OD : object detection

التاريخ: / /

Deep detection

⇐ في الصورة الواحدة يمكن تصنيف أكثر من غرض مثلاً :
يمكن وجود أكثر من غرض في الصورة (كلب، قطة، بطة)
وهناك إمكانية لتصنيف كل الأغراض في نفس الصورة.

1 what are the challenges of object detection ? (OD)

التحديات التي تواجه

1- قد تحتوي الصورة على عدة class فمثلاً لو كانت صورة حديقة ستحتوي على عدة أشياء مختلفة وقد يتواجد عدة أشخاص من نفس النوع وهذا يعني تكرار النادع.

2- تحديات في تحديد حجم وموقع الكائن الموجود في الصورة باستخدام Bounding Box وتحديات في تحديد دقة عملية BB

BB : Bounding Box

* object detection evaluation

⇐ عند إجراء اختبار "Test" على النموذج المستخدم لتحديد مواقع الأشياء والكائنات في الصور بإسقاط Bounding Box يتم التنبؤ بمواقع المربعات وتصنيف الأشياء الموجودة و تحديد درجة الثقة confidence scores لكل مربع

⇐ ثم يتم تحديد فيما إذا كان المربع المكتشف ذو إيجابي حقيقي True positive أو False positive

ثم يتم رسم منحنى Recall - precision curve لكل فئة

حساب المساحة تحت المنحنى لتحديد متوسط الدقة Avg precision لكل فئة.

⇐ يتم حساب معدل متوسط الدقة "mAP" ويستخدم هذا

التاريخ: / /

المقياس لتقييم أداء نموذج تحديد المواقع للأشياء وتحديد مدى قدرته على التعرف على الأشياء بدقة وفعالية.

حيث:

$$\text{الدقة Precision} = \frac{\text{True Positive detections}}{\text{total detections}}$$

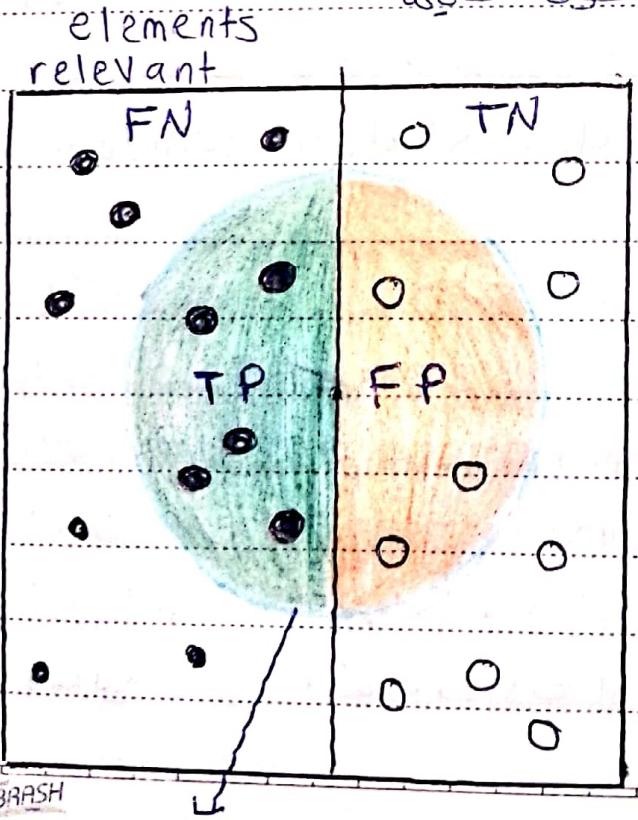
$$= \frac{TP}{TP + FP}$$

$$Recall = \frac{\text{True Positive detection}}{\text{total positive test instances}}$$

$$= \frac{TP}{TP + FN}$$

حيث Total detections عدد الحالات الكلية التي تم كشفها

TP عدد الحالات المكشوفة الصحيحة



حالة FP اكتشف غرض ولكن ليس هو الغرض المطلوب مثلاً صنف الغرض على أنه وجه وهو ليس وجه

FN: في حال كان وجه مثلاً ولم يتم كشفه

TP: حال كان وجه وقد اكتشف على أنه وجه

التاريخ: / /

⇒ How many selected items are relevant?

$$precision = \frac{TP}{TP+FP}$$

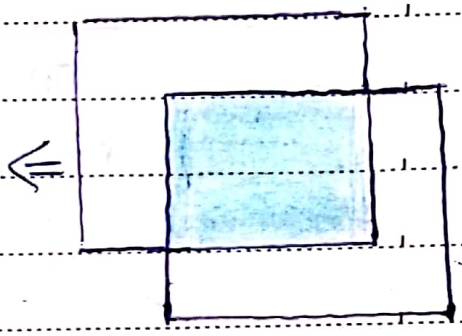
⇒ How many relevant items are selected?

$$Recall = \frac{TP}{TP+FN}$$

تقاطع

$$\frac{Area(GTN\,Det)}{Area(GTU\,Det)} > 0.5$$

اقتباع



$$\frac{\text{النقطة التي تروي الفرض}}{\text{كامل النقطة}} > 0.5$$

* هام

why Recall increase when precision decreases? precision تقل عند Recall لماذا تزيد

لأنه عند زيادة Recall يجب انقاص المقام ولكن لا يجب انقاص TP لأنها موجودة بالبسط والمقام لذلك تنقص FN + الـ TP والقوانين السابقة

conceptual approach: sliding window detection

إن الطريقة المتقدمة لتحديد مواقع الأشياء في الصورة من خلال تحريك window نافذة = مربع على كامل الصورة وهذه النافذة تجري على نموذج الكشف (detection model) وسيتم إحاطة العرض بمربع (Bounding Box) وتتابع هذه العملية لكل تقييم النموذج المستخدم لتحديد المواقع ويجب أن يحقق هذا النموذج فعالية عالية وخفض معدلات الإيجابيات الخاطئة FN للحصول

التاريخ: / /

على نتائج دقيقة جدا، ومن ذلك فإن هذه العملية صعبة التنفيذ والتطبيق على مجموعة واسعة من المقاييس ونسب العرض إلى الطول (Aspect ratios) وهذه العملية تستهلك الكثير من الموارد والوقت لذلك هناك بعض طرق تجريبية لتجريب المواقع في الصور مثل استخدام المقياس الهرمي "pyramid scale" والتعلم العميق لتحسين دقة التدريب.

* Histogram of oriented gradients (HOG)

يتم تقسيم الصورة إلى كتل (blocks) وحساب عدد تكراري لكل كتلة "histogram"

Image input → Normalize gamma & color → compute gradients → weights vote into spatial & orientation cells → contrast normalize over overlapping spatial blocks

* pedestrian detection with HOG

يتم تدريب نموذج لتدريب المشاة في الصور باستعمال آلية دعم المتجهات الخطية linears svm باستخدام مجموعة من الصور التي تحتوي على مشاة وغيرها من الأشياء كمجموعة تدريب.

وعند إجراء الاختبار يتم تطبيق هذا النموذج على Feature map ويتم البحث عن local maxima في الاستجابة للنموذج ويتم تكرار هذه العملية على مجموعة من الأحجام المختلفة لتدريب المشاة في الصور أو مقاطع الفيديو.

التاريخ: / /

R-CNN: Region proposals + CNN Features

R-CNN تعمل وفق الخطوات التالية:

Input image → Region proposals → warped image regions → Forward each region through ConvNet → classify regions with SVMs.

صورة input image في الصورة التي يتم العمل عليها و Region proposals في اقتراح مجموعة مناطق ذات حجوم مختلفة من الصورة المدخلة وهذه المناطق قد تحوي أغراض مراد كشفها warped image regions تدريبها على حجم كل منطقة و Forward each region convnet تحويله الى حجم ثابت في تطبيق CNN على هذه المناطق لكشف الأغراض و SVMs في القيام بعملية تصنيف لهذه الأغراض.

إيجابيات R-CNN : pros

1. دقيقة : تحقق نتائج دقيقة في تحديد مواقع الأشياء في الصور
2. قابلة للتطبيق على العديد من تصاميم الشبكات المختلفة

سلبيات R-CNN : cons

1. ليست نظام واحد شامل بل يتطلب الأمر تدريب أجزاء مختلفة للتعرف بشكل منفصل
2. يتطلب التدريب الكثير من الوقت والموارد يصل الى (48) ساعة
3. يتطلب التدريب مرور الصور عبر الشبكات العصبونية العميقة حوالي "2000" مرة
4. تكون عملية الاستدلال بطيئة (Inference) وتستغرق حوالي 47 ثانية لكل صورة باستعمال (VGG16) وبالتالي هذه الطريقة غير فعالة في الحالات التي يتطلب فيها الأمر الكشف عن الأشياء بسرعة

التاريخ: / /

* Fast R-CNN

تعمل Fast R-CNN وفق الخطوات:

input image → Forward whole image through ConvNet → conv5 feature map of image → RoI pooling layer → fully-connected layers → Bounding-box regresser

حيث يستخدم ConvNet لاستخراج العالم (features) من الصورة conv5 Feature map... و Feature map لتحويل الصورة إلى مجموعة من التي تحتوي على معلومات عديدة عن المناطق المهمة في الصورة تسمى الحواف والأشكال و Region of Interest في RoI pooling للحصول على المعلومات المحددة من المواقع المحتملة التي تم التنبؤ بها ثم تتم عملية Bounding-box و fully connected لتحديد المواقع و التصنيف.

* Fast R-CNN training

بعد القيام بخطوات Fast R-CNN يتم القيام بخطوة جديدة في multi-task loss أي تستخدم هذه الخطوة لكي تزيد دقة التصنيف بالاعتماد على الخسارة loss:

$$L(y, p, b, \hat{b}) = -\log p(y) + \lambda \mathbb{I}[y \geq 1] L_{reg}(b, \hat{b})$$

softmax loss / regression

حيث L loss for ground truth class y في $p(y)$، و $\hat{b}$ في predicted class probabilities، و b في predicted box.

// Yolo : You only look once //

التاريخ: / /

$$L_{reg}(b, \hat{b}) = \sum_{i=\{x,y,w,h\}} smooth_{L_1}(b_i - \hat{b}_i)$$

$$smooth_{L_1}(x) = \begin{cases} 0.5x^2 & \text{if } |x| < 1 \\ |x| - 0.5 & \text{otherwise} \end{cases}$$

مقارنة بين R-CNN ← Fast R-CNN ← Faster R-CNN:

system	time	07 data	07+12 data
R-CNN	~50 s	66.0	-
Fast R-CNN	~2 s	66.9	70.0
Faster R-CNN	198 ms	69.6	73.2

objects detection by Yolo

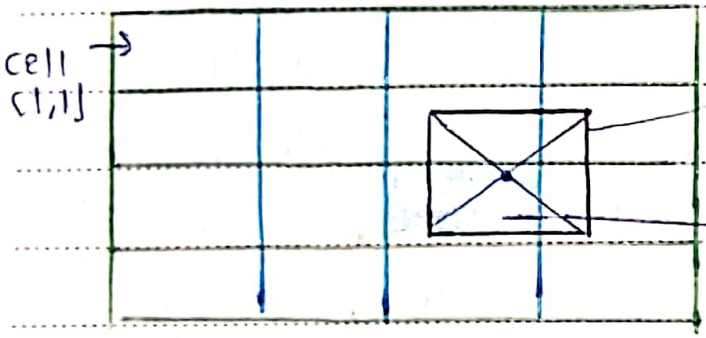
تعمل Yolo على تقسيم صورة الدخل إلى (SxS) خلية (grid = cell) وكل خلية تمثل المربع الحاوي فيها ويكون هذا المربع واقعاً في هذه الخلية يجب أن يكون مركز الإطار (x, y) المحدد للمربع يقع في هذه الخلية.

حيث سيتم تحديد مواقع وجود الكائنات في صورة ما وتحديد

التاريخ: / /

عددها باستخدام مربعات تسمى // boundary boxes //
ويحتوي كل مربع على ٥ عناصر (x, y, w, h)
بالإضافة إلى درجة الثقة box confidence score بحيث:

- (x, y) هما الإحداثيات الأفقية والشاقولية
- (w, h) هما العرض والارتفاع
- و $0 < x < 1$, $0 < y < 1$



Input image

وتكون خطوات YOLO:

1- استخراج خصائص الصورة باستخدام شبكة تحويل الصورة إلى خريطة ملامح conv feature maps وبدقة 7X7
2- يتم إضافة طبقتي Fully connected (FC) إلى map للتنبؤ بنتائج التصنيف وتحديد عدد المربعات لكل فئة، ويتم توقع نتائج التصنيف على شكل نقاط درجات الثقة score لكل فئة ممكنة بالإضافة إلى عدد مربعين (bbox) مع الثقة لكل مربع
3- بالنسبة لبيانات PASCAL فإن الناتج المتوقع هو 7X7X30 الـ 30 هي ناتج $20 + 2 * (4+1)$
الـ 20 هي عدد الفئات ممكنة التصنيف.

التاريخ: / /

* الهيكلية الرئيسية لنموذج YOLO:

⇐ 24 convalution layers

⇐ طبقتين (F.C) Fully connected

⇐ مخرجات ذات أبعاد 7X7X1024

أما لتوليد المربعات الحدودية:

يتم في YOLO توليد عدة مربعات حدودية bounding boxes لكل cell في صورة الدخل بهدف تغطية كل الحواف المحتملة لموجود كائنات ولتوليد المربع الحدودي المؤول عن الكائن الحقيقي في الصورة يتم اختيار المربع الذي حقق أعلى قيمة لقياس IOU (intersection over union)

وعند حساب الخسارة loss المتنبأ بالكيانات الحقيقية يتم حساب الخسارة فقط للمربع الحدودي الذي تم اختياره.

* لحساب loss في نموذج YOLO: يتم في YOLO استخدام خسارة مربعات الأخطاء "sum-squared error" بين التنبؤات و ground truth والتي تتكون من ثلاث مكونات رئيسية:

1. خسارة التصنيف classification loss
2. خسارة الموقع localization loss: يتم حسابها باستخدام مقياس (IOU) وهي الخسارة المرتبطة بدقة تحديد موقع المربع الحدودي
3. خسارة الثقة confidence loss وهي الخسارة المرتبطة بمدى ثقة وجود الكائن داخل المربع الحدودي

التاريخ: / /

* The localization loss:

$$\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{obj} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right]$$

$$+ \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{obj} \left[(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 \right.$$

$$\left. + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2 \right]$$

حيث λ_{coord} في:

increase the weight for the loss in the boundary box coordinates

* $\mathbb{1}_{ij}^{obj}$: 1 if the j th boundary box in cell i is responsible for detecting the object, otherwise 0

* confidence loss

إذا تم الكشف عن غرض في box:

$$\sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{obj} (C_i - \hat{C}_i)^2$$

إذا لم يتم الكشف عن الغرض في الـ box:

$$\lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{noobj} (C_i - \hat{C}_i)^2$$

حيث: $\hat{C}_i$ is the box confidence score of the box j in cell i

$\mathbb{1}_{ij}^{noobj}$: is the complement of $\mathbb{1}_{ij}^{obj}$

التاريخ: / /

λ_{noobj} : weights down the loss when detecting background

* class Probability

$$\sum_{i=0}^{S^2} \mathbb{1}_i^{obj} \sum_{c \in classes} (p_i(c) - \hat{p}_i(c))^2$$

ملاحظة: وفقاً لعمل شبكة Yolo وتقسيم الصورة إلى خلايا وفحص كل خلية هل كانت تحوي أو تعرض أو لا بالتالي من الممكن أن يكون العرض يغطي أكثر من خلية فهذا يسبب رسم أكثر من إطار على نفس العرض ولكل هذه المشكلة نقوم بعمل حذف للقيم الدنيا