

# Grid R-CNN

2018/2/19 Xin Lu<sup>1</sup> Buyu Li Yuxin Yue Quanquan Li Junjie Yan

<https://arxiv.org/pdf/1811.12030v1.pdf>

## どんなもの？

物体検出の精度(たぶん bounding box の精度)を向上させたもの  
2ステージ物体検出において、box座標をRegressionするかわりに、Boxのグリッド点をヒートマップで学習する。

## 技術の肝、先行研究との比較

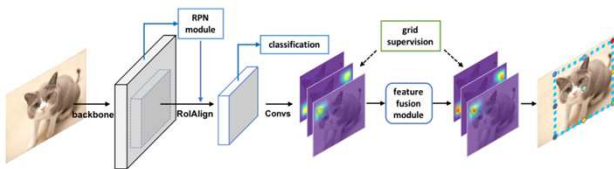
図D1のように、入力画像に対してbackboneネットワークで特徴抽出、Region Proposal NetworkおよびROIAlignでROIクロップをおこなう。ここまではMask R-CNNと同じである。

grid prediction branch :

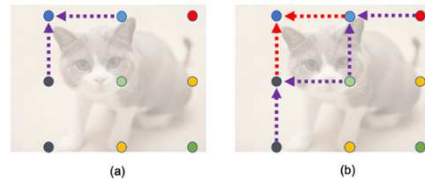
クロップしたfeature map ( $14 \times 14$ ) に対し、8層のdilated convolution層、および2層のdeconvolution層を経て、 $56 \times 56 \times (N \times N)$  のfeature mapを得る。 $N \times N$  はグリッドの点数であり、標準は $3 \times 3$ である。Ground Truthは正解グリッド点を中心とする+型の5画素がpositiveとされており、推定されたヒートマップとのBinary Cross Entropy Lossにより学習される。

feature fusion module (図D2) :

隣接するgrid点には空間的相関がある。feature fusion moduleでは隣のgrid点を用いてgrid featureを修正する。Fiを注目するgrid点のfeatureとすると、近隣のFjに対しいくつかの $5 \times 5$  convolution層を通し、 $T_j \rightarrow i(F_j)$ を作る。Fiとそれらの和を最終的なgrid featureとする。推定時は、得られた各グリッドヒートマップにおいて、最大値をとる座標がピックアップされて元画像にマッピングされる。



図D1 Grid R-CNNのパイプライン。(f5より引用)



図D2 Feature Fusion Moduleの説明図。(f5より引用)

## 結果

方法	背骨	AP	AP <sub>s</sub>	AP <sub>75</sub>	AP <sub>s</sub>	AP <sub>M</sub>	AP <sub>L</sub>
より高速なR-CNN	ResNet-50	33.8	55.4	35.9	17.4	37.9	45.3
グリッドR-CNN	ResNet-50	<b>35.9</b>	54.0	38.0	18.6	40.2	47.8
R-CNNとFPNの高速化	ResNet-50	37.4	59.3	40.3	21.8	40.9	47.9
グリッドR-CNN w FPN	ResNet-50	<b>39.6</b>	58.3	42.4	22.6	43.8	51.5
R-CNNとFPNの高速化	ResNet-101	39.5	61.2	43.1	22.7	43.7	50.8
グリッドR-CNN w FPN	ResNet-101	<b>41.3</b>	60.3	44.4	23.4	45.8	54.1

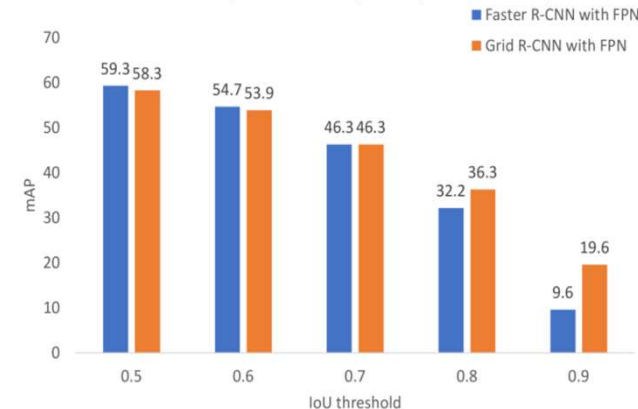


図5: 0.5から0.9のIoUしきい値での間隔0.1でのAP結果。

COCOで評価。Faster-RCNN + ResNet-50 + FPNに対してmAP 37.4%から39.6%へ2.2%の向上。

IoU閾値を厳しくした場合の精度向上が著しい  
(IoU=0.8でmAP+4.1%,  
IoU=0.9でmAP+10.0%)。

## 次に読むべき論文は？

R-FCN、FPN、マスクR-CNN、CornerNet

日付

# R-FCN

2016/7/21 Jifeng Dai Yi Li Kaiming He Jian Sun

<https://arxiv.org/pdf/1605.06409v2.pdf>

## どんなもの？

FASTER RCNN等でROIプーリング後の全結合(FC)レイヤーを削除し、代わりに、ROIプーリング前に出力ch数が $k \times k \times c+1$ のポジティブセンシティブスコアマップを配置

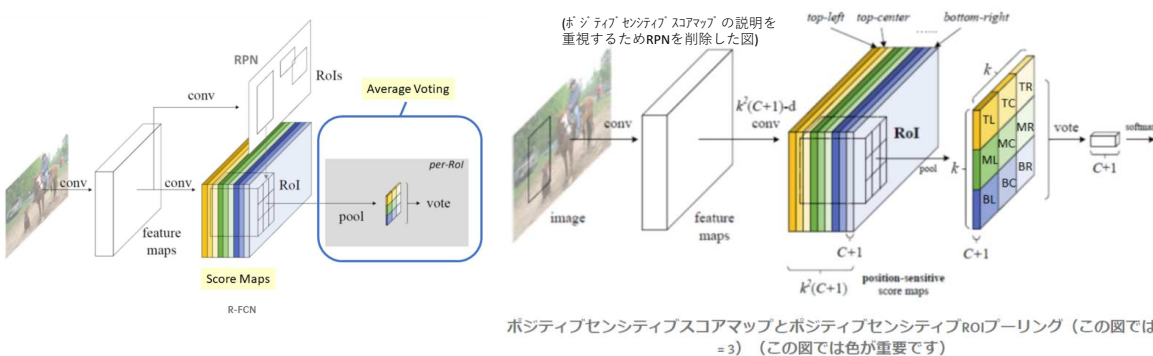
## 技術の肝、先行研究との比較

Faster R-CNN等の従来の物体検出の場合、ROIプーリング後に、分類および境界ボックス回帰のために全結合(FC)レイヤーを通過しますが、このFCレイヤーはROI間で共有されず、時間がかかり、RPNアプローチが遅くなる。また接続(パラメータ)の数が増え、より複雑になる。

R-FCNでは、ROIプーリング前に出力ch数が $k \times k \times c+1$ のポジティブセンシティブスコアマップを配置  
ROIプーリング後のすべての提案領域は、同じスコアマップのセットを使用して、単純な計算である平均投票を実行します。したがって、ほぼ無料のROIレイヤーの後に学習可能なレイヤーはありません。  
結果として、R-FCNは、競争力のあるmAPを備えたFaster R-CNNよりも高速です。

( $k$ はROIプーリングの縦or横、 $C+1$ はオブジェクトクラス数+バックグラウンド)

これらの $k^2$ 特徴マップは、検出するオブジェクトの{左上(TL)、上中央(TC)..、右下(BR)}を表す。



ポジティブセンシティブスコアマップとポジティブセンシティブROIプーリング (この図では $k=3$ ) (この図では色が重要です)

## 結果

	training data	test data	AP@0.5	AP	AP small	AP medium	AP large	test time (sec/img)
Faster R-CNN [9]	train	val	48.4	27.2	6.6	28.6	45.0	0.42
R-FCN	train	val	48.9	27.6	8.9	30.5	42.0	0.17
R-FCN multi-sc train	train	val	49.1	27.8	8.8	30.8	42.2	0.17
Faster R-CNN +++ [9]	trainval	test-dev	55.7	34.9	15.6	38.7	50.9	3.36
R-FCN	trainval	test-dev	51.5	29.2	10.3	32.4	43.3	0.17
R-FCN multi-sc train	trainval	test-dev	51.9	29.9	10.8	32.8	45.0	0.17
R-FCN multi-sc train, test	trainval	test-dev	53.2	31.5	14.3	35.5	44.2	1.00

MS COCOデータセット

## 次に読むべき論文は？

FPN、マスクR-CNN、CornerNet

日付

このストーリーでは、マイクロソフトと清華大学によるR-FCN（Region-based Fully Convolutional Network）について簡単に説明します。ポジティブセンシティブスコアマップにより、競合時間の精度を維持しながら、推論時間はFaster R-CNNよりもはるかに高速です。

1. R-CNNに対するR-FCNの利点

R-CNN、Fast R-CNN、Faster R-CNNなどの従来の地域提案ネットワーク（RPN）アプローチの場合、提案領域はRPNによって最初に生成されます。次にROIプーリングが行われ、分類および境界ボックス回帰のために完全に接続された（FC）レイヤーを通過します。ROIプーリング後のプロセス（FCレイヤー）はROI間で共有されず、時間がかかり、RPNアプローチが遅くなります。また、FCレイヤーは接続（パラメーター）の数を増やし、複雑さも増します。

R-FCNには、まだ提案領域を取得するRPNがありますが、R-CNNシリーズとは異なり、ROIプーリング後のFCレイヤーは削除されます。代わりに、すべての主要な複雑さがROIプーリングの前に移動され、スコアマップが生成されます。ROIプーリング後のすべての提案領域は、同じスコアマップのセットを使用して、単純な計算である平均投票を実行します。したがって、ほぼ無料のROIレイヤーの後に学習可能なレイヤーはありません。結果として、R-FCNは、競争力のあるmAPを備えたFaster R-CNNよりも高速です。

2. ポジティブセンシティブスコアマップとROIプーリング

2.1ポジティブセンシティブスコアマップ

簡単にするために、RPNを削除します。そして、Cクラスを検出する必要があります。（C+1）は、C個のオブジェクトクラスとバックグラウンドクラスを意味します。ポジティブセンシティブスコアマップの直前の最初の多数の畳み込みの後、k<sup>2</sup>（C+1）-d畳み込みを実行します。クラスごとに、k<sup>2</sup>特徴マップがあります。これらのk<sup>2</sup>特徴マップは、検出するオブジェクトの{左上（TL）、上中央（TC）、...、右下（BR）}を表します。

2.2ポジティブ/センシティブROIプーリング

ROIプーリングの場合、サイズがk<sup>2</sup>の（C+1）機能マップ、つまりk<sup>2</sup>（C+1）が生成されます。プーリングは、図の同じ面積と同じ色でプールされるという意味で行われます。（C+1）1dベクトルを生成するために、平均投票が実行されます。最後に、ソフトマックスがベクターで実行されます。リージョンの提案がオブジェクトと以下のように重ならない場合、ノーの投票があります。

2.3境界ボックス回帰

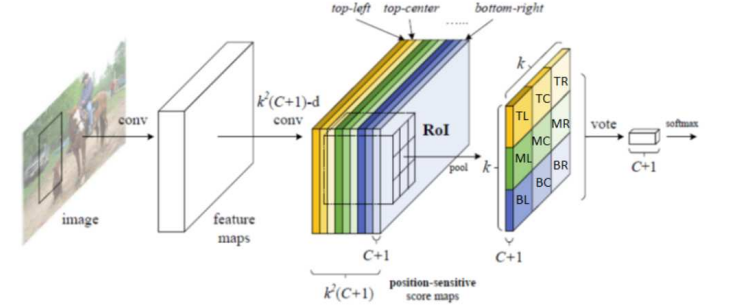
クラスに依存しない境界ボックス回帰が実行されます。つまり、回帰はクラス間で共有されます。k<sup>2</sup>（C+1）-d畳み込み層では、兄弟の4k<sup>2</sup>-d畳み込み層が追加されます。この4k<sup>2</sup>マップのバンクで位置依存ROIプーリングが実行され、各RoIに対して4k<sup>2</sup>-dベクトルが生成されます。次に、平均投票によって4次元ベクトルに集約されます。これは、バウンディングボックスの{tx、ty、tw、th}（位置とサイズ）を表します。これは、Fast R-CNNと同じです。

3.1バックボーンアーキテクチャ

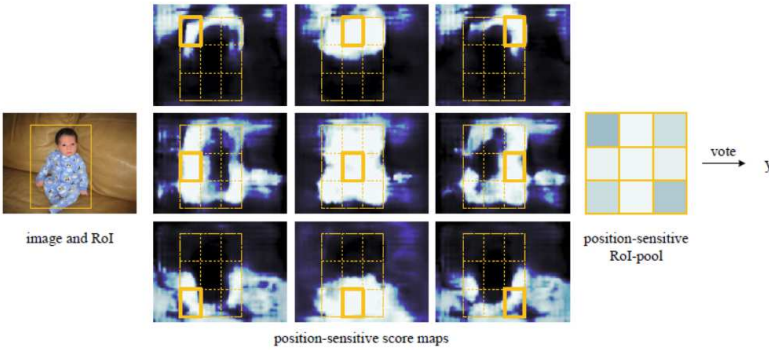
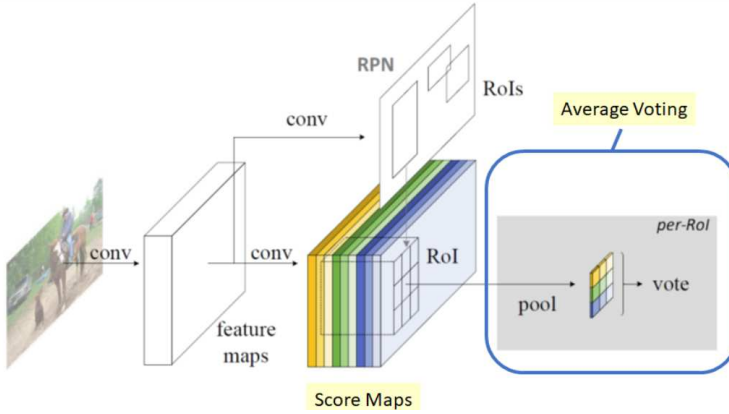
ImageNetから事前トレーニングされたResNet-101の最初の100の変換は、ポジティブセンシティブスコアマップの直前に機能マップ

3.2トレーニング

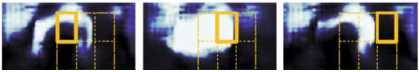
損失は次のFast R-CNNです。



ポジティブセンシティブスコアマップとポジティブセンシティブROIプーリング（この図ではk = 3）（この図では色が重要です）



ポジティブ/センシティブROIプーリングの例



# ANOGAN

2017/3/17 Thomas Schlegl, Philipp Seeb, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, Georg Langs

<https://arxiv.org/pdf/1703.05921.pdf>

## どんなもの？

GAN-baseで初めて異常検知をした。

## 議論はある？

- ・ 通常データでノイズ $z$ が見つからない可能性
- ・  $z$ の探索の非効率さ

## どうやって有効だと検証した？

医療画像でテスト、異常データはテストデータのみに含まれ、訓練データは正常データのみ。

## 先行研究と比べて何がすごい？

Mapping等を工夫する事によって、異常データのアノテーションがなくても異常検知できる。

## 技術の手法や肝は？

異常データはGANによってモデリングされないという仮定を元に、入力画像を生成するようなノイズ $z$ を勾配法で探索し、最適化したノイズの元での生成画像と入力画像の距離を計る(二乗誤差+Discriminatorのfeature matching)

## 次に読むべき論文は？

EfficientGAN、ADGAN





どんなもの？

GAN-baseで初めて異常検知をした。

議論はある？

一般的にGANのチューニングは難しく、学習や探索における不安定さが問題になっている。

どうやって有効だと検証した？

MNIST、CIFAR-10、LSUNでSOTA

先行研究と比べて何がすごい？

ANOGANに対し、Zの探索時にGeneratorを学習させる。初期シードなどに依存しないような工夫を行っている。

技術の手法や肝は？

探索時においてGeneratorを学習する際に、パラメタ $\theta$ を用いてモデルに表現力を持たせている点と、GANや探索時におけるハイパーパラメータの問題を平均を取る事で解決している。

次に読むべき論文は？

EfficientGAN

日付

# Efficient GAN

2019/5/1 Houssam Zenati, Chuan Sheng Foo, Bruno Lecouat, Gaurav Manek, Vijay

<https://arxiv.org/pdf/1802.06222.pdf>

どんなもの？

推論速度の速いGAN-baseでの異常検知

議論はある？

パラメタが多い分チューニングが難しそう。

どうやって有効だと検証した？

MNISTとKDDCU99に対してSOTA,画像データ、配列データ共に良い性能。

先行研究と比べて何がすごい？

ANOGANの改良版。Encoderを学習することによって推論速度を大幅に速めた。

技術の手法や肝は？

Encoder⇒Decoder型のGANを用いて、reconset errorとDiscriminator出力を組み合わせた指標で異常度を測る。Encoderの関数を学習する事で、潜在空間におけるZの探索をする必要がなくなった。

次に読むべき論文は？

BiGAN : Adversarial Feature Learning

日付