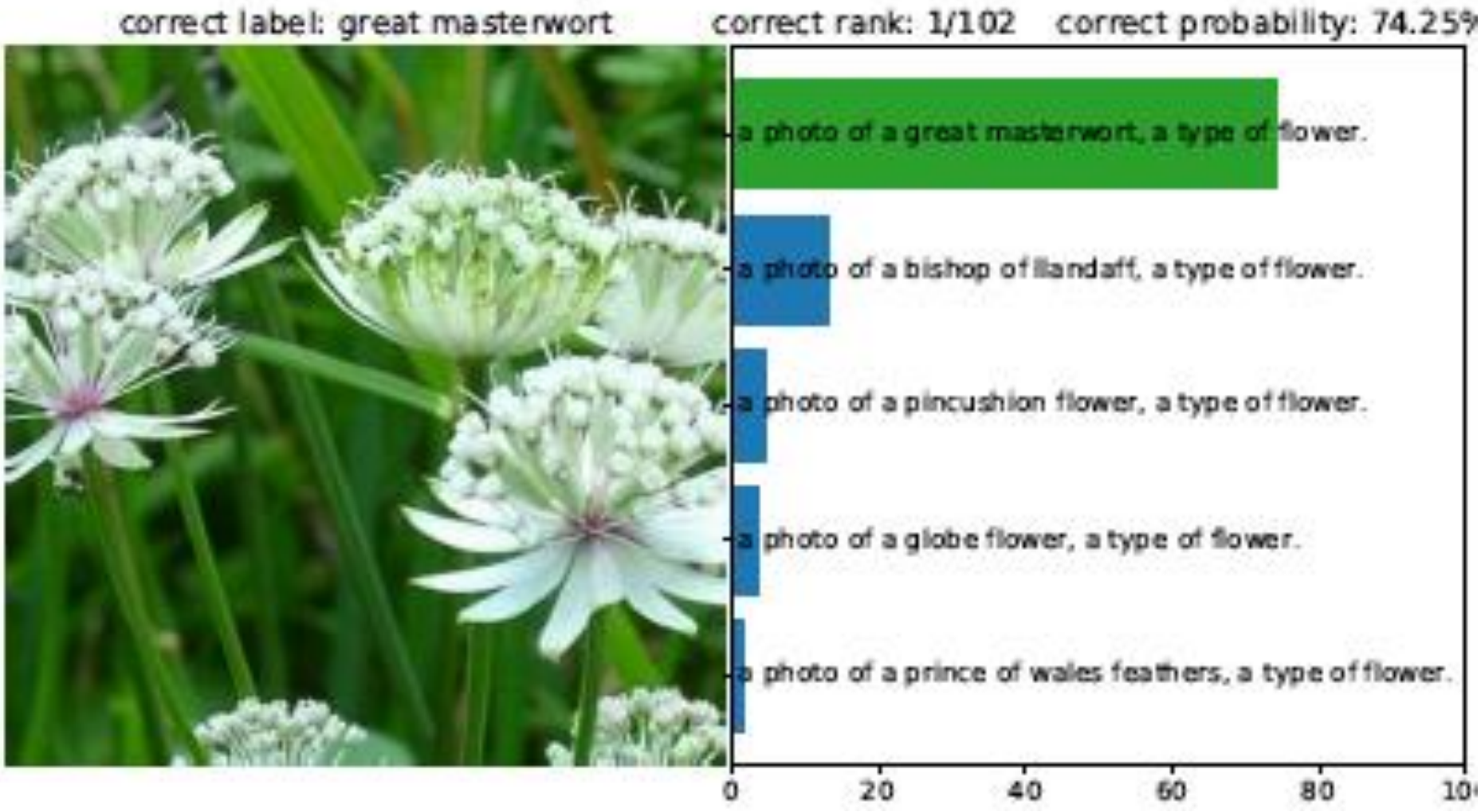
satellite imagery of roundabout.

RESISC45



a photo of a great masterwort, a type of flower.

Flowers-102

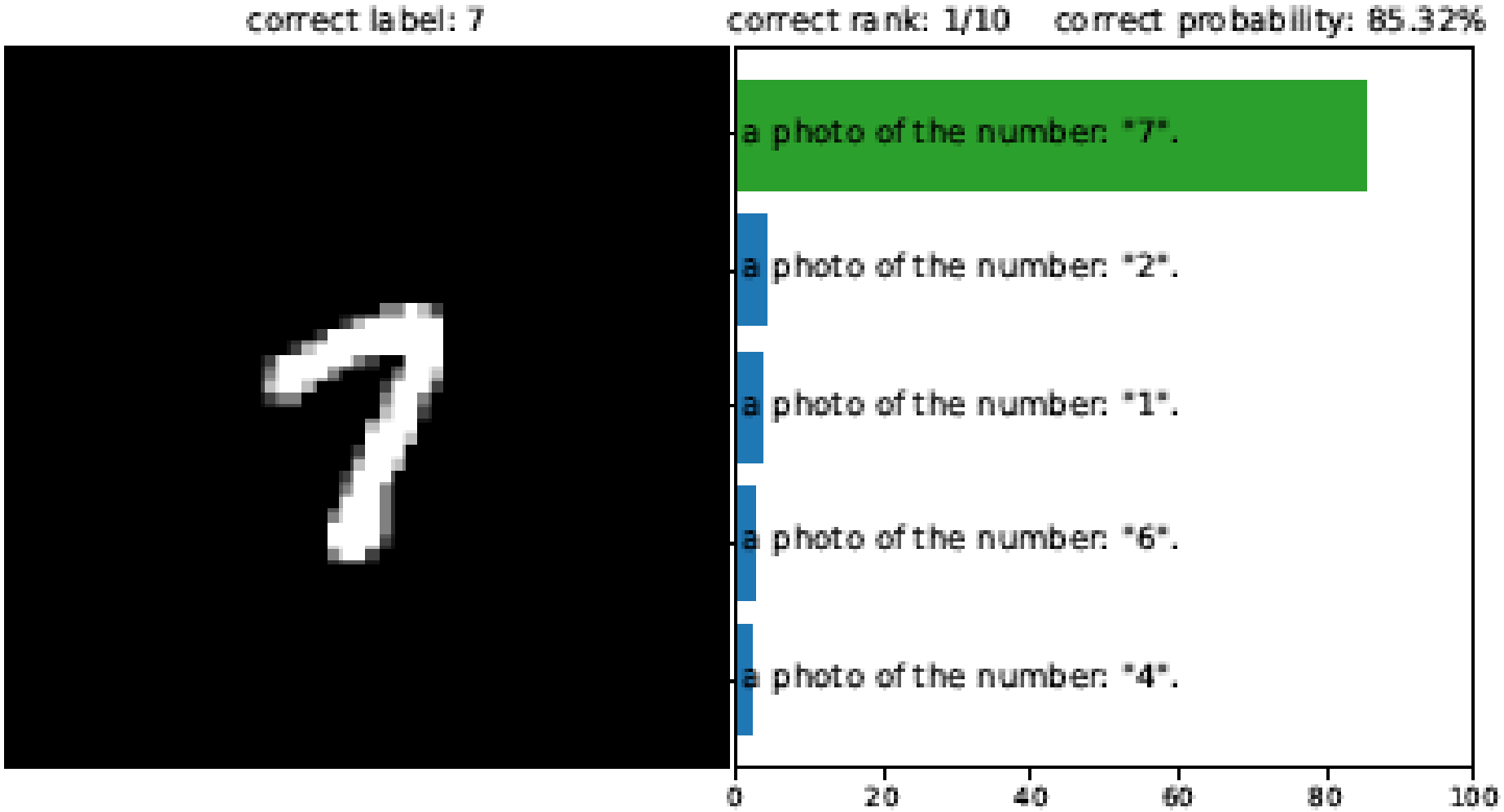


a photo of a pill bottle.

ObjectNet ImageNet Overlap

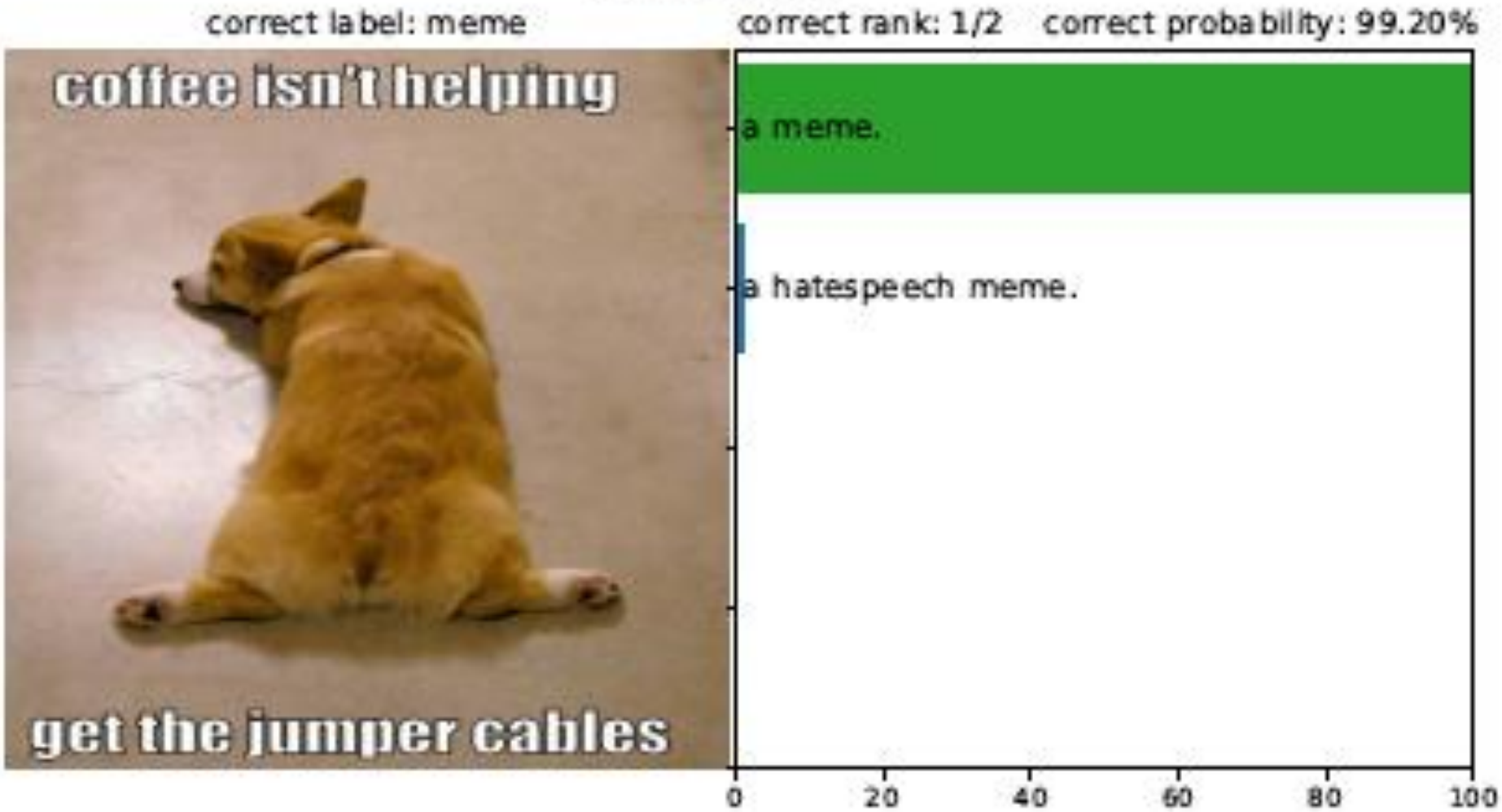
a photo of the number: "7".

MNIST



meme.

Hateful Memes



a photo of a airplane.

Youtube-BB

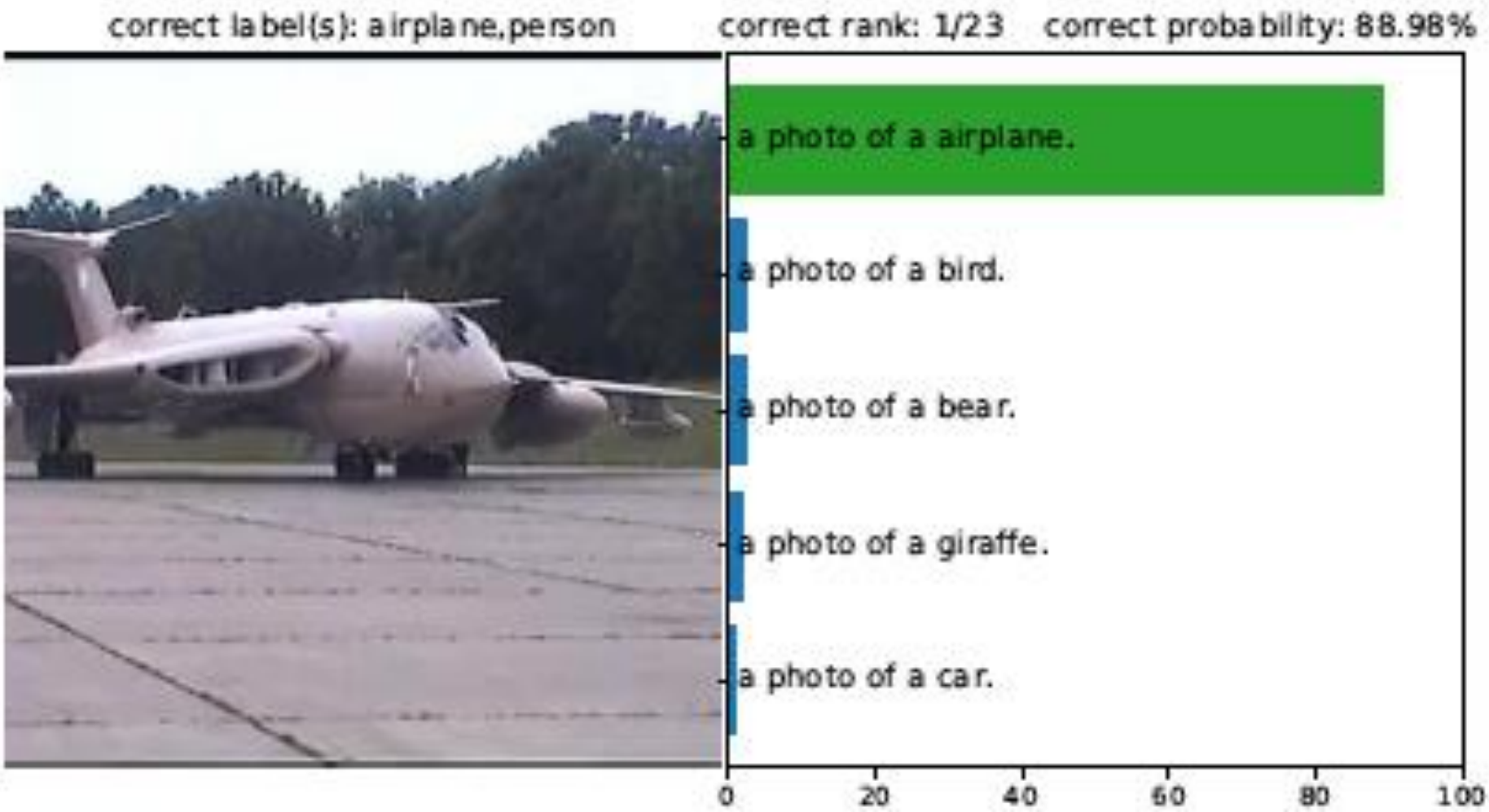
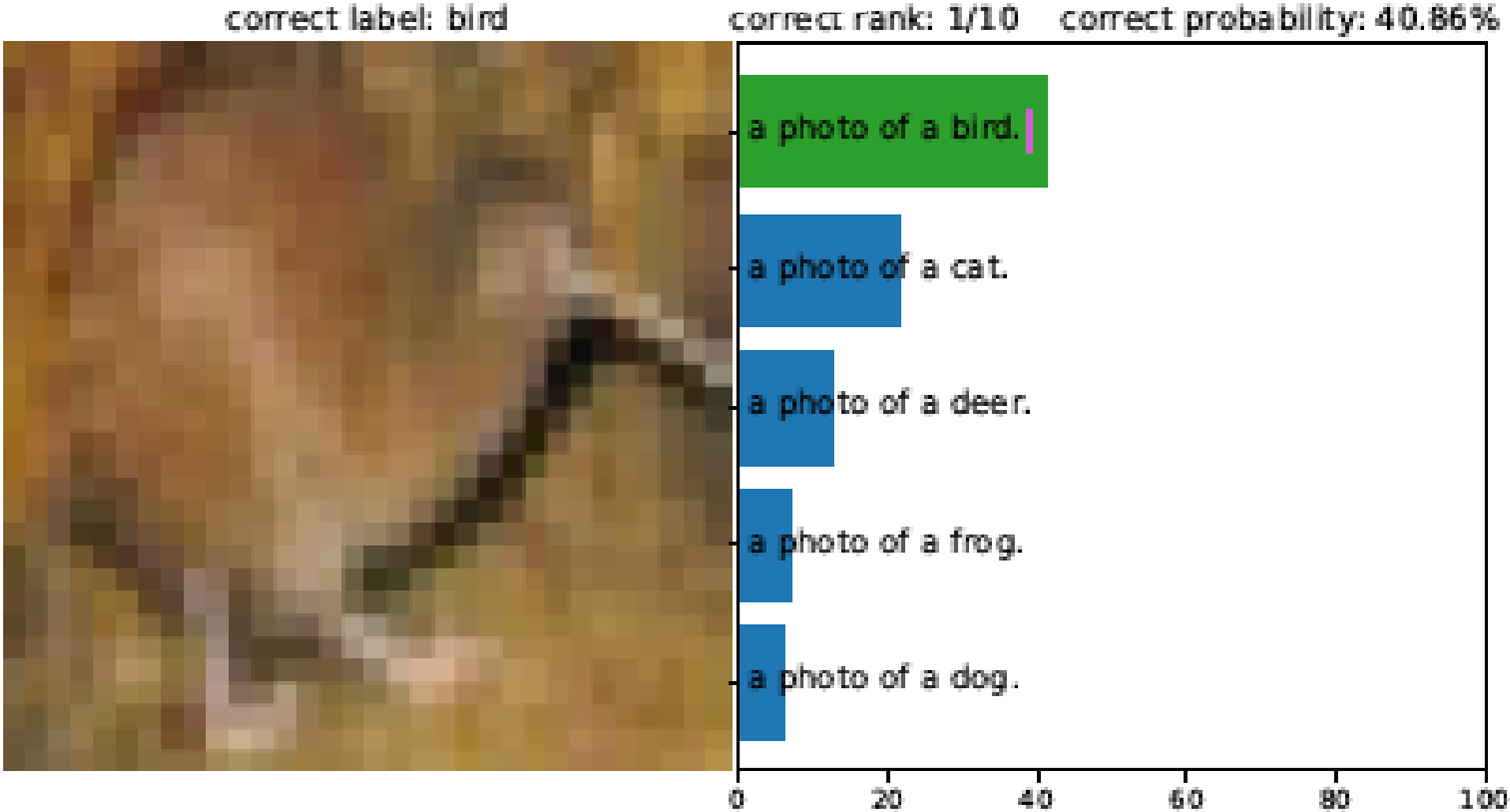


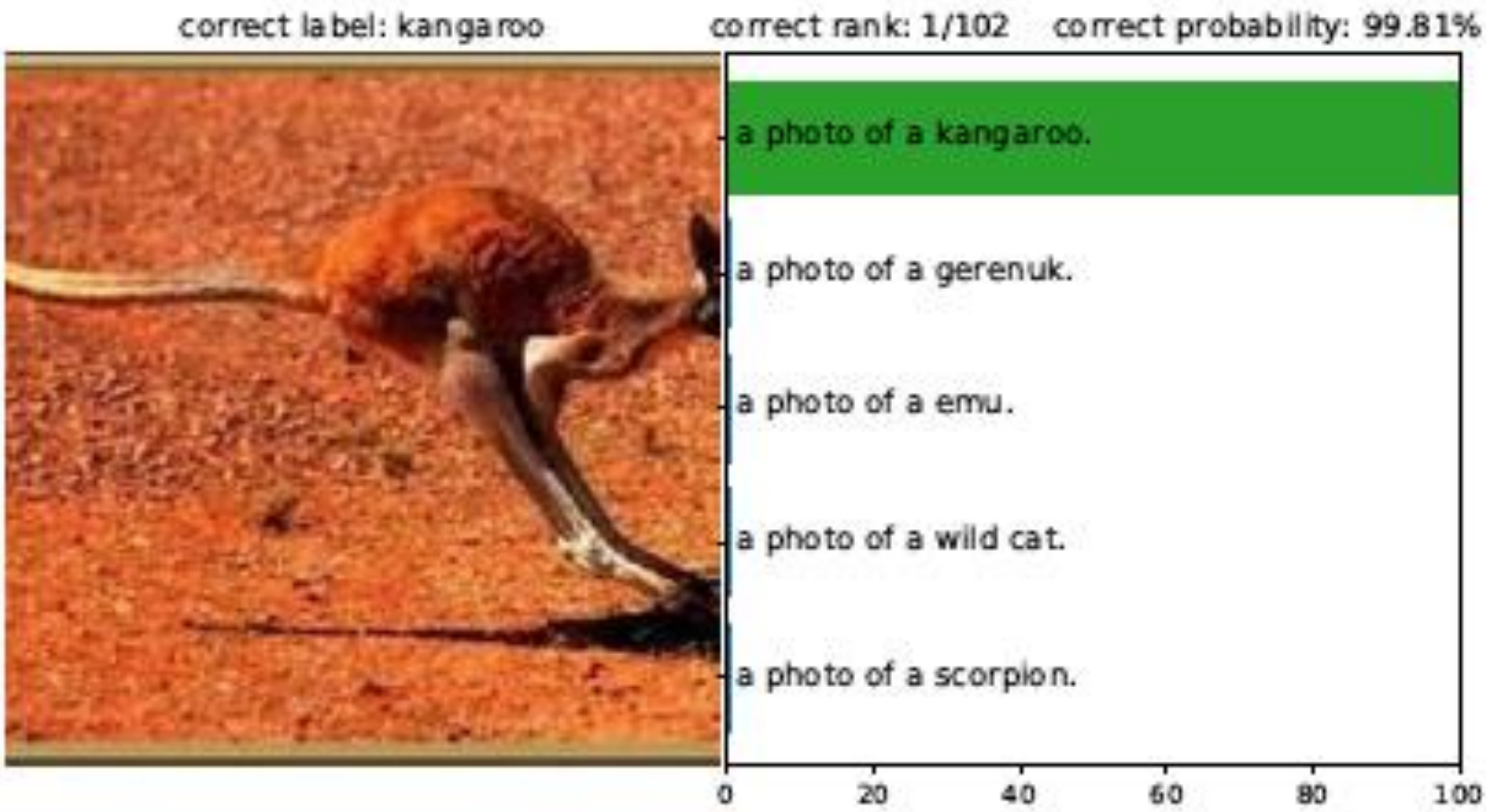
photo of a bird.

CIFAR-10



a photo of a kangaroo.

Caltech-101



a photo of a beer bottle.

ImageNetV2 Matched Frequency

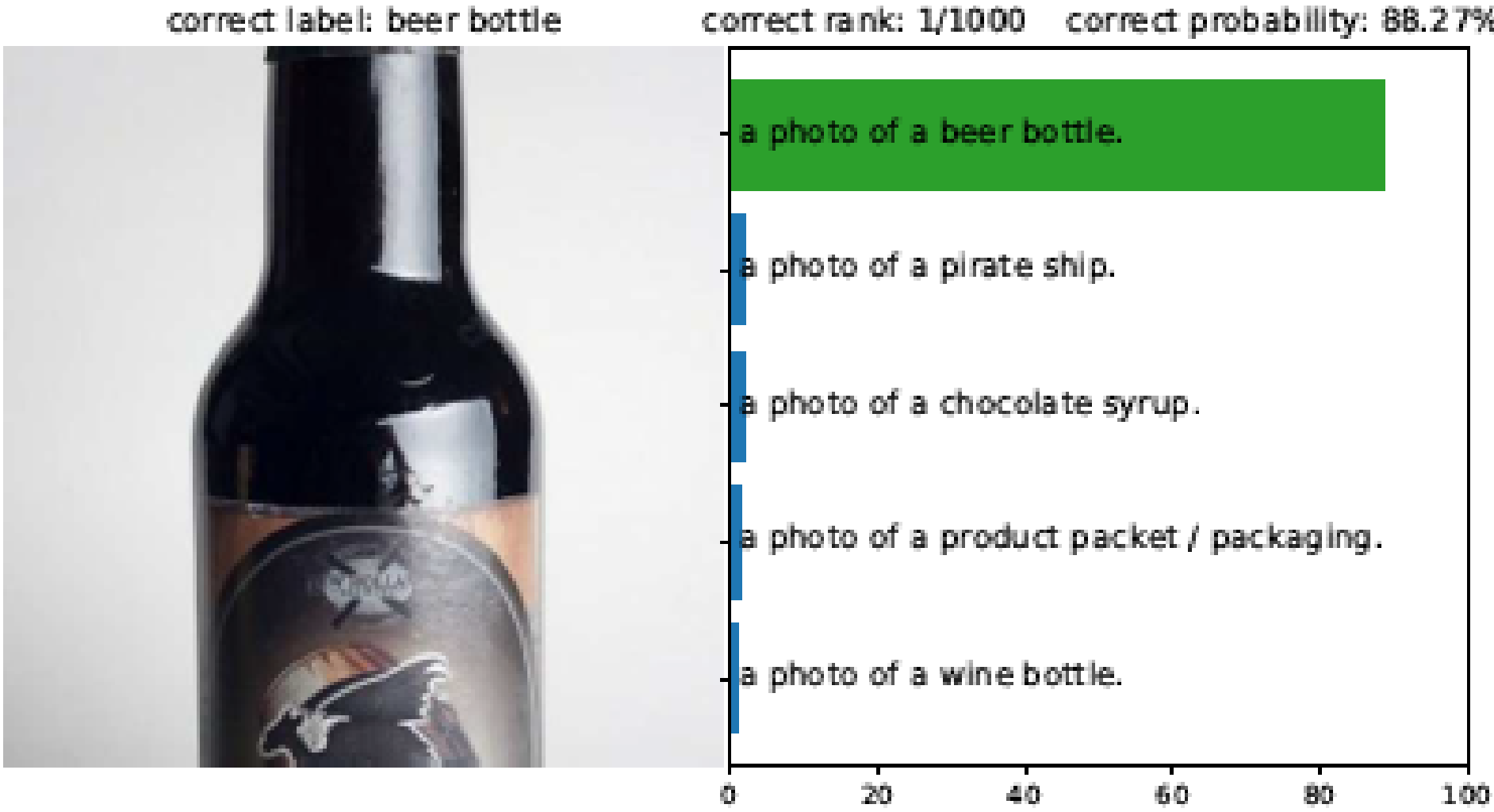
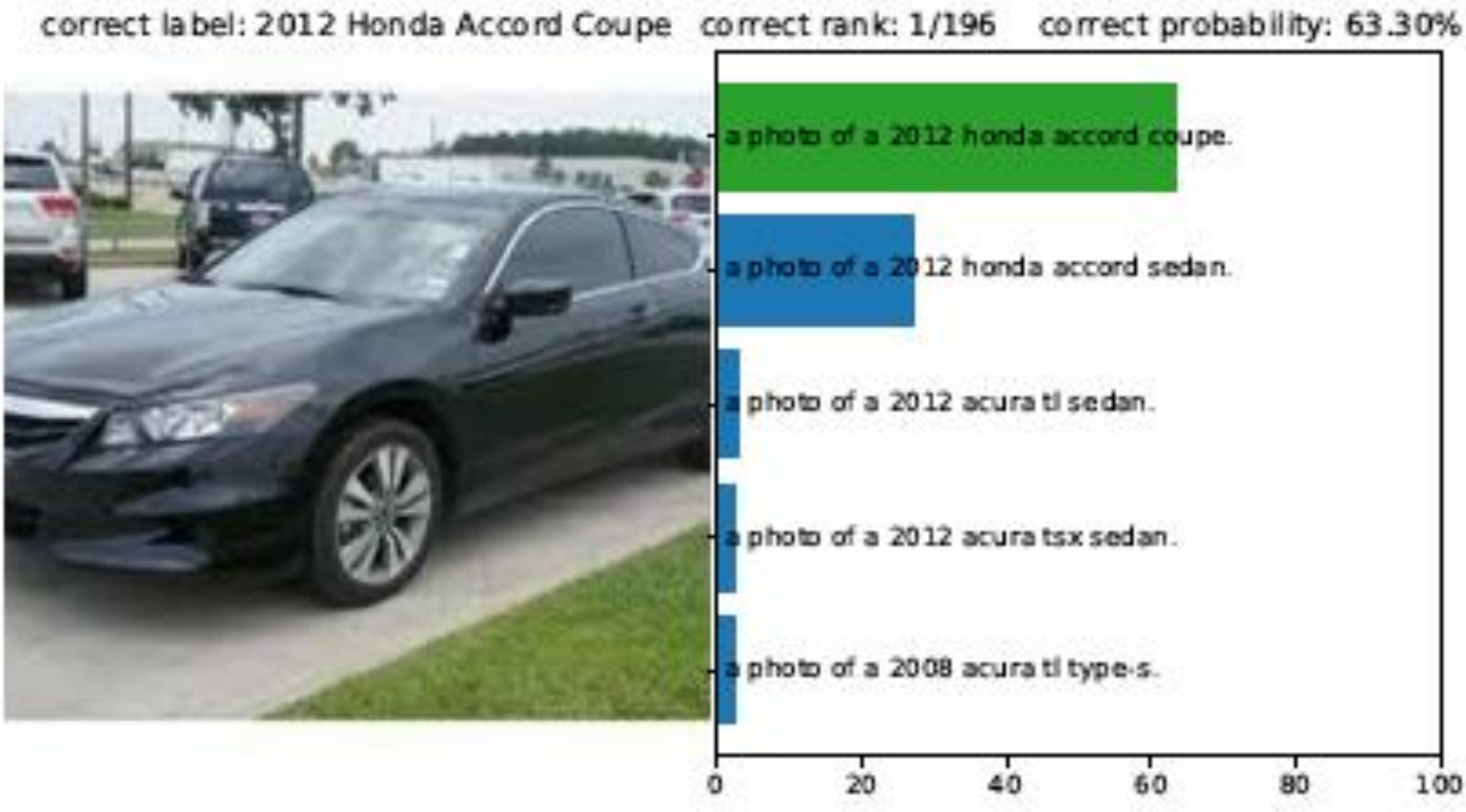


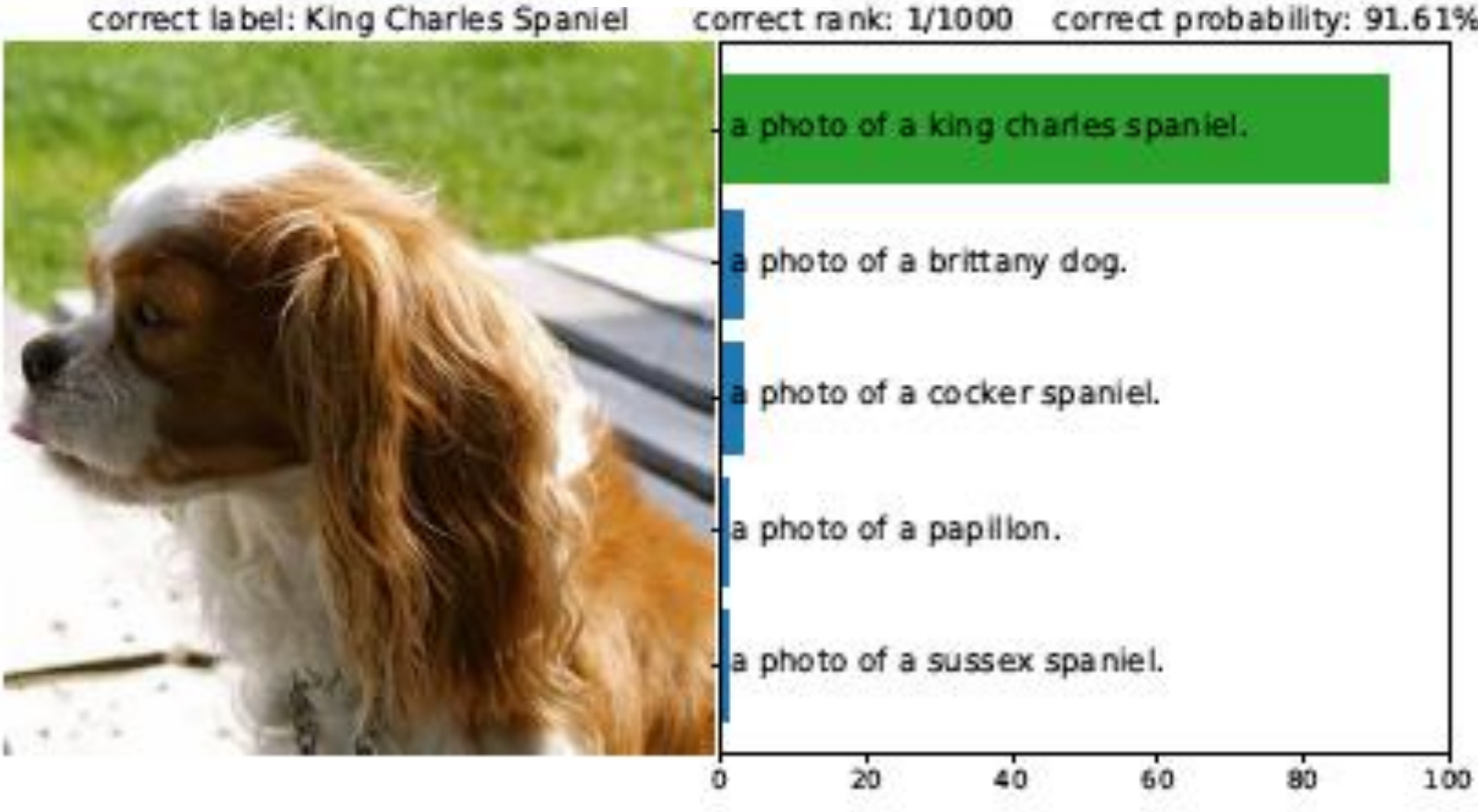
photo of a 2012 honda accord coupe.

Stanford Cars



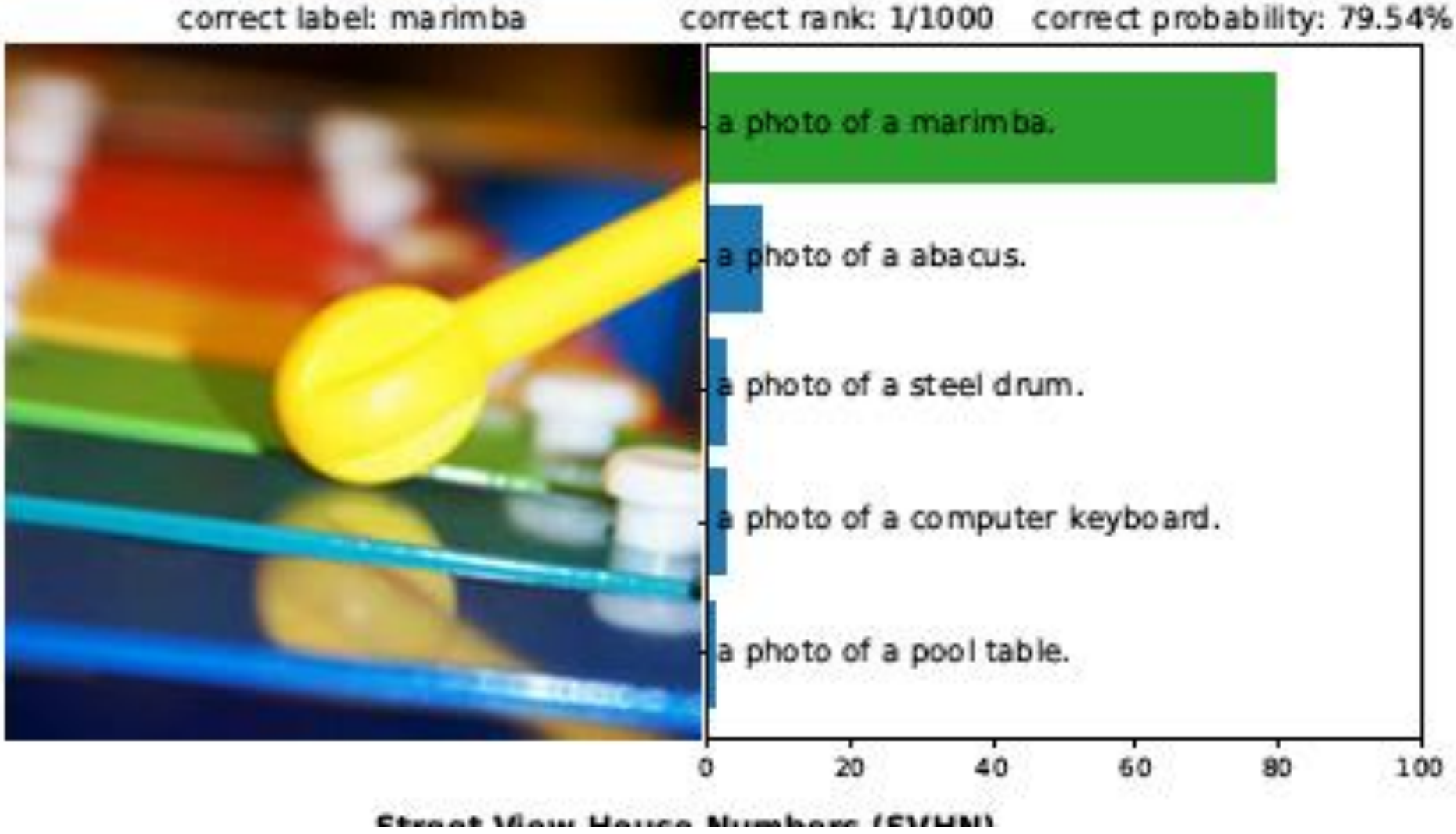
a photo of a king charles spaniel.

ImageNet



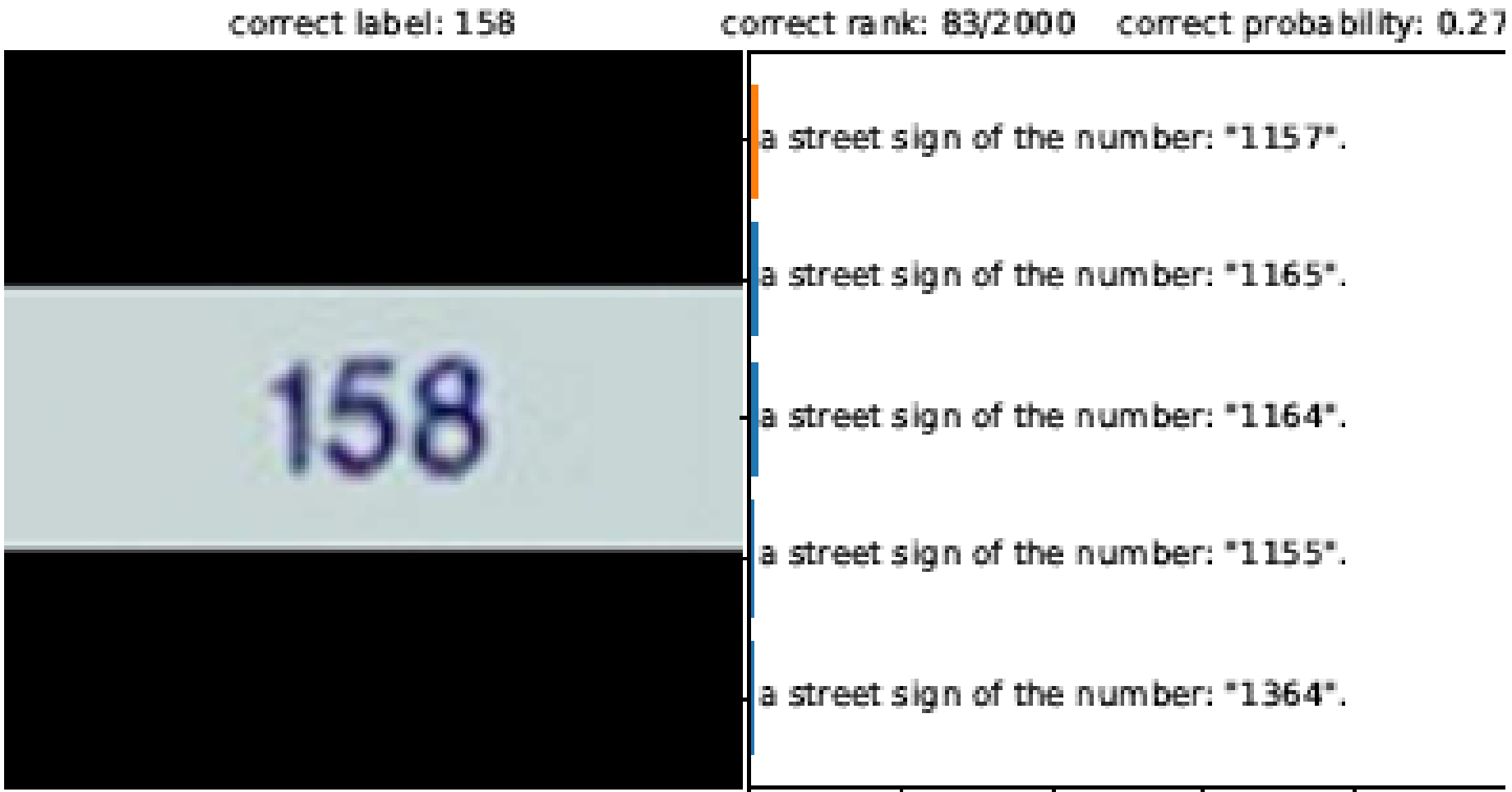
a photo of a marimba.

ImageNet Blurry



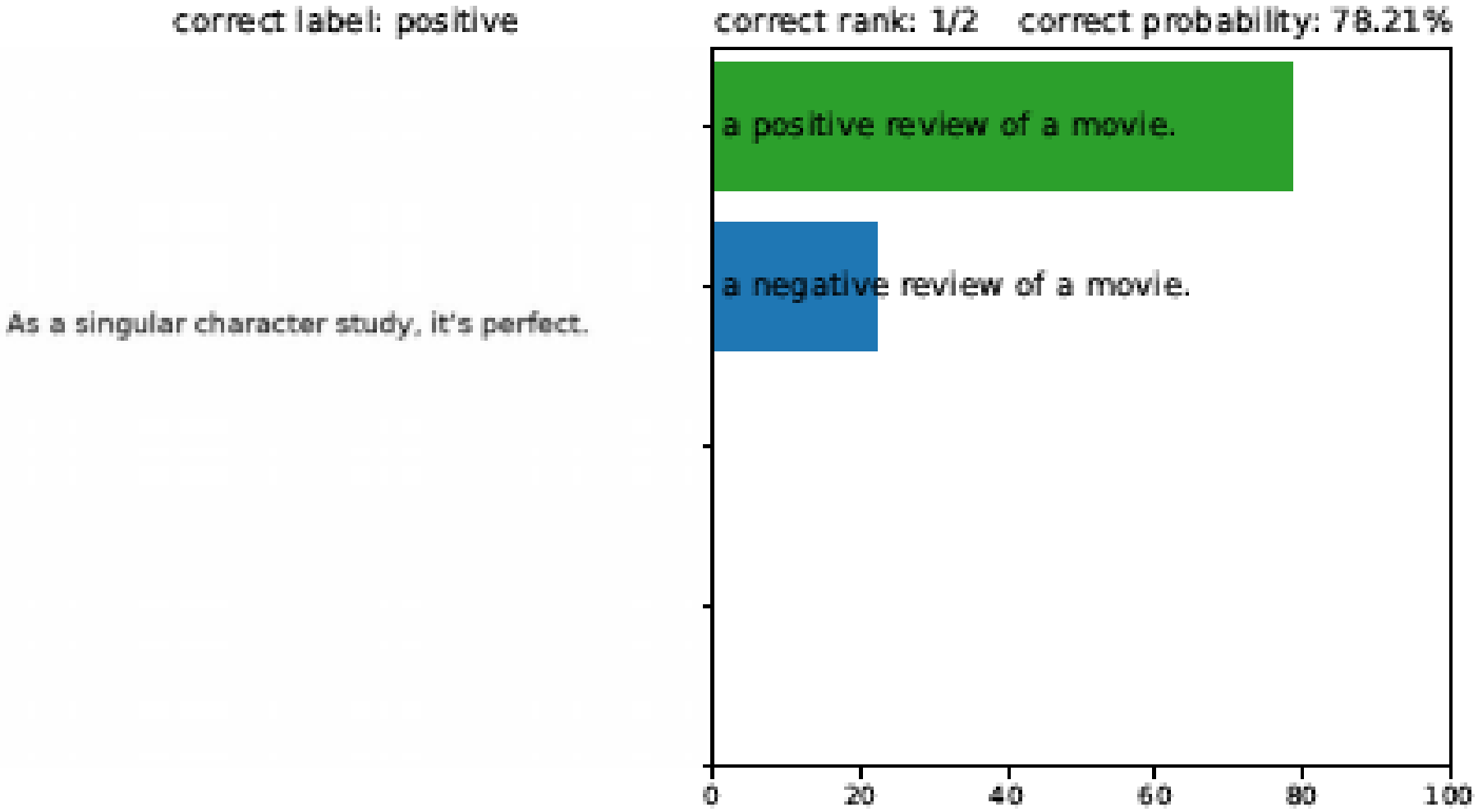
a street sign of the number “1157”

Street View House Numbers (SVHM)



a positive review of a movie.

Stanford Sentiment Treebank



centered satellite photo of permanent crop land.
centered satellite photo of annual crop land

EuroSAT

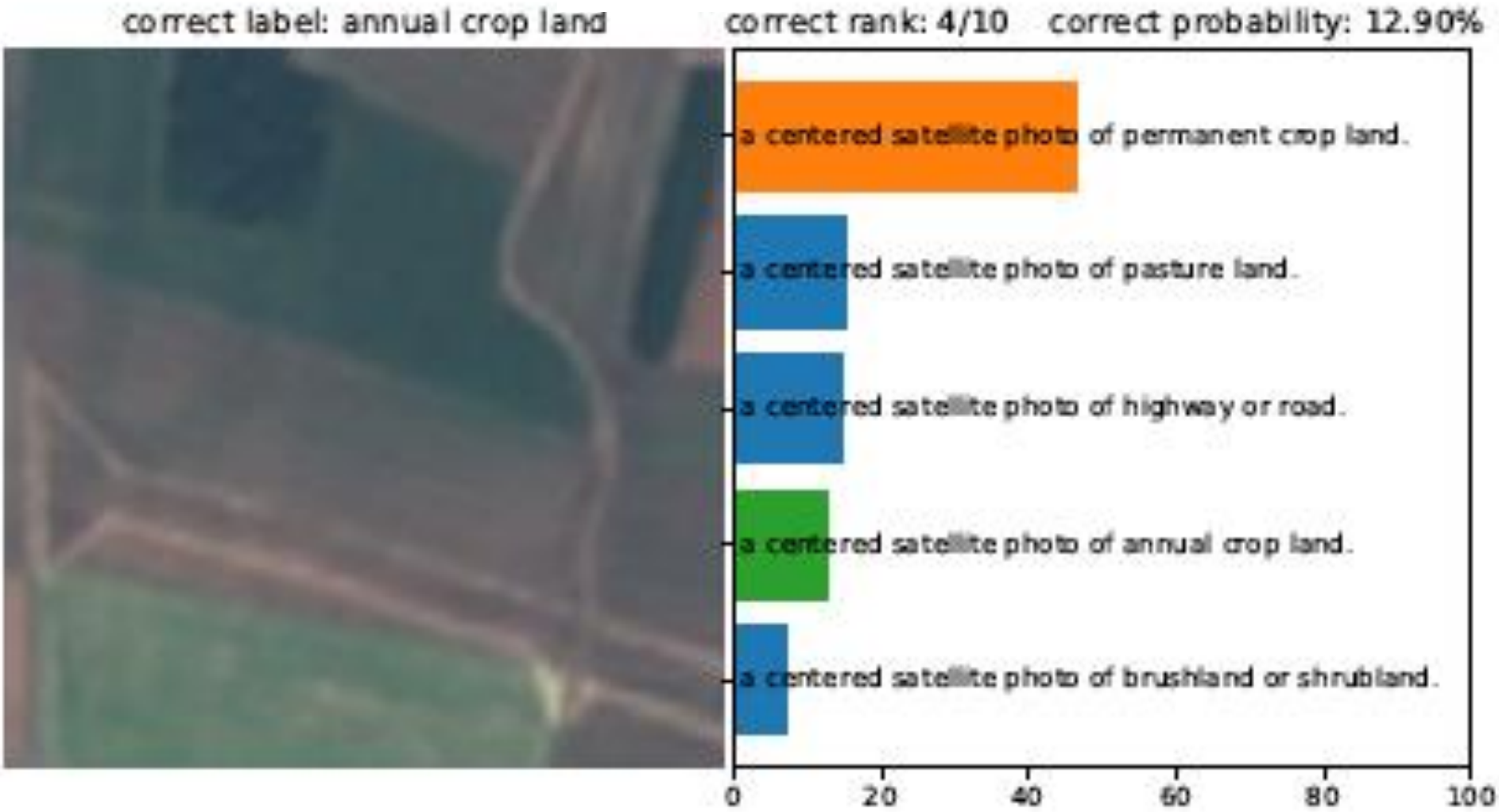


photo of 3 objects

photo of 4 objects

CLEVR Count

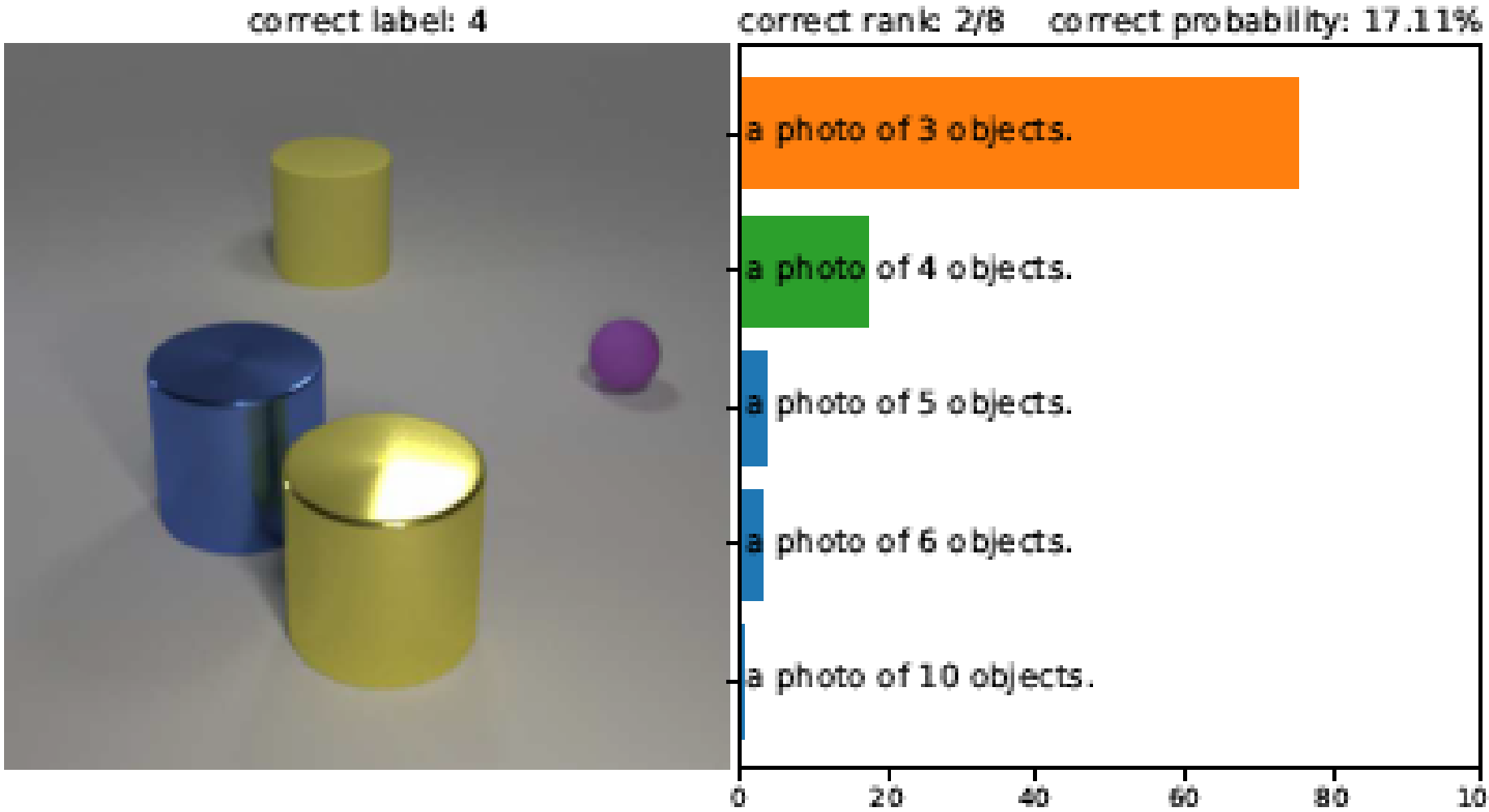
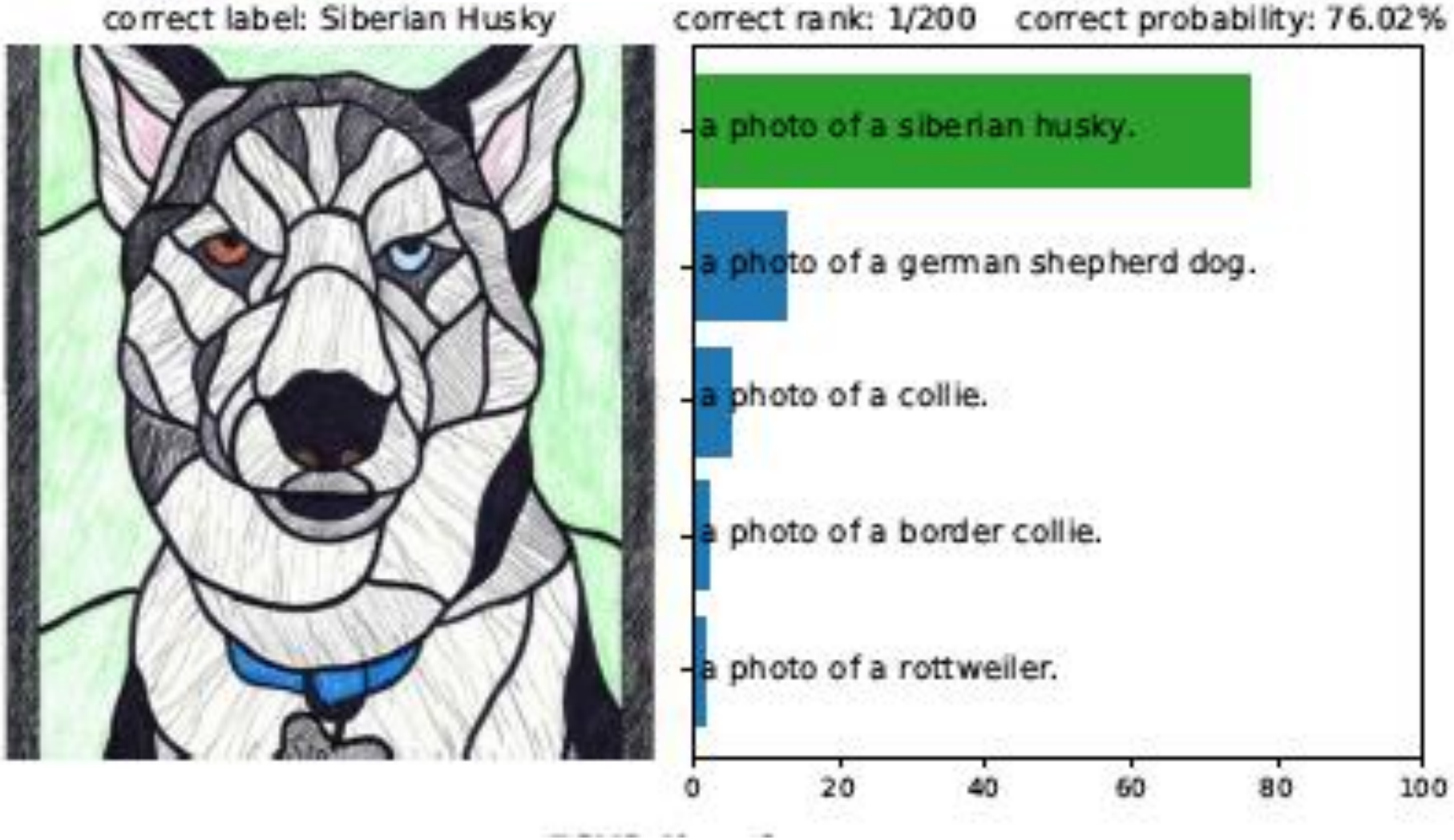


photo of a siberian husky.

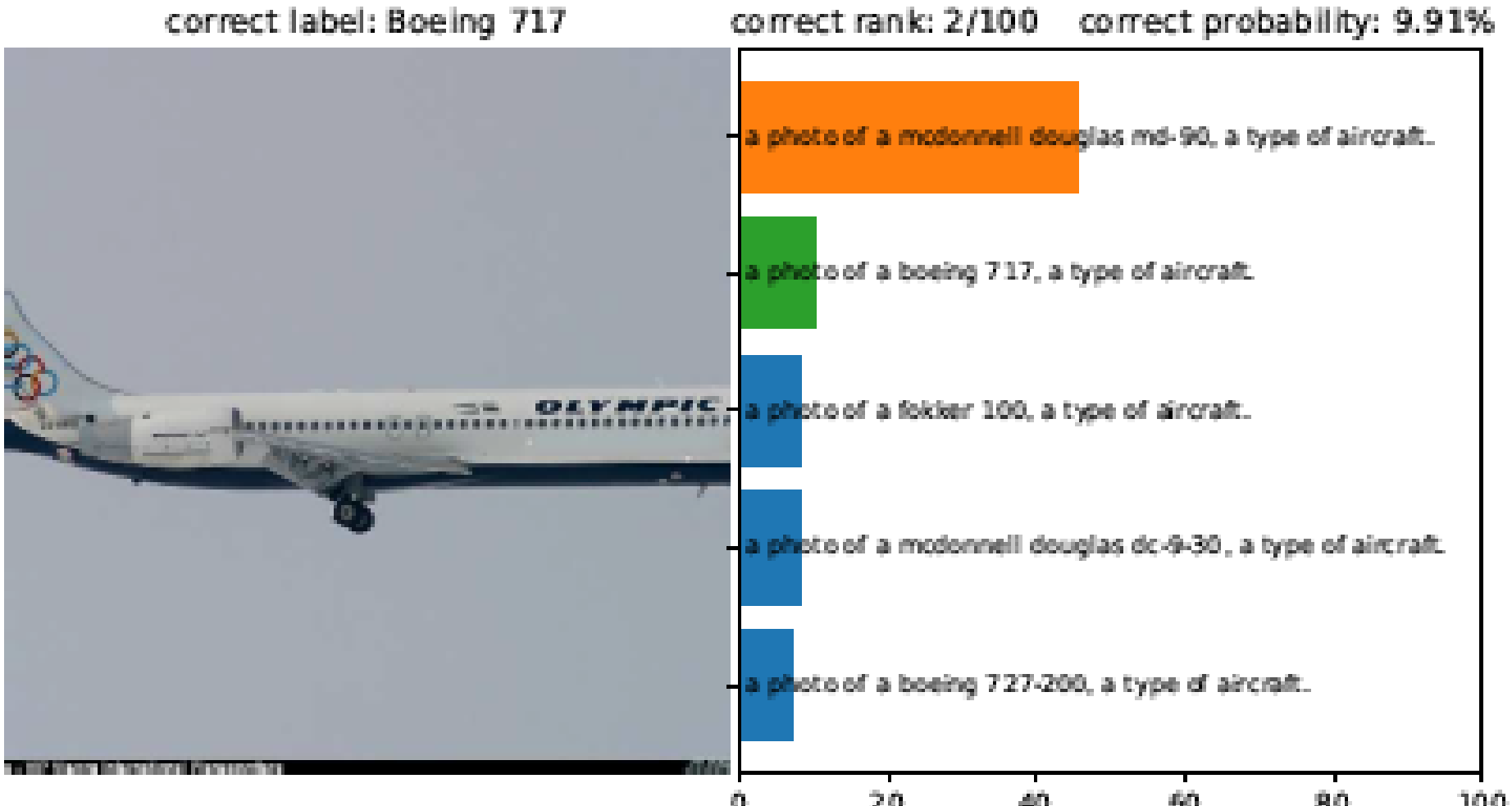
ImageNet-R (Rendition)



a photo of a mcdonnell douglas md-90, a type of aircraft

photo of a boeing 717, a type of aircraft

FGVC Aircraft



a photo of a kennel indoor.

SUN

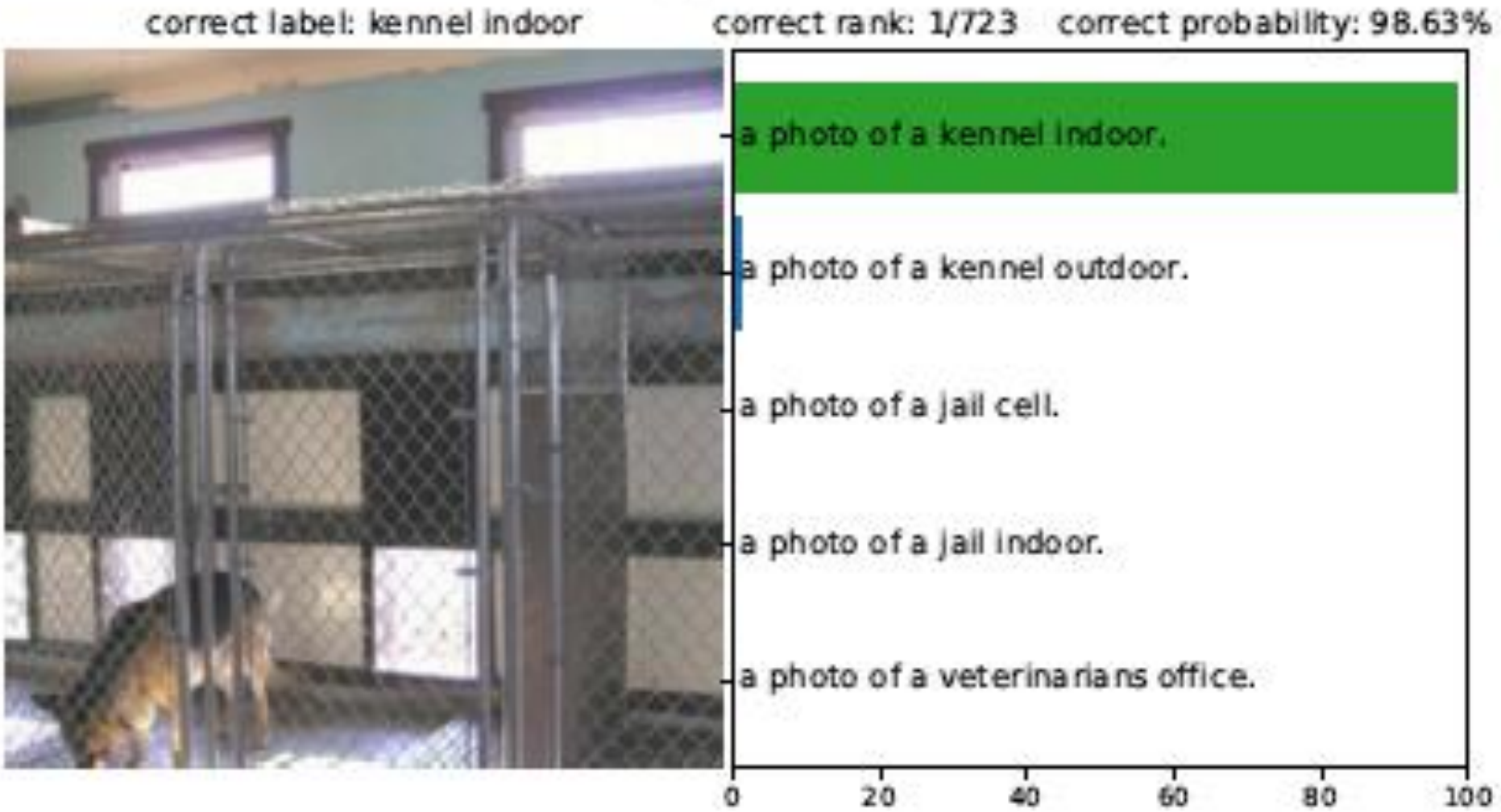


photo of a broad tailed hummingbird, a type of bird.

photo of a black chinned hummingbird, a type of bird.

Birdsnap

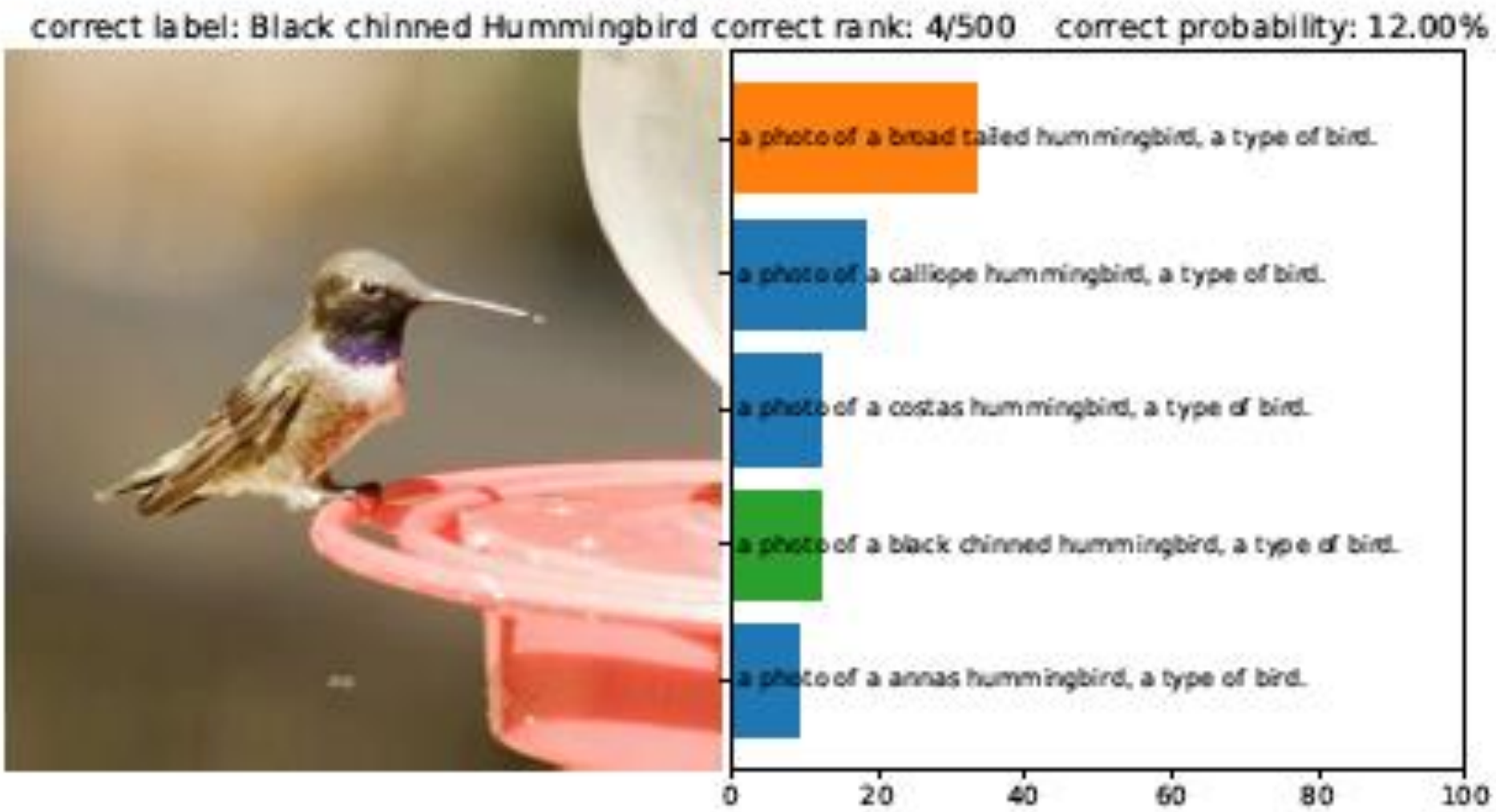


photo of a polka-dotted texture

photo of a perforated texture.

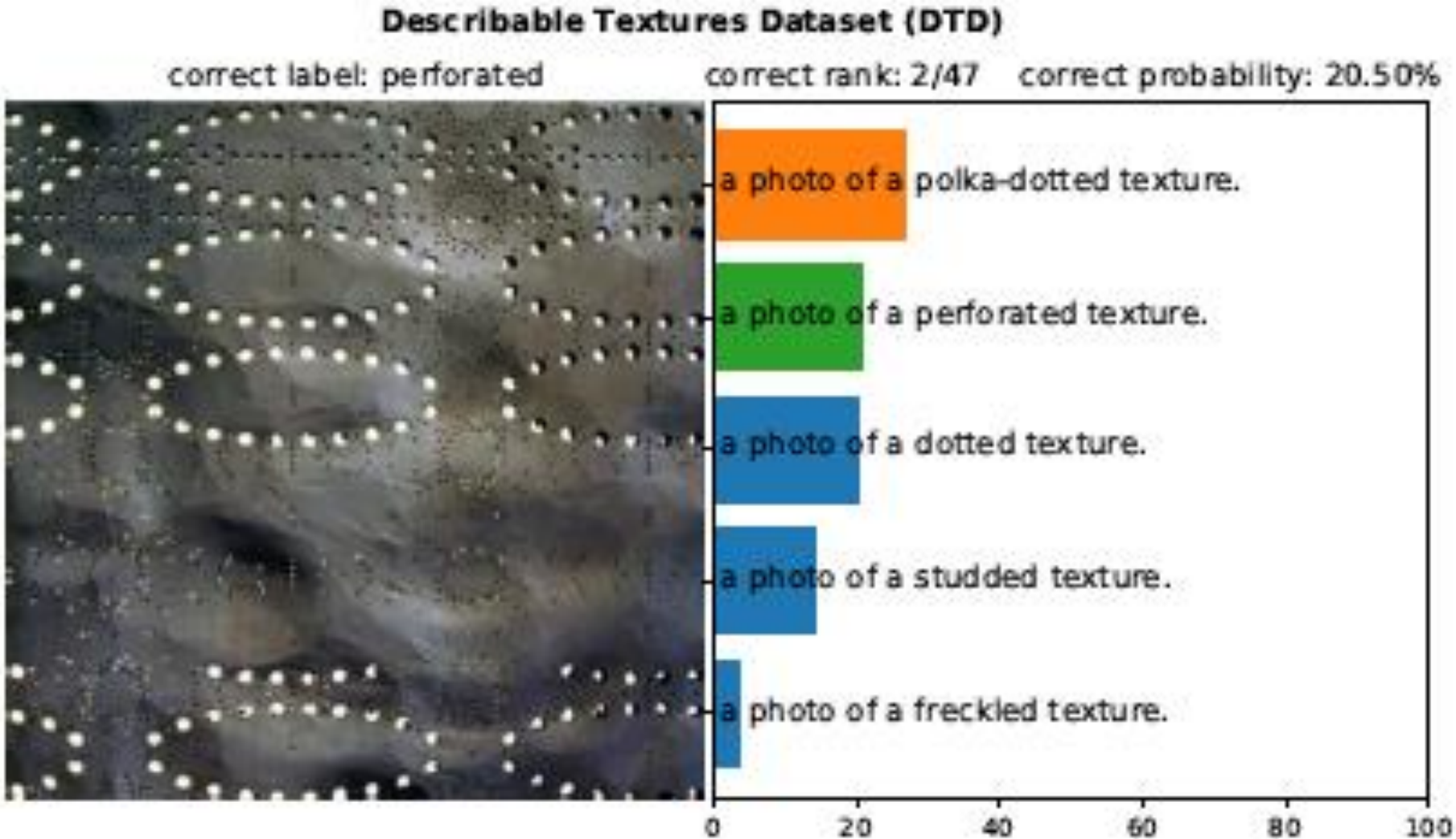
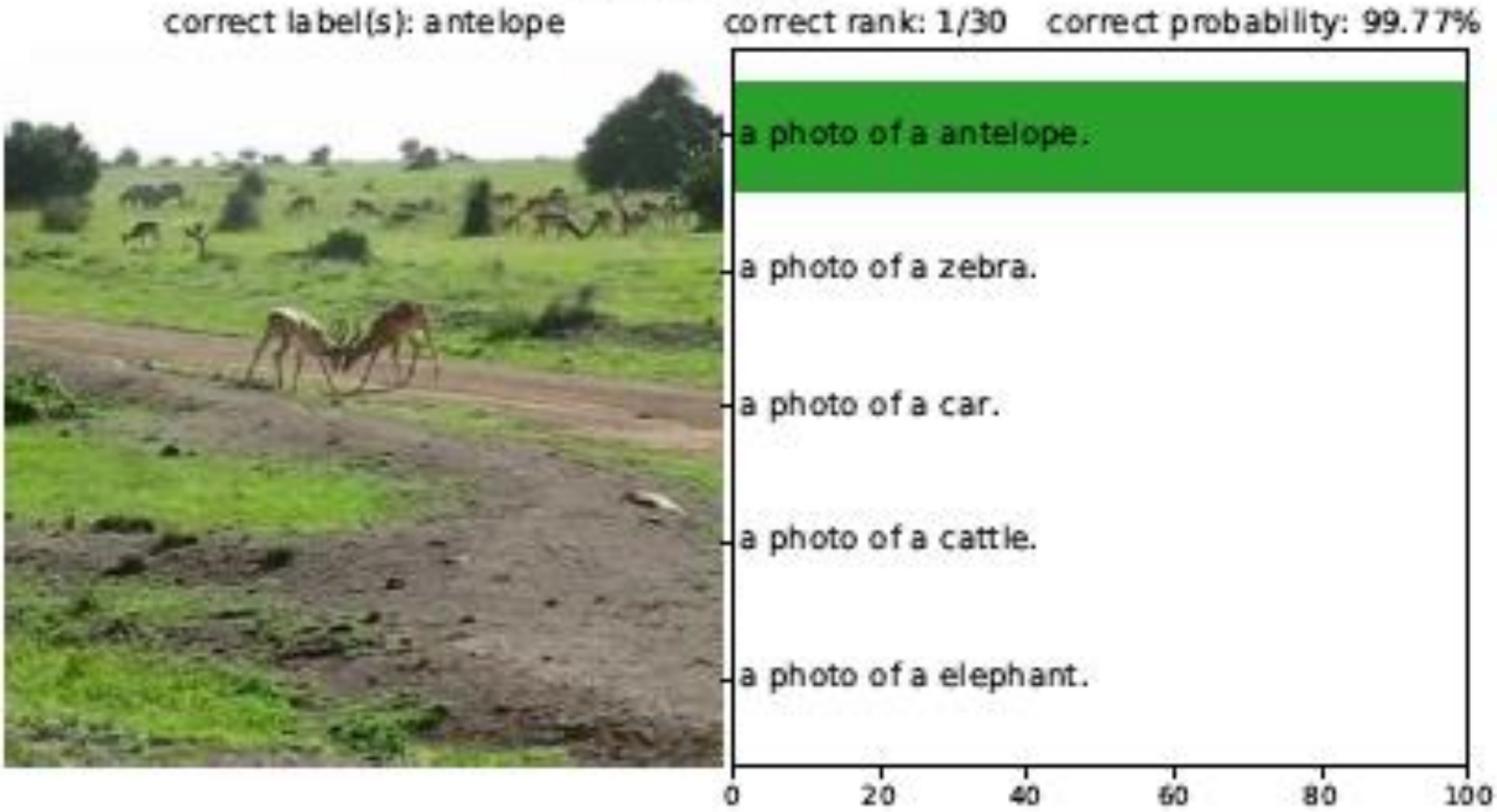


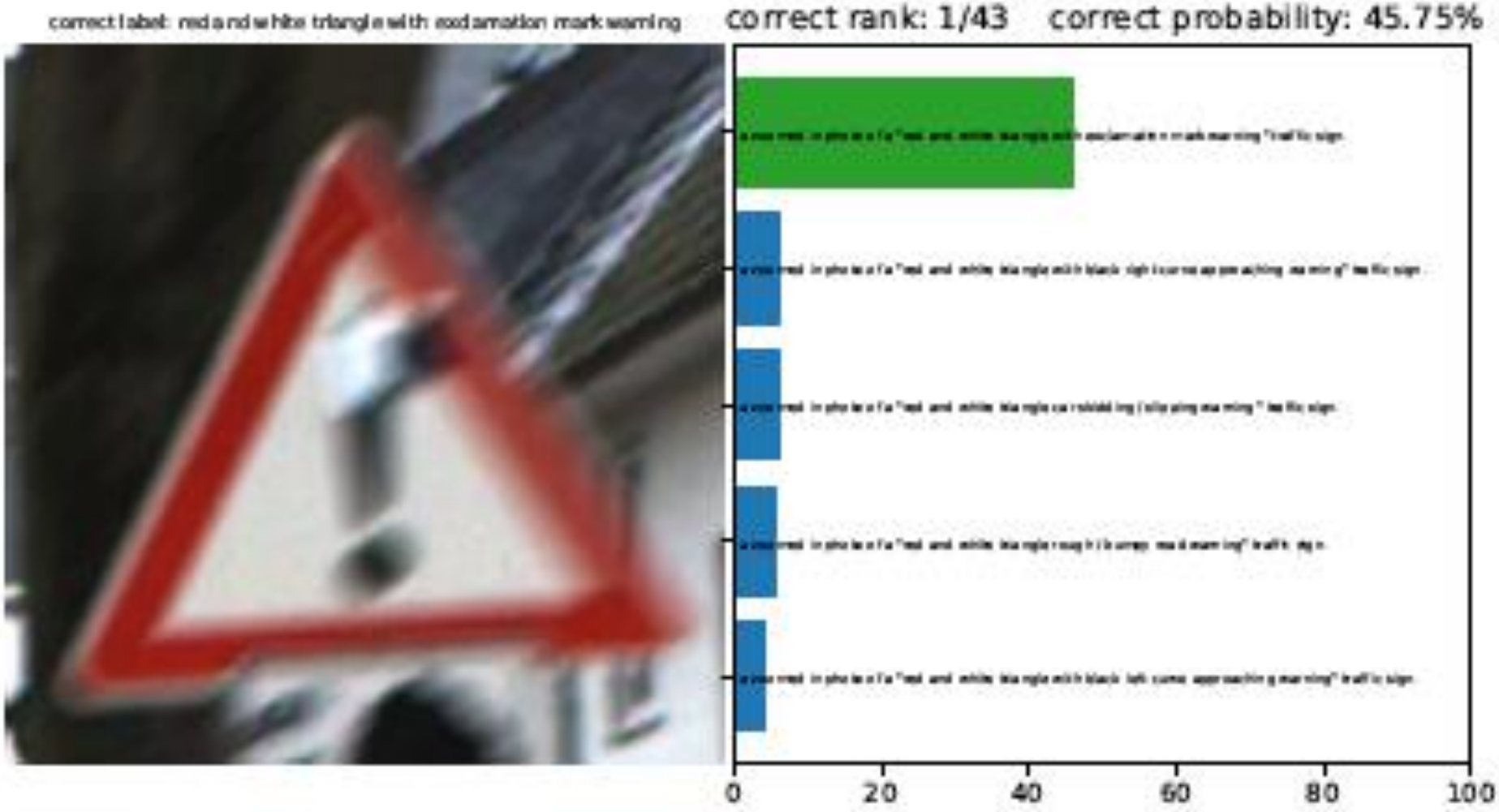
photo of a antelope.

ImageNet Vid



a zoomed in photo of a "red and white triangle with exclamation mark warning" traffic sign.

German Traffic Sign Recognition Benchmark (GTSRB)



CLIPによる予測の評価

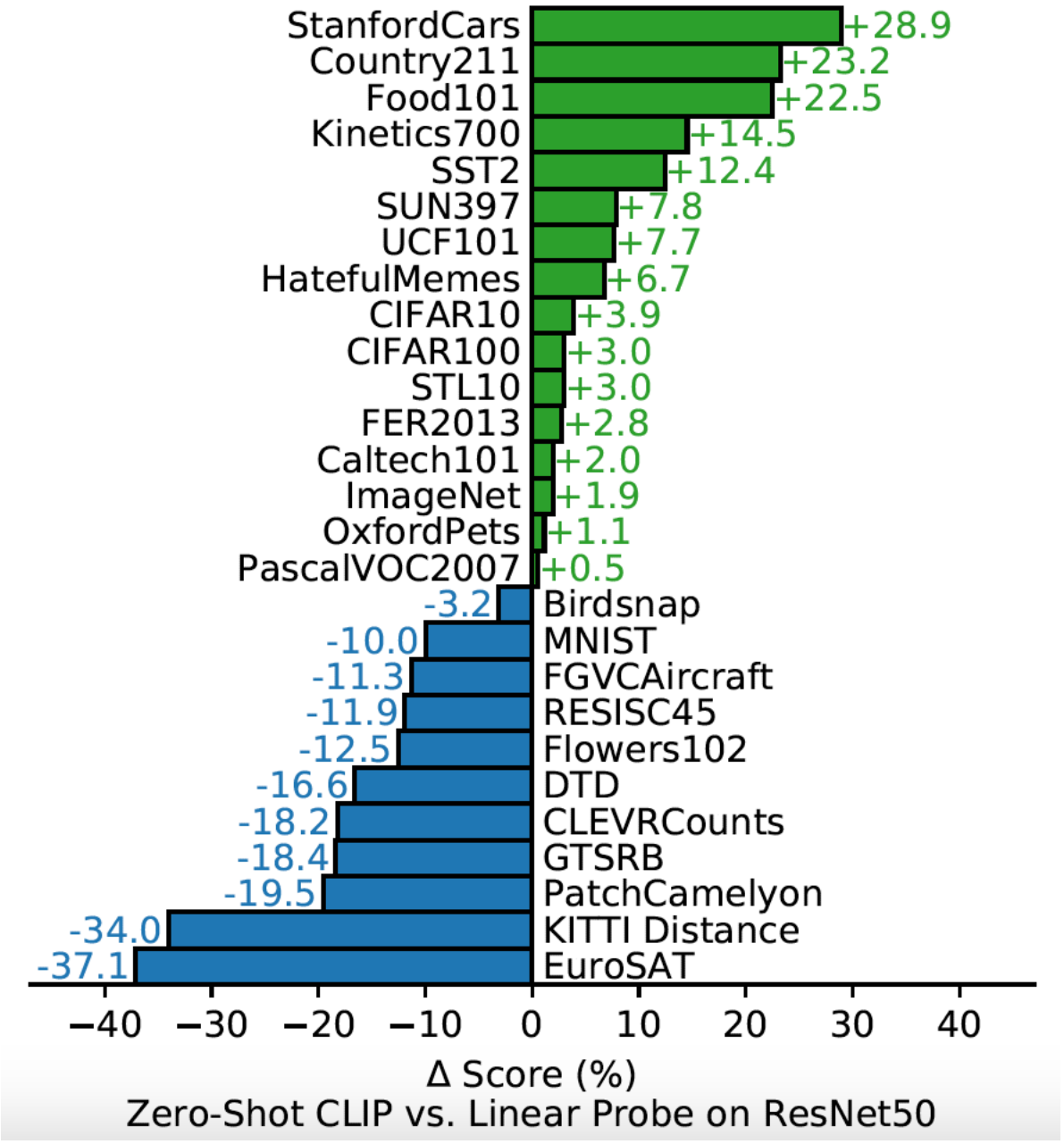


図5. ゼロショットCLIPは完全教師ありベースラインと競合する。

27のデータセットで評価した結果，ゼロショットCLIP分類器は，ImageNetを含む16のデータセットにおいて，ResNet-50特徴量に適合させた完全教師あり線形分類器を上回った.

ゼロショットのCLIPは、27データセットのうち16データセットで勝利している。

個々のデータセットを見ると、いくつかの興味深い挙動が見られる。きめの細かい分類タスクでは、性能に大きなばらつきが見られる。

これらのデータセットのうち2つ、Stanford CarsとFood101では、ゼロショットCLIPがResNet-50特徴量のロジスティック回帰を20%以上上回り、他の2つ、Flowers102とFGVCAircraftでは、ゼロショットCLIPが10%以上下回っている。OxfordPetsとBirdsnapでは、性能はかなり接近している。

これらの差は、主にWITとImageNetのタスクごとの監視量の違いによるものだと思われる。

PascalVOC2007のような「一般的な」物体分類データセットでは、性能は比較的似ており、すべてのケースでゼロショットCLIPがわずかに有利である。STL10では、CLIPは99.3%を達成し、これは学習例を使用していないにもかかわらず、新しい技術水準であると思われます。

ゼロショットCLIPは、動画中の行動認識を測定する2つのデータセットにおいて、ResNet-50を大幅に上回った。Kinetics700では、CLIPはResNet-50を14.5%上回った。UCF101においても、Zeroshot CLIPはResNet-50の特徴量を7.7%上回った。

これは、自然言語が、ImageNetの名詞中心のオブジェクト監視と比較して、動詞を含む視覚概念に対してより広い監視を提供するためであると推測される。

衛星画像の分類（EuroSATとRESISC45）、リンパ節腫瘍の検出（PatchCamelyon）、合成シーンでのオブジェクトのカウント（CLEVRCounts）、ドイツの交通標識認識（GTSRB）のような自動運転関連のタスク、最も近い車までの距離の認識（KITTI Distance）のようないくつかの専門的、複雑、または抽象的なタスクでは、ゼロショットCLIPはかなり弱いことがわかる。これらの結果は、より複雑なタスクにおけるゼロショットCLIPの能力の低さを浮き彫りにしている。

対照的に、熟練者でない人間は、計数、衛星画像分類、交通標識認識など、これらのタスクのいくつかを頑健にこなすことができ、改善の余地が大きいことを示唆している。

しかし、ほとんど全ての人間（そしておそらくCLIP）にとってリンパ節腫瘍分類のような、学習者が事前に経験したことのない難しいタスクに対して、数ショット伝達ではなくゼロショット伝達を測定することが意味のある評価であるかどうかは不明であることに注意する。

Contrastive Representation Learning

次回は、CLIP（Contrastive Language-Image Pre-training）の名前の元となっていて、CLIPの実装の基本になっている Contrastive Representation Learning （対比的表現学習）について説明します。

CLIP

Contrastive Representation Learning

2.3. Selecting an Efficient Pre-Training Method

効率的な事前訓練法を選択する

最先端のコンピュータビジョンシステムは、非常に大量の計算を使用する。

Mahajanら（2018）はResNeXt101-32x48dの学習に19GPU年、Xieら（2020）はNoisy Student EfficientNet-L2の学習に33TPUv3コア年を要した。

これらのシステムが1000個ののImageNetクラスのみを予測するために訓練されたことを考慮すると、自然言語から視覚概念のオープンセットを学習するタスクは困難であるように思われる。

私たちの努力の過程で、学習効率がnatural language supervision のスケーリングを成功させる鍵であることがわかり、この指標に基づいて最終的な事前学習方法を選択した。

キャプション予測と bag-of-word予測の難しさ

我々の最初のアプローチは、画像のキャプションを予測するために、画像CNNとテキストTransformerをゼロから共同で学習させた。しかし、この方法を効率的に拡張することは困難であった。

図2では、ResNet-50画像エンコーダの2倍の計算量を既に使用している6,300万パラメータの変換言語モデルが、同じテキストのbag-of-wordsエンコーディングを予測する、より単純なベースラインよりも3倍遅くImageNetクラスを認識するように学習することを示している。

これらのアプローチには重要な共通点がある。

それらは、各画像に付随するテキストの正確な単語を予測しようとする。これは、画像と共起する多種多様な説明、コメント、関連テキストのために困難なタスクである。

スケーリングの限界
論文 “6. Limitations” から

スケーリングは今のところ着実に性能を向上させており、継続的な改善の道筋を示唆しているが、ゼロショットCLIPが全体的な最先端性能に到達するためには、約1000倍の計算量の増加が必要であると推定される。これは現在のハードウェアでは訓練不可能である。CLIPの計算効率とデータ効率を改善するためのさらなる研究が必要であろう。

CLIPはまた、ディープラーニングのデータ効率の悪さにも対処していない。その代わりにCLIPは、何億もの学習例に拡張可能な監視ソースを使用することで補っている。

CLIPモデルのトレーニング中に見られるすべての画像が1秒に1枚の割合で提示された場合、32のトレーニングエポックにわたって見られる128億枚の画像を反復するのに405年かかる。

Contrastive Representation Learning への注目

画像に対するContrastive Representation Learning（対比的表現学習）における最近の研究では、対比的目的は、同等の予測目的よりも優れた表現を学習できることがわかっている（Tian et al.）

他の研究では、画像の生成モデルは高品質な画像表現を学習できるが、同じ性能を持つ対比モデルよりも1桁以上多くの計算量を必要とすることが分かっている（Chen et al.）

これらの知見に注目し、我々は、テキスト全体がどの画像と対になっているかだけを予測し、そのテキストの正確な単語は予測しないという、より簡単な代理タスクを解決するシステムの学習を試みた。

同じBag-of-Wordsエンコーディングのベースラインから始めて、図2で予測目的を対比目的に入れ替えたところ、ImageNetへのゼロショット転送率がさらに4倍効率化された。

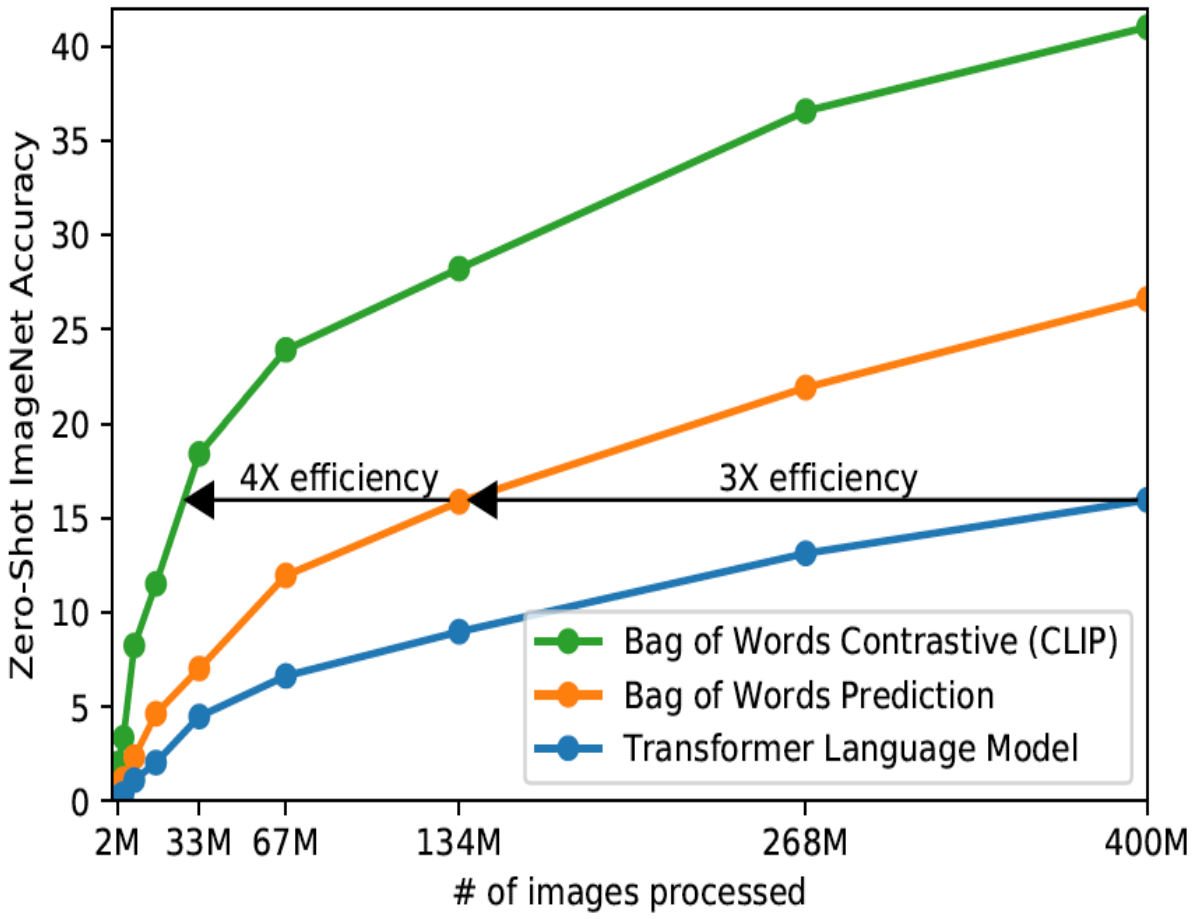


図2. CLIPは我々の画像キャプションベースラインよりもゼロショット転送においてはるかに効率的である。

表現力は高いものの、Transformerベースの言語モデルはゼロショットImageNet分類には比較的弱いことがわかった。ここでは、テキストのBag-of-words（BoW）エンコーディングを予測するベースライン（Joulin et al.）の予測目的をCLIPの対比的目的に入れ替えると、効率はさらに4倍向上する。

CLIPの方法
Contrastive Representation Learning

長さNのバッチの（画像、テキスト）ペアが与えられた時、CLIPは、バッチ全体で可能な$N \times N$個の（画像、テキスト）ペアのうちどれが実際に発生したかを予測するように学習される。

テキスト全体がどの画像と対になっているかだけを予測し、そのテキストの正確な単語は予測しない

CLIPの方法
Contrastive Representation Learning

長さNのバッチの（画像、テキスト）ペアが与えられた時、CLIPは、バッチ全体で可能な$N \times N$個の（画像、テキスト）ペアのうちどれが実際に発生したかを予測するように学習される。

これを行うために、CLIPは画像エンコーダとテキストエンコーダを共同で訓練することで、バッチ内のN個の正しいペアの画像とテキストの埋め込みのコサイン類似度を最大化する一方で、$N^2 - N$個の不正確なペアの埋め込みのコサイン類似度を最小化するように、マルチモーダル埋め込み空間を学習する。

Contrastive Representation Learning

CLIPの方法
Contrastive Representation Learning

長さNのバッチの（画像、テキスト）ペアが与えられた時、CLIPは、バッチ全体で可能な$N \times N$個の（画像、テキスト）ペアのうちどれが実際に発生したかを予測するように学習される。

これを行うために、CLIPは画像エンコーダとテキストエンコーダを共同で訓練することで、バッチ内のN個の正しいのペアの画像とテキストの埋め込みのコサイン類似度を最大化する一方で、$N^2 - N$個の不正確なペアの埋め込み度のコサイン類似度を最小化するように、マルチモーダル埋め込み空間を学習する。

これらの類似度スコアに対してsymmetric cross entropy lossを最適化する。

Contrastive Representation Learning とは何か

Contrastive Representation Learning:
A Framework and Review

Phuc H. Le-Khac, Graham Healy, Alan F. Smeaton

https://arXiv.org/2010.05113v2

「表現」の学習

表現学習とは、生の入力データ領域から特徴ベクトルやテンソルへのパラメトリック・マッピングを学習するプロセスを指す。

多くの場合、入力領域は高次元の空間（画像、ビデオ、サウンド、テキスト）を持ち、エンコードされた表現はより低次元の多様体に存在する。

すべての次元削減手法は、高次元の入力を低次元の表現に変換するが、これらの手法の中には、新しいデータサンプルに対して有意義に汎化するマッピングを学習しないものもある。

「良い」表現？

何をもって良い表現とするかは完全には明らかではない。

Bengio, Courville, and Vincent [9]の分析によると、良い表現とは、入力と表現の局所的な滑らかさ、一連の観察において時間的・空間的に首尾一貫していること、タスク間で共有される複数の階層的に組織化された説明因子を持つこと、因子間の単純な依存関係を持つこと、特定の入力に対してまばらに活性化されること、といった特性を持つ。

Deep Learningでの優れた表現

- 分散表現: 表現力があり、そのサイズに対して指数関数的な量の構成を表すことができる表現。これは、多くのクラスタリングアルゴリズムで学習されるワンホットエンコーディングのような他のタイプの表現とは対照的である:
- 抽象化と不変性:優れた表現は、入力データの小さな変化や局所的な変化に対して不変な、より抽象的な概念を捉えることができる:
- 分離された表現:優れた表現は、できるだけ多くの要素を捕捉し、できるだけ少ないデータを破棄すべきであるが、各要素はできるだけ分離されるべきである。学習システムにおける特徴の再利用を促進するだけでなく、説明可能性など他の目的にも有益である。

Constractive Presentation Learning
対比的表現学習

直観的には、対比的表現学習は比較することによって学習すると考えることができる。

ある（擬似）ラベルへのマッピングを学習する識別モデルや、入力サンプルを再構成する生成モデルとは異なり、対比的学習では、入力サンプル間の比較によって表現が学習される。

比較は、「似ている」入力の正のペアと「似ていない」入力の負のペアの間で行うことができる。

対比的学習の目的

対比的学習の目的は非常に単純である。

「似ている」サンプルの表現は近くにマッピングされるべきであり、「似ていない」サンプルの表現は埋め込み空間において遠くにマッピングされるべきである。

従って、肯定的なサンプルと否定的なペアのサンプルを対比することで、肯定的なペアの表現は引き寄せられ、否定的なペアの表現は遠くに押しやられることになる。

類似度分布

すべての入力サンプルxに対して人間のアノテーションyが必要な教師付き手法とは異なり、対比的学習アプローチでは、入力サンプルxに関して、

正の入力 $x^+ \sim p^+(\cdot|x)$ と
負の入力 $x^- \sim p^-(\cdot|x)$ のデータ分布

をサンプリングするために、類似度分布を定義するだけでよい。

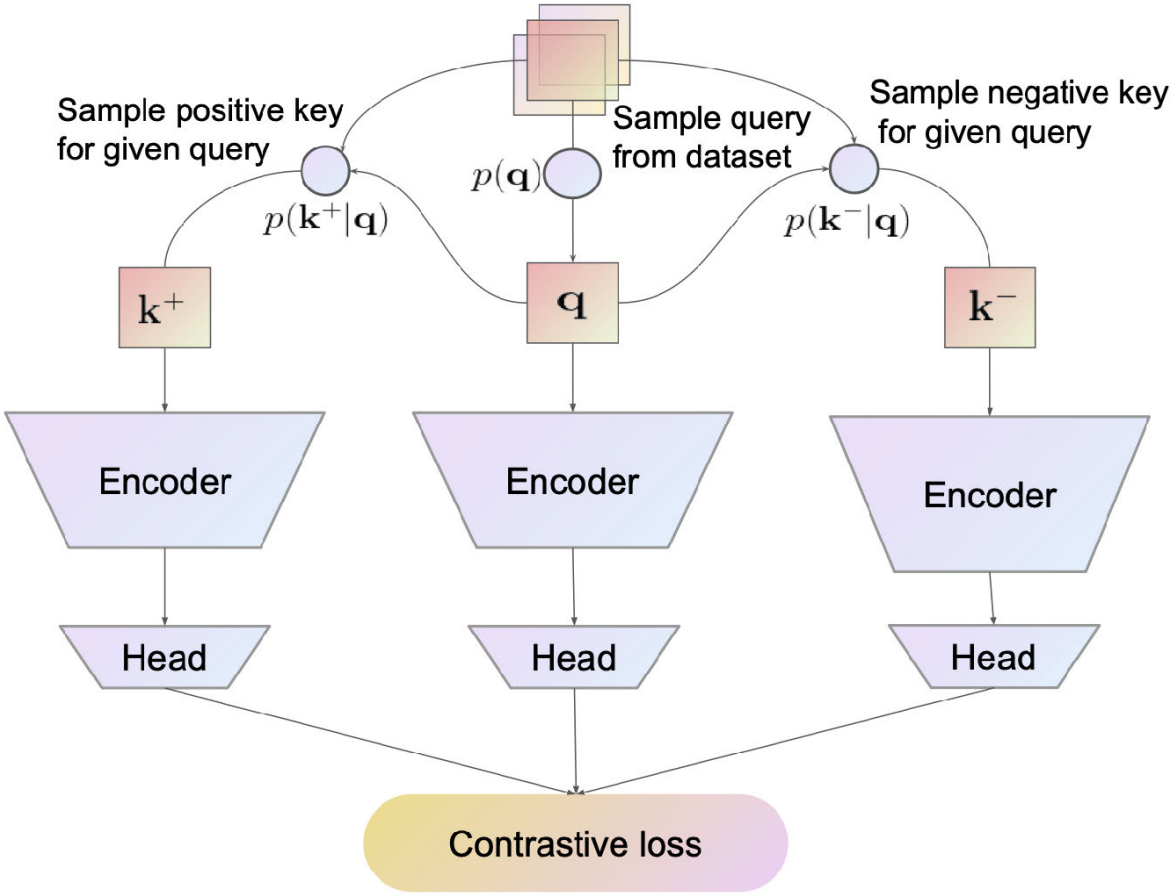
定義 類似度分布

類似度分布$p^+(q,k^+)$は、入力サンプルのペア上の結合分布であり、対比的学習タスクにおける類似度（および非類似度）の概念を形式化したものである。

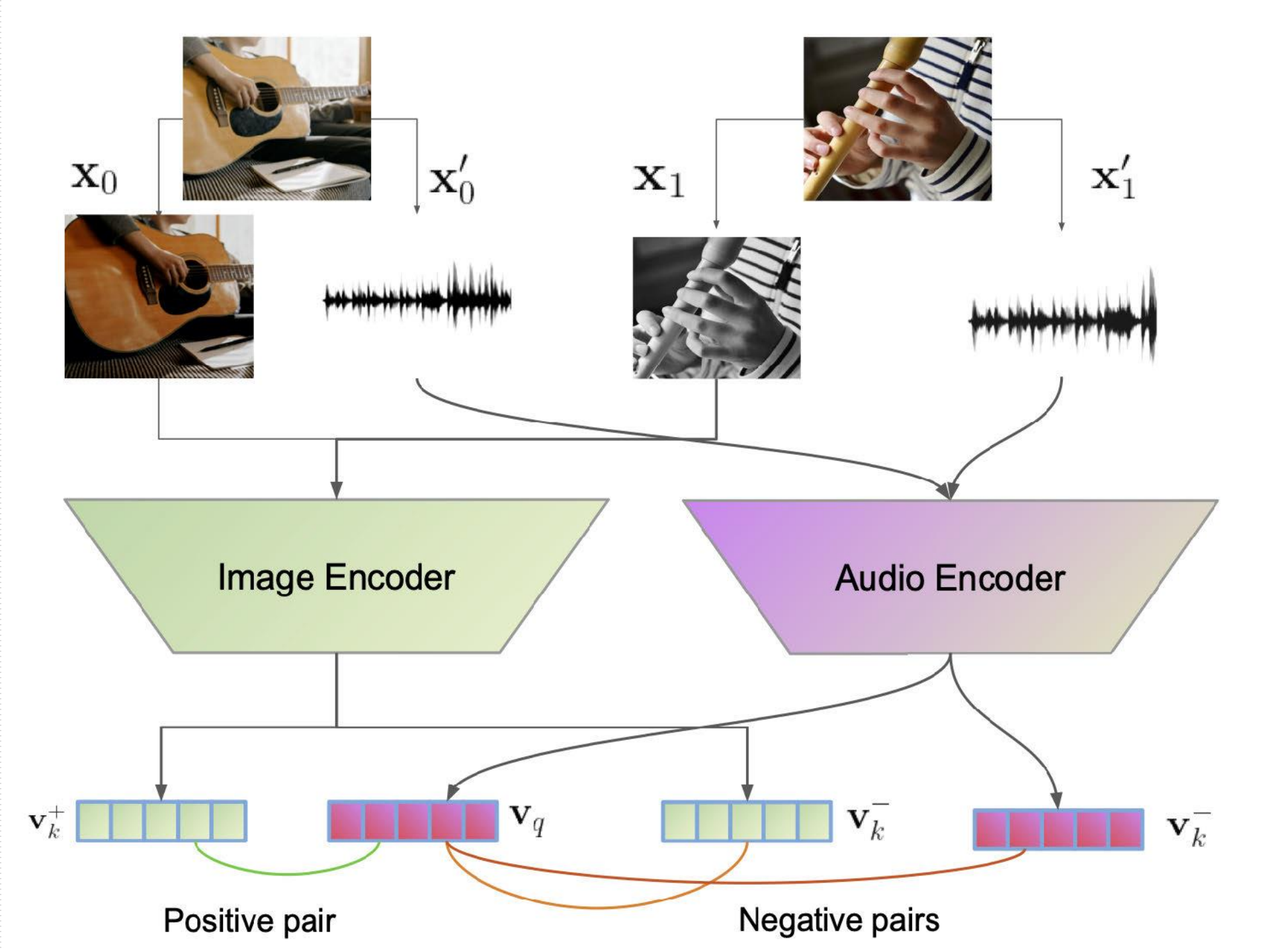
データ分布が1つの入力サンプル$p(x)$に対して定義される他の機械学習手法とは異なり、対比的手法で必要とされる類似度は、サンプルのペアの結合分布$p(q,k)$から入力を受ける。

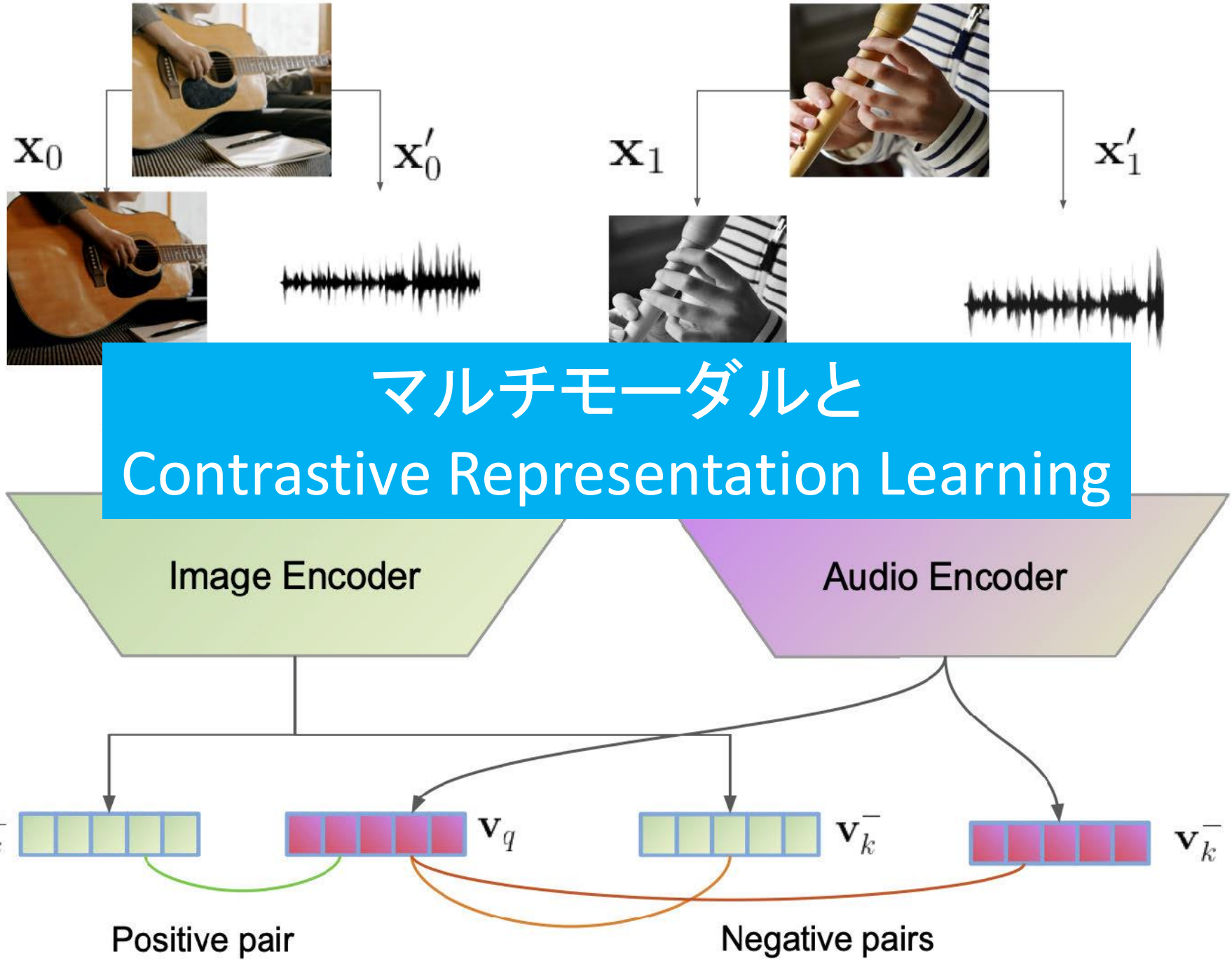
キーがこの類似度分布からサンプリングされた場合、クエリqに対して正のk^+とみなされ、非類似度分布$p^-(q,k^-)$からサンプリングされた場合、負のk^-とみなされる。

図3. 対比的表現学習フレームワークの概要



その構成要素は、クエリに対する正と負のキーをサンプリングするための類似度と非類似度分布、各データモダリティに対する1つ以上のエンコーダと変換ヘッド、正と負のペアのバッチを評価する対比的損失関数である。


$\mathbf{x}_0$
$\mathbf{x}'_0$
$\mathbf{x}_1$
$\mathbf{x}'_1$
Image Encoder
Audio Encoder
$\mathbf{v}_k^+$
$\mathbf{v}_q$
$\mathbf{v}_k^-$
$\mathbf{v}_k^-$
Positive pair
Negative pairs


$\mathbf{x}_0$
$\mathbf{x}'_0$
$\mathbf{x}_1$
$\mathbf{x}'_1$
マルチモーダルと
Contrastive Representation Learning
Image Encoder
Audio Encoder
$\mathbf{v}_k^+$
$\mathbf{v}_q$
$\mathbf{v}_k^-$
$\mathbf{v}_k^-$
Positive pair
Negative pairs


Adobe Firefly