



جامعة تشرين

كلية الهندسة الميكانيكية والكهربائية

قسم هندسة الاتصالات والالكترونيات

السنة الخامسة

برمجة وإدارة الشبكات – مشروع فصلي

عنوان البحث

**Detection objects using yolo**

إشراف

د. مهند عيسى

إعداد الطلاب

رشا عصفور

فراس بركات

2022-2021

## جدول المحتويات

4	مقدمة:
5	1- كشف الكائن:
5	2- اكتشاف الأشخاص:
6	3- أهمية اكتشاف الكائنات:
6	4-كشف الكائنات والتعلم العميق:
7	5-اخر التطورات التكنولوجية:
7	6-كيف يعمل اكتشاف الكائن:
9	7- معالم في أحدث اكتشاف للكائن:
10	8-حالات وتطبيقات اكتشاف الأجسام:
12	9-YOLO (You Only Look Once):
13	10-اكتشاف الكائنات باستخدام إطار YOLO :
13	11-الآلية عمل إطار YOLO:
17	12-الآلية تشفير صناديق التثبيت:
19	<b>Intersection over Union and Non-Max Suppression</b>
23	13- صناديق المرساة Anchor Boxes:
25	14- الجمع بين الأفكار:
26	15- اختبار:
28	16- الكود البرمجي:
29	17- المراجع:



## مقدمة

هذا البحث مقدمة لاكتشاف الكائن ويوفر لمحة عامة عن أحدث خوارزميات "اكتشاف كائن الرؤية الحاسوبية". اكتشاف الكائنات هو مجال رئيسي في الذكاء الاصطناعي، مما يسمح لأنظمة الكمبيوتر «برؤية» بياناتها عن طريق اكتشاف الأشياء في الصور المرئية أو مقاطع الفيديو.

على وجه الخصوص، سوف تتعلم عن :

- ما هو اكتشاف الكائن وكيف تطور خلال السنوات 20 الماضية.
- أنواع طرق الكشف عن أجسام الرؤية الحاسوبية.
- نسرذ الأمثلة وحالات الاستخدام وتطبيقات الكشف عن الكائنات.
- أشهر خوارزميات اكتشاف الكائنات اليوم.
- خوارزميات التعرف على الكائن الجديدة التي تم تقديمها في عام 2022.

## 1- كشف الكائن:

اكتشاف الكائنات هي مهمة مهمة للرؤية الحاسوبية تستخدم للكشف عن حالات الأشياء المرئية من فئات معينة (على سبيل المثال، البشر أو الحيوانات أو السيارات أو المباني) في الصور الرقمية مثل الصور أو إطارات الفيديو. الهدف من اكتشاف الكائنات هو تطوير نماذج حسابية توفر المعلومات الأساسية التي تحتاجها تطبيقات الرؤية الحاسوبية: «ما هي الكائنات التي توجد فيها؟»



الشكل (1) - اكتشاف الكائنات مع صناديق الحدود في إطار الفيديو

## 2- اكتشاف الأشخاص:

اكتشاف الشخص (بالإنجليزية: **Person detection**) هو نوع متغير من اكتشاف الأجسام يستخدم لاكتشاف «شخص» من الفئة الأولية في الصور أو إطارات الفيديو. يعد الكشف عن الأشخاص في تدفقات الفيديو مهمة في أنظمة المراقبة بالفيديو الحديثة. توفر خوارزميات التعلم العميق الأخيرة نتائج قوية لاكتشاف الشخص. يتم تدريب معظم تقنيات كشف الأشخاص الحديثة على وجهات النظر الأمامية وغير المتماثلة.

ومع ذلك، فإن نماذج التعلم العميق مثل **YOLO** التي تم تدريبها على اكتشاف الشخص على مجموعة بيانات الرؤية الأمامية لا تزال توفر نتائج جيدة بشكل ملحوظ عند تطبيقها على العد العام للشخص (TPR بنسبة 95٪، FPR حتى 0.2٪). لمعرفة المزيد حول اكتشاف الشخص وكيفية استخدام إخراج جهاز الكشف عن الشخص لإنشاء تطبيق مثل عد الأشخاص، تحقق من مقالتنا: كيفية إنشاء نظام عد الأشخاص مع التعلم العميق.

### 3- أهمية اكتشاف الكائنات:

اكتشاف الكائن هو أحد المشاكل الأساسية للرؤية الحاسوبية. إنه يشكل أساس العديد من مهام الرؤية الحاسوبية الأخرى، على سبيل المثال، التجزئة، والتعليق على الصورة، وتتبع الكائن، والمزيد. تشمل تطبيقات الكشف عن الأجسام المحددة اكتشاف المشاة، وعد الأشخاص، وكشف الوجه، وكشف النصوص، واكتشاف الوضع، أو التعرف على لوحة الأرقام.

### 4- كشف الكائنات والتعلم العميق:

في السنوات القليلة الماضية، أدى التقدم السريع لتقنيات التعلم العميق إلى تسريع زخم اكتشاف الأجسام بشكل كبير. مع شبكات التعلم العميق والقوة الحاسوبية لوحدة معالجة الرسومات، تحسن أداء أجهزة الكشف عن الكائنات وأجهزة التتبع بشكل كبير، مما حقق اختراقات كبيرة في اكتشاف الكائنات.

التعلم الآلي (ML Machine learning) هو فرع من فروع الذكاء الاصطناعي (AI artificial intelligence)، ويتضمن بشكل أساسي أنماط التعلم من الأمثلة أو بيانات العينة حيث تصل الآلة إلى البيانات ولديها القدرة على التعلم منها (التعلم الخاضع للإشراف على الصور المشروحة). التعلم العميق هو شكل متخصص من التعلم الآلي يتضمن التعلم في مراحل مختلفة.

## 5- آخر التطورات التكنولوجية:

أصبحت مجموعة واسعة من تطبيقات الرؤية الحاسوبية متاحة لاكتشاف الكائنات وتتبعها. نتيجة لذلك، تعتمد العديد من تطبيقات العالم الحقيقي، مثل مراقبة الرعاية الصحية، والقيادة الذاتية، والمراقبة بالفيديو، والكشف عن الشذوذ، أو رؤية الروبوت، على اكتشاف كائن التعلم العميق.

تقدمت تقنية التصوير بشكل كبير في السنوات الأخيرة. الكاميرات أصغر وأرخص وذات جودة أعلى من أي وقت مضى. وفي الوقت نفسه، زادت قوة الحوسبة بشكل كبير وأصبحت أكثر كفاءة. في السنوات الماضية، تحركت منصات الحوسبة نحو التوازي من خلال المعالجة متعددة النواة، ووحدة المعالجة الرسومية (GPU graphical processing unit)، ومسرعات الذكاء الاصطناعي مثل وحدات معالجة الموتر (tensor processing units TPU).

تسمح هذه الأجهزة بإجراء رؤية حاسوبية لاكتشاف الكائنات وتتبعها في التطبيقات في الوقت الفعلي تقريبًا. ومن ثم، فإن التطور السريع في الشبكات العصبية التلافيفية العميقة (convolutional CNN neural networks) وقوة الحوسبة المعززة لوحدة معالجة الرسومات هي المحركات الرئيسية وراء التقدم الكبير في اكتشاف الكائن القائم على الرؤية الحاسوبية.

## 6- كيف يعمل اكتشاف الكائن:

يمكن إجراء الكشف عن الكائنات إما باستخدام تقنيات معالجة الصور التقليدية (1) أو شبكات التعلم العميق الحديثة (2).

1. (Image processing techniques) لا تتطلب تقنيات معالجة الصور عمومًا بيانات تاريخية

للتدريب وهي غير خاضعة للإشراف بطبيعتها.

• Pro's: وبالتالي، لا تتطلب هذه المهام صورًا مشروحة، حيث قام البشر بتسمية البيانات

يدويًا (للتدريب تحت الإشراف).

• Con's: تقتصر هذه التقنيات على عوامل متعددة، مثل السيناريوهات المعقدة (بدون خلفية

وحيدة اللون)، والانسداد (الأشياء المخفية جزئيًا)، والإضاءة والظلال، وتأثير الفوضى.

## 2. (Deep Learning methods) تعتمد أساليب التعلم العميق بشكل عام على التدريب

الخاضع للإشراف. الأداء محدود بسبب قوة حساب وحدات معالجة الرسومات التي تتزايد بسرعة عامًا بعد عام.

- Pro's: اكتشاف كائن التعلم العميق أكثر قوة بشكل ملحوظ في الانسداد والمشاهد المعقدة والإضاءة الصعبة.

كون: هناك حاجة إلى قدر هائل من بيانات التدريب، عملية شرح الصورة كثيفة العمالة ومكلفة. على سبيل المثال، يعتبر وضع العلامات 500 "الصور 000 لتدريب خوارزمية اكتشاف كائن DL مخصصة مجموعة بيانات صغيرة. ومع ذلك، فإن العديد من مجموعات البيانات القياسية ( MS COCO و Caltech و KITTI و PASCAL VOC و V5) توفر توافر البيانات المحددة.

اليوم، يتم قبول اكتشاف كائن التعلم العميق على نطاق واسع من قبل الباحثين وتبنته شركات الرؤية الحاسوبية لبناء منتجات تجارية.



الشكل (2) - اكتشاف الأجسام القائمة على التعلم العميق للمركبات (السيارات والشاحنات والدراجات وما إلى ذلك). إطار مثالي لتطبيق تجاري في الوقت الفعلي مع التعرف على الذكاء الاصطناعي على تدفق كاميرات IP، المبنية على Viso Suite.



## 7- معالم في أحدث اكتشاف للكائن:

إن مجال اكتشاف الأجسام ليس جديدًا كما قد يبدو. في الواقع، تطور اكتشاف الجسم خلال السنوات 20 الماضية. عادة ما يتم فصل تقدم اكتشاف الأجسام إلى فترتين تاريخيتين منفصلتين (قبل وبعد إدخال التعلم العميق):

قبل عام 2014 - فترة اكتشاف الأجسام التقليدية:

- Viola-Jones Detector (2001)، العمل الرائد الذي بدأ في تطوير طرق اكتشاف الأجسام التقليدية
- HOG Detector (2006)، واصف ميزة شائع لاكتشاف الكائنات في الرؤية الحاسوبية ومعالجة الصور
- DPM (2008) مع أول إدخال لانحدار صندوق الحدود

بعد عام 2014 - فترة اكتشاف التعلم العميق:

- أهم خوارزميات اكتشاف الأجسام ذات المرحلتين:
  1. RCNN و SPPNet (2014)
  2. Fast RCNN و Faster RCNN (2015)
  3. قناع R-CNN (2017)
  4. شبكات الهرم/FPN (2017)
  5. G-RCNN (2021)
- أهم خوارزميات اكتشاف الكائن من مرحلة واحدة:
  1. YOLO (2016)
  2. SSD (2016)
  3. RetinaNet (2017)
  4. YoloV3 (2018)
  5. YoloV4 (2020)
  6. YOLOR (2021)

ستحدث في مشرعنا عن YoloV3 (2018).

## 8-حالات وتطبيقات اكتشاف الأجسام:

حالات الاستخدام التي تتطوي على كشف الأجسام متنوعة جداً، هناك طرق غير محدودة تقريباً لجعل أجهزة الكمبيوتر ترى مثل البشر لأتمتة المهام اليدوية أو إنشاء منتجات وخدمات جديدة تعمل بالذكاء الاصطناعي. تم تنفيذه في برامج الرؤية الحاسوبية المستخدمة لمجموعة من التطبيقات، من الإنتاج الرياضي إلى تحليلات الإنتاجية. للعثور على قائمة شاملة من تطبيقات الرؤية الحاسوبية الحديثة، أوصيك بالتحقق من مقالتنا حول 56 تطبيقاً للرؤية الحاسوبية الأكثر شيوعاً في عام 2022.

اليوم، يعد التعرف على الكائن جوهر معظم برامج وبرامج الذكاء الاصطناعي القائمة على الرؤية. يلعب اكتشاف الكائنات دوراً مهماً في فهم المشهد، وهو أمر شائع في حالات الأمن والنقل والاستخدام الطبي والعسكري.

- **كشف الأجسام في البيع بالتجزئة:** يتم استخدام الأشخاص الذين تم وضعهم بشكل استراتيجي في عد الأنظمة في جميع متاجر البيع بالتجزئة المتعددة لجمع المعلومات حول كيفية قضاء العملاء لوقتهم وإقبال العملاء. يساعد تحليل العملاء المستند إلى الذكاء الاصطناعي لاكتشاف العملاء وتتبعهم باستخدام الكاميرات على اكتساب فهم لتفاعل العملاء وتجربة العملاء، وتحسين تخطيط المتجر، وجعل العمليات أكثر كفاءة. حالة الاستخدام الشائعة هي اكتشاف قوائم الانتظار لتقليل وقت الانتظار في متاجر البيع بالتجزئة.

- **القيادة الذاتية:** تعتمد السيارات ذاتية القيادة على اكتشاف الجسم للتعرف على المشاة وعلامات المرور والمركبات الأخرى والمزيد. على سبيل المثال، يستخدم الطيار الآلي AI من Tesla بشكل كبير اكتشاف الأشياء لإدراك التهديدات البيئية والمحيطية مثل المركبات أو العقبات القادمة.

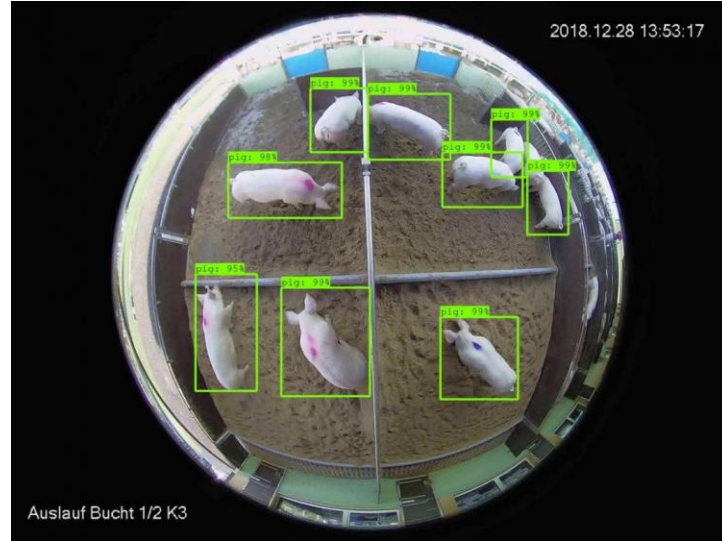
- **الكشف عن الحيوانات في الزراعة:** يستخدم الكشف عن الأجسام في الزراعة لمهام مثل العد ومراقبة الحيوانات وتقييم جودة المنتجات الزراعية. يمكن اكتشاف المنتجات التالفة أثناء معالجتها باستخدام خوارزميات التعلم الآلي.

- **كشف الناس في الأمن:** تعتمد مجموعة واسعة من التطبيقات الأمنية في المراقبة بالفيديو على اكتشاف الأجسام، على سبيل المثال، للكشف عن الأشخاص في المناطق المحظورة أو الخطرة، أو منع الانتحار، أو أتمتة مهام التفتيش في المواقع النائية ذات الرؤية الحاسوبية.



الشكل (3) - مثال على اكتشاف الأجسام في تحليلات الفيديو لاكتشاف الأشخاص في المناطق الخطرة باستخدام كاميرات (CCTV)

- **كشف المركبات مع الذكاء الاصطناعي في النقل:** يُستخدم التعرف على الجسم لاكتشاف وحساب المركبات لتحليل حركة المرور أو للكشف عن السيارات التي تتوقف في المناطق الخطرة، على سبيل المثال، على مفترق الطرق أو الطرق السريعة.
- **الكشف عن الميزات الطبية في الرعاية الصحية:** سمح اكتشاف الأجسام بالعديد من الاختراقات في المجتمع الطبي. نظرًا لأن التشخيصات الطبية تعتمد بشكل كبير على دراسة الصور والفحوصات والصور، فقد أصبح اكتشاف الكائنات الذي يتضمن التصوير المقطعي المحوسب والتصوير بالرنين المغناطيسي مفيدًا للغاية لتشخيص الأمراض، على سبيل المثال باستخدام خوارزميات ML للكشف عن الورم.



الشكل (4) - تطبيق التعلم العميق التجاري لاكتشاف الكائنات في مراقبة الحيوانات، مبني على جناح Viso

### YOLO-9 (You Only Look Once):

كنظام اكتشاف الأجسام في الوقت الفعلي، يستخدم اكتشاف أجسام YOLO شبكة عصبية واحدة. يدعم الإصدار الأخير من ImageAI v2.1.0 الآن تدريب نموذج YOLO مخصص لاكتشاف أي نوع وعدد من الأشياء. الشبكات العصبية التلافيفية هي حالات من الأنظمة القائمة على التصنيف حيث يقوم النظام بإعادة استخدام المصنفات أو الموضعات لإجراء الكشف وتطبيق نموذج الكشف على صورة في مواقع ومقاييس متعددة. باستخدام هذه العملية، تعتبر مناطق الصورة «عالية الدرجات» اكتشافات. ببساطة، يتم تحديد المناطق التي تشبه إلى حد كبير صور التدريب المقدمة بشكل إيجابي.

ككشف من مرحلة واحدة، يقوم YOLO بإجراء التصنيف وانحدار صندوق الحدود في خطوة واحدة، مما يجعله أسرع بكثير من معظم الشبكات العصبية التلافيفية. على سبيل المثال، اكتشاف جسم YOLO أسرع بأكثر من 1000 × من R-CNN و 100 × أسرع من Fast R-CNN.

يحقق mAP%YOLOv3 57.9 على مجموعة بيانات MS COCO مقارنة DSSD513 53.3% و YOLOv3/RetinaNet 61.1. يستخدم تصنيفًا متعدد العلامات مع أنماط متداخلة للتدريب. ومن ثم يمكن استخدامه في السيناريوهات المعقدة لكشف الجسم. بسبب قدرات التنبؤ

متعددة الفئات، يمكن استخدام YOLOv3 لتصنيف الأجسام الصغيرة بينما يظهر أداء أسوأ للكشف عن الأجسام الكبيرة أو المتوسطة الحجم. اقرأ المزيد عن YOLOv3 هنا.

YOLOv4 نسخة محسنة من YOLOv3. الابتكارات الرئيسية هي تحسين بيانات الفسيفساء، والتدريب على الخصومة الذاتية، وتطبيع الدفعة الصغيرة المتقاطعة.

## 10- اكتشاف الكائنات باستخدام إطار YOLO :

يتعامل إطار YOLO (You Only Look Once) مع اكتشاف الكائن بطريقة مختلفة. يأخذ الصورة بأكملها في حالة واحدة ويتنبأ بإحداثيات صندوق الحدود واحتمالات الفئة لهذه الصناديق. أكبر ميزة لاستخدام YOLO هي سرعتها الرائعة - فهي سريعة بشكل لا يصدق ويمكنها معالجة 45 إطارًا في الثانية. يفهم YOLO أيضًا تمثيل الكائن المعمم. هذه واحدة من أفضل الخوارزميات لاكتشاف الكائن سنتعرف على التقنيات المختلفة المستخدمة في خوارزمية YOLO.

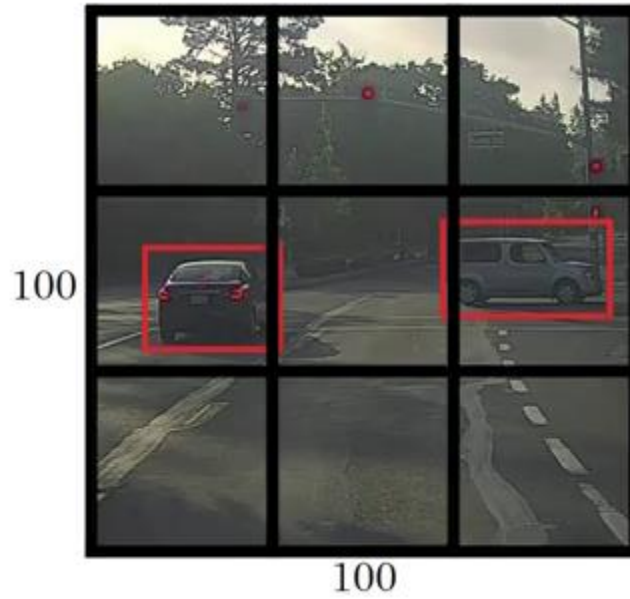
## 11- الية عمل إطار YOLO:

الآن بعد أن فهمنا سبب كون YOLO إطارًا مفيدًا، دعونا نقفز إلى كيفية عمله بالفعل. في هذا القسم، ذكرت الخطوات التي اتبعتها YOLO لكشف الأشياء في صورة معينة.

- يأخذ YOLO أولاً صورة إدخال:



- ثم يقسم الإطار صورة الإدخال إلى شبكات (لنقل شبكة 3 3 X):



- يتم تطبيق تصنيف الصور وتوطينها على كل شبكة. ثم يتنبأ YOLO بصناديق الحدود واحتمالات الفئة المقابلة للأشياء (إذا تم العثور على أي منها بالطبع).

نحن بحاجة إلى تمرير البيانات الملصقة إلى النموذج من أجل تدريبه. لنفترض أننا قسمنا الصورة إلى شبكة مقاس 3 3 X وهناك ما مجموعه 3 فئات نريد تصنيف الكائنات إليها. لنفترض أن

الفصول هي المشاة والسيارة والدراجات النارية على التوالي. لذلك، لكل خلية شبكية، سيكون الملصق  $y$  متجه ثمانية أبعاد:

$y =$	pc
	bx
	by
	bh
	bw
	c1
	c2
	c3

هنا، (Pc) يحدد ما إذا كان الجسم موجودًا في الشبكة أم لا (إنه الاحتمال).

$c1$ ،  $c2$ ،  $c3$  تمثل الفئات. لذا، إذا كان الجسم سيارة،  $c2$  سيكون 1 و  $c1$  و  $c3$  سيكون 0، وهكذا.

لنفترض أننا نختار الشبكة الأولى من المثال أعلاه:



نظرًا لعدم وجود كائن في هذه الشبكة، سيكون الكمبيوتر صفرًا وستكون ملصق  $y$  لهذه الشبكة:

$y =$	0
	?
	?
	?
	?
	?
	?
	?

هنا، «؟» يعني أنه لا يهم ما يحتويه  $bx$ ،  $bh$ ،  $bw$ ،  $c1$ ،  $c2$ ، و  $c3$  حيث لا يوجد كائن في الشبكة. لنأخذ شبكة أخرى لدينا فيها سيارة ( $c2 = 1$ ):



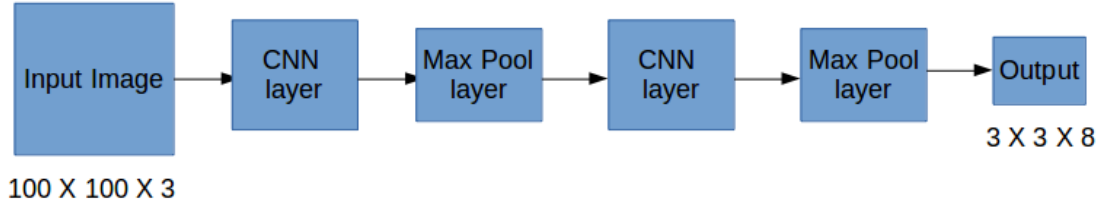
قبل أن نكتب ملصق  $y$  لهذه الشبكة، من المهم أن نفهم أولاً كيف تقرر YOLO ما إذا كان هناك بالفعل كائن في الشبكة. في الصورة أعلاه، هناك جسمان (سيارتان)، لذلك ستأخذ YOLO نقطة الوسط لهذين الجسمين وسيتم تعيين هذه الأجسام في الشبكة التي تحتوي على نقطة الوسط لهذه الأجسام. ملصق  $y$  للشبكة اليسرى الوسطى مع السيارة سيكون:

$y =$	1
	$bx$
	$by$
	$bh$
	$bw$
	0
	1
	0

نظرًا لوجود كائن في هذه الشبكة، فإن الكمبيوتر الشخصي سيكون مساويًا لـ 1.  $bx$ ،  $bh$ ،  $bw$  سيتم حسابها بالنسبة لخلية الشبكة المعينة التي نتعامل معها. بما أن السيارة هي الدرجة الثانية،  $c2 = 1$  و  $c1 = 0$  و  $c3 = 0$ . لذلك، لكل من الشبكات 9، سيكون لدينا متجه خرج ثماني الأبعاد. سيكون لهذا المخرج شكل  $3 \times 8 \times 3$ .



إذاً لدينا الآن صورة مدخلة وهي متجه الهدف المقابل باستخدام المثال أعلاه (صورة الإدخال -  $100 \times 100 \times 3$ ، المخرج -  $3 \times 3 \times 8$ )، سيتم تدريب نموذجنا على النحو التالي:



سنقوم بالانتشار للأمام والخلف لتدريب نموذجنا. خلال مرحلة الاختبار، نقوم بتمثيل صورة إلى النموذج وتشغيل الانتشار إلى الأمام حتى نحصل على مخرج  $y$ . من أجل إبقاء الأمور بسيطة، لقد شرحت ذلك باستخدام شبكة  $3 \times 3 \times 3$  هنا، ولكن بشكل عام في سيناريوهات العالم الحقيقي، نأخذ شبكات أكبر (ربما  $19 \times 19 \times 3$ ).

حتى لو امتد الجسم إلى أكثر من شبكة واحدة، فسيتم تخصيصه فقط لشبكة واحدة تقع فيها نقطة الوسط. يمكننا تقليل فرص ظهور أجسام متعددة في نفس خلية الشبكة عن طريق زيادة عدد الشبكات ( $19 \times 19 \times 3$ ، على سبيل المثال).

## 12- الية تشفير صناديق التثبيت:

كما ذكرنا سابقاً، يتم حساب  $bx$  و  $bh$  و  $bw$  بالنسبة للخلية الشبكية التي نتعامل معها. دعونا نفهم هذا المفهوم مع مثال. ضع في اعتبارك الشبكة اليمنى الوسطى التي تحتوي على سيارة

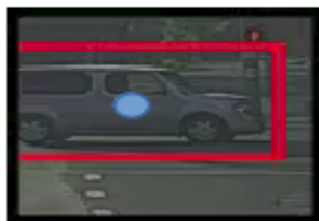


لذا،  $bx$ ،  $by$ ،  $bh$ ،  $bw$  سيتم حسابها بالنسبة لهذه الشبكة فقط. ملصق  $y$  لهذه الشبكة سيكون:

$y =$	1
	$bx$
	$by$
	$bh$
	$bw$
	0
	1
	0

$pc = 1$  نظرًا لوجود كائن في هذه الشبكة وبما أنها سيارة،  $c2 = 1$ . الآن، دعونا نرى كيف نقرر  $bx$ ،  $be$ ،  $bh$ ، و  $bw$ . في YOLO، الإحداثيات المخصصة لجميع الشبكات هي:

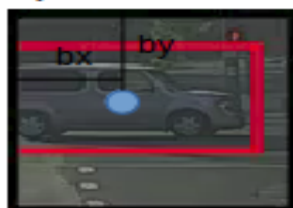
$(0,0)$



$(1,1)$

$bx$ ،  $by$  إحداثيات  $x$  و  $y$  لنقطة منتصف الجسم فيما يتعلق بهذه الشبكة. في هذه الحالة، سيكون (حوالي)  $bx = 0.4$  و  $by = 0.3$ :

$(0,0)$



$(1,1)$

bh هي نسبة ارتفاع الصندوق الحدودي (الصندوق الأحمر في المثال أعلاه) إلى ارتفاع خلية الشبكة المقابلة، والتي تبلغ في حالتنا حوالي 0.9. إذاً  $bh = 0.9 bw$  هي نسبة عرض صندوق الحدود إلى عرض خلية الشبكة. لذا،  $bw = 0.5$  (تقريباً). ملصق y لهذه الشبكة سيكون:

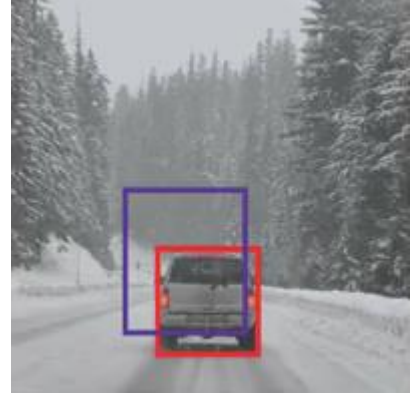
y =	1
	0.4
	0.3
	0.9
	0.5
	0
	1
	0

لاحظ هنا أن  $bx$  و  $by$  سيتراوح دائماً بين 0 و 1 لأن نقطة الوسط ستقع دائماً داخل الشبكة. في حين أن  $bh$  و  $bw$  يمكن أن يكونا أكثر من 1 في حال كانت أبعاد صندوق الحدود أكثر من بعد الشبكة. في القسم التالي، سننظر في المزيد من الأفكار التي يمكن أن تساعدنا في تحسين أداء هذه الخوارزمية.

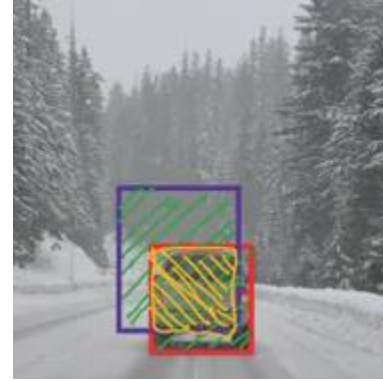
## Intersection over Union and Non-Max Suppression

تقاطع فوق قمع الاتحاد وغير ماكس

كيف يمكننا أن نقرر ما إذا كان المربع الحدودي المتوقع يمنحنا نتيجة جيدة (أم سيئة) ؟ هذا هو المكان الذي يأتي فيه التقاطع فوق الاتحاد في الصورة. يحسب التقاطع على اتحاد صندوق الحد الفعلي وصندوق الترابط المتوقع. ضع في اعتبارك صناديق الحدود الفعلية والمتوقعة للسيارة كما هو موضح أدناه:



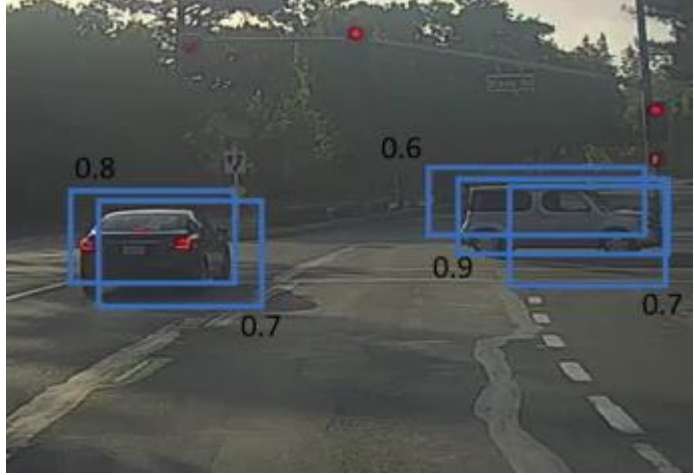
هنا، الصندوق الأحمر هو صندوق الحدود الفعلي والصندوق الأزرق هو الصندوق المتوقع. كيف يمكننا أن نقرر ما إذا كان تنبؤًا جيدًا أم لا ؟ ستحسب IoU، أو التقاطع فوق الاتحاد، مساحة التقاطع فوق اتحاد هذين الصندوقين. ستكون تلك المنطقة:



$IoU = \text{منطقة تقاطع/منطقة الاتحاد}$ .

$IoU = \text{منطقة الصندوق الأصفر/منطقة الصندوق الأخضر}$ .

إذا كان IoU أكبر من 0.5، فيمكننا القول إن التنبؤ جيد بما فيه الكفاية. 0.5 هي عتبة تعسفية اتخذناها هنا، ولكن يمكن تغييرها وفقًا لمشكلتك المحددة. بشكل حدسي، كلما زادت العتبة، أصبحت التنبؤات أفضل، هناك تقنية أخرى يمكنها تحسين إنتاج YOLO بشكل كبير - قمع Non-Max. واحدة من أكثر المشاكل شيوعًا في خوارزميات اكتشاف الجسم هي أنه بدلاً من اكتشاف جسم مرة واحدة فقط، قد يكتشفونه عدة مرات. كما هو موضح أدناه:



هنا، يتم التعرف على السيارات أكثر من مرة. تقنية القمع Non-Max تتنظف هذا حتى نحصل على اكتشاف واحد فقط لكل كائن. دعونا نرى كيف يعمل هذا النهج.

(1) يبحث أولاً في الاحتمالات المرتبطة بكل اكتشاف ويأخذ أكبر واحدة. في الصورة أعلاه، 0.9 هو أعلى احتمال، لذلك سيتم اختيار المربع مع 0.9 احتمال أولاً:



الآن، ينظر إلى جميع الصناديق الأخرى في الصورة. يتم قمع الصناديق التي تحتوي على IoU عالية مع الصندوق الحالي. لذلك، سيتم قمع الصناديق ذات الاحتمالات 0.6 و 0.7 في مثالنا:



(2) بعد قمع الصناديق، يختار الصندوق التالي من جميع الصناديق بأعلى احتمال، وهو 0.8 في حالتنا



(3) مرة أخرى، سينظر إلى IoU لهذا الصندوق مع الصناديق المتبقية ويضغط على الصناديق باستخدام IoU مرتفع:



(4) نكرر هذه الخطوات حتى يتم اختيار جميع الصناديق أو ضغطها ونحصل على صناديق الحدود النهائية:



هذا ما يدور حوله قمع Non-Max. نحن نأخذ الصناديق بأقصى احتمالية ونقمع الصناديق القريبة ذات الاحتمالات غير القصوى. دعونا نلخص بسرعة النقاط التي رأيناها في هذا القسم حول خوارزمية القمع Non-Max:

1) تخلص من جميع الصناديق التي لديها احتمالات أقل من أو تساوي عتبة محددة مسبقاً (على سبيل المثال، 0.5)

2) بالنسبة للصناديق المتبقية:

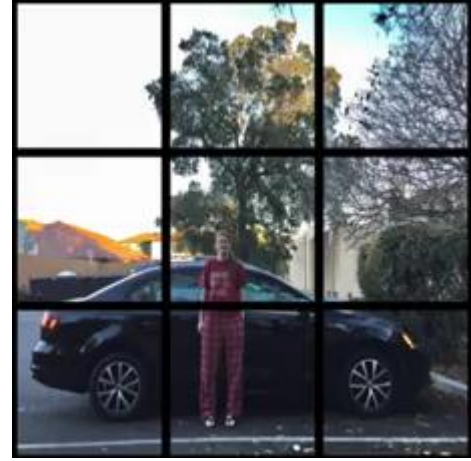
- اختيار الصندوق بأعلى احتمالية واخذ ذلك كتنبؤ بالإخراج.
- التخلص من أي مربع آخر يحتوي على IoU أكبر من العتبة مع مربع الإخراج من الخطوة أعلاه.

3) تكرار الخطوة 2 حتى يتم أخذ جميع الصناديق إما كتنبؤ بالإخراج أو التخلص منها

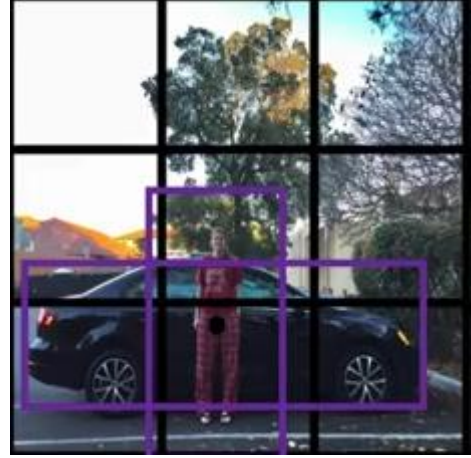
هناك طريقة أخرى يمكننا استخدامها لتحسين أداء خوارزمية YOLO - دعنا نتحقق منها

### 13- صناديق المرساة Anchor Boxes :

لقد رأينا أن كل شبكة يمكنها تحديد كائن واحد فقط. ولكن ماذا لو كانت هناك أشياء متعددة في شبكة واحدة؟ يمكن أن يكون هذا هو الحال في كثير من الأحيان في الواقع. وهذا يقودنا إلى مفهوم صناديق التثبيت. ضع في اعتبارك الصورة التالية، مقسمة إلى شبكة 3 3 X:



تذكر كيف قمنا بتعيين كائن لشبكة ؟ أخذنا نقطة الوسط للجسم وبناءً على موقعه، قمنا بتعيين الجسم في الشبكة المقابلة. في المثال أعلاه، تقع نقطة الوسط لكلا الجسمين في نفس الشبكة. هذه هي الطريقة التي ستكون بها صناديق الحدود الفعلية للأجسام:



سنحصل على أحد الصندوقين فقط، إما للسيارة أو للشخص. ولكن إذا استخدمنا صناديق التثبيت، فقد نتمكن من إخراج كلا الصندوقين! كيف يتم ذلك؟ أولاً، نحدد مسبقاً شكلين مختلفين يسمى صناديق التثبيت أو أشكال صندوق التثبيت. الآن، لكل شبكة، بدلاً من وجود مخرج واحد، سيكون لدينا مخرجان. يمكننا دائماً زيادة عدد صناديق التثبيت أيضاً. لقد أخذت اثنين هنا لجعل المفهوم سهل الفهم:

Anchor box 1:



Anchor box 2:



هكذا تبدو علامة  $y$  بدون صناديق تثبيت:

$y =$	pc
	bx
	by
	bh
	bw
	c1
	c2
	c3



إذا كان لدينا 2 صناديق تثبيت، ملصق  $y$  سيكون:

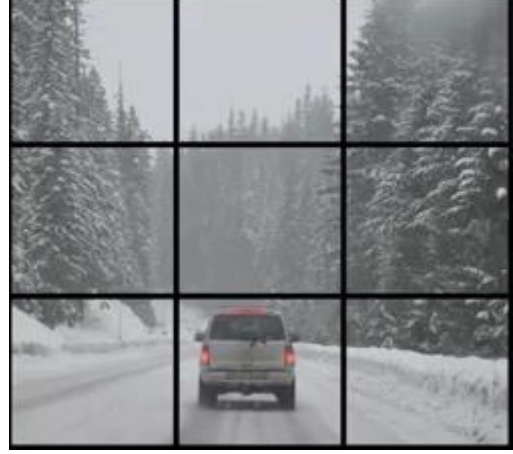
$y =$	pc
	bx
	by
	bh
	bw
	c1
	c2
	c3
	pc
	bx
	by
	bh
	bw
	c1
	c2
	c3

تنتمي صفوف 8 الأولى إلى صندوق المرساة 1 وينتمي 8 المتبقي إلى صندوق المرساة 2. يتم تعيين الكائنات في صناديق التثبيت بناءً على تشابه الصناديق الحدودية وشكل صندوق التثبيت. نظرًا لأن شكل مربع المرساة 1 مشابه لصندوق التثبيت الخاص بالشخص، فسيتم تعيين الأخير في مربع التثبيت 1 وسيتم تعيين السيارة في مربع التثبيت 2. سيكون الناتج في هذه الحالة، بدلاً من  $3 \times 3 \times 8$  (باستخدام شبكة  $3 \times 3$  و 3 فئات)،  $3 \times 3 \times 16$  (نظرًا لأننا نستخدم مرساتين). لذلك، لكل شبكة، يمكننا اكتشاف جسمين أو أكثر بناءً على عدد المراسي. دعونا نجمع بين جميع الأفكار التي غطيناها حتى الآن وندمجها في إطار عمل YOLO.

#### 14- الجمع بين الأفكار:

في هذا القسم، سنرى أولاً كيف يتم تدريب نموذج YOLO ثم كيف يمكن عمل التنبؤات لصورة جديدة وغير مرئية سابقًا.

من الواضح أن مدخلات تدريب نموذجنا ستكون الصور وملصقات  $y$  المقابلة لها. دعونا نرى صورة ونجعل علامة  $y$ :



ضع في اعتبارك السيناريو الذي نستخدم فيه شبكة  $3 \times 3 \times 16$  مع مرساتين لكل شبكة، وهناك 3 فئات مختلفة من الكائنات. لذا فإن ملصقات  $y$  المقابلة سيكون لها شكل  $3 \times 3 \times 16$ . الآن، لنفترض أننا إذا استخدمنا 5 صناديق ترسيخ لكل شبكة وقد تم زيادة عدد الفصول إلى 5. إذاً الهدف سيكون  $3 \times 3 \times 50 = 3 \times 3 \times 10 \times 5$  هذه هي الطريقة التي تتم بها عملية التدريب - التقاط صورة لشكل معين ورسم خرائط له بهدف  $3 \times 3 \times 16$  (قد يتغير هذا وفقاً لحجم الشبكة وعدد صناديق المرساة وعدد الفصول الدراسية).

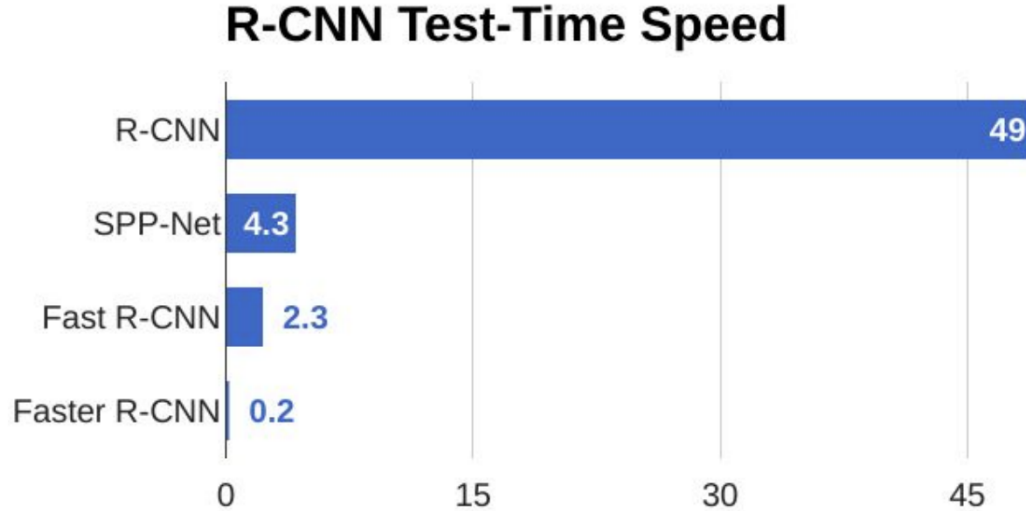
### 15- اختبار:

سيتم تقسيم الصورة الجديدة إلى نفس عدد الشبكات التي اخترناها خلال فترة التدريب. لكل شبكة، سيتنبأ النموذج بإخراج الشكل  $3 \times 3 \times 16$  (بافتراض أن هذا هو شكل الهدف أثناء وقت التدريب). ستكون القيم الـ 16 في هذا التنبؤ بنفس شكل علامة التدريب. تتوافق القيم الـ 8 الأولى مع مربع التثبيت 1، حيث ستكون القيمة الأولى هي احتمال وجود كائن في تلك الشبكة. القيم الـ 2-5 ستكون أحداثيات صندوق الحدود لهذا الكائن، والقيم الثلاث الأخيرة ستخبرنا عن الفئة التي ينتمي إليها الكائن القيم الـ 8 التالية ستكون لصندوق التثبيت 2 وبغض النظر عن الشكل، أي الاحتمال أولاً، ثم أحداثيات صندوق التثبيت، وأخيراً الفئات.

أخيراً، سيتم تطبيق تقنية القمع غير ماكس على الصناديق المتوقعة للحصول على تنبؤ واحد لكل كائن.

يقودنا ذلك إلى نهاية الجانب النظري لفهم كيفية عمل خوارزمية YOLO، بدءًا من تدريب النموذج ثم إنشاء صناديق تنبؤ للأشياء. فيما يلي الأبعاد والخطوات الدقيقة التي تتبعها خوارزمية YOLO:

- يأخذ صورة مدخلة للشكل (3، 608، 608)
- يمرر هذه الصورة إلى شبكة عصبية تلافيفية (CNN)، والتي تعيد (19، 19، 5، 85) بعدًا للخروج
- تم تسوية البعدين الأخيرين للنواتج المذكور أعلاه للحصول على حجم ناتج قدره (19 و 19 و 425):
  - هنا، كل خلية من شبكة 19 19 X ترجع 425 رقم
  - $425 = 5 * 85$ ، حيث 5 هو عدد صناديق المرساة لكل شبكة
  - $85 = 80 + 5$ ، حيث 5 (pc، bx، bh، bw) و 80 هو عدد الفصول التي نريد اكتشافها
- أخيرًا، نقوم بقمع IoU و Non-Max لتجنب اختيار الصناديق المتداخلة



## 16- الكود البرمجي:

تم رفع المشروع على <https://github.com/firas830/yolo-object-detection> : GitHub.com

```
import cv2
import numpy as np
net = cv2.dnn.readNet("yolov3.weights", "yolov3.cfg")
classes = []
with open("coco.names", "r") as f:
    classes = [line.strip() for line in f.readlines()]
layer_names = net.getLayerNames()
output_layers = [layer_names[i-1] for i in
net.getUnconnectedOutLayers()]
colors = np.random.uniform(0, 255, size=(len(classes), 3))
img = cv2.imread("0.jpg")
img = cv2.resize(img, None, fx=0.4, fy=0.4)
height, width, channels = img.shape
blob = cv2.dnn.blobFromImage(img, 0.00392, (416, 416), (0, 0, 0), True,
crop=False)
net.setInput(blob)
outs = net.forward(output_layers)
class_ids = []
confidences = []
boxes = []
for out in outs:
    for detection in out:
        scores = detection[5:]
        class_id = np.argmax(scores)
        confidence = scores[class_id]
        if confidence > 0.5:
            center_x = int(detection[0] * width)
            center_y = int(detection[1] * height)
            w = int(detection[2] * width)
            h = int(detection[3] * height)
            x = int(center_x - w / 2)
            y = int(center_y - h / 2)
            boxes.append([x, y, w, h])
            confidences.append(float(confidence))
            class_ids.append(class_id)
indexes = cv2.dnn.NMSBoxes(boxes, confidences, 0.5, 0.4)
print(indexes)
font = cv2.FONT_HERSHEY_PLAIN
for i in range(len(boxes)):
    if i in indexes:
        x, y, w, h = boxes[i]
        label = str(classes[class_ids[i]])
        color = colors[class_ids[i]]
        cv2.rectangle(img, (x, y), (x + w, y + h), color, 2)
        cv2.putText(img, label, (x, y + 30), font, 3, color, 3)
cv2.imshow("Image", img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

## 17- المراجع:

1- مقرر معالجة الصوت والصورة الرقمية – د.السموول صالح – د. ثناء جبيلي - جامعة تشرين –  
قسم هندسة الاتصالات والالكترونيات-2021.

2-J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once

Pattern Recognit., 2016, .Unified, real-time object detection,” in Proc. IEEE Conf. Comput. Vis  
pp. 779–788

3- A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional  
neural networks,” Proc. Adv. Neural Inf. ProcessSyst., 2012, pp. 1106–1114

4- J. Redmon, and A. Farhadi, “YOLOv3: An incremental improvement

.arXiv:1804.02767. [Online]. Available: <https://arxiv> ,2018

[org/abs/1804.02767](https://arxiv.org/abs/1804.02767)

High Education Ministry

Tishreen University

Mechanical & Electrical Engineering Faculty

**Communication & Electronic Department**

**5th, Network Programming**



**Detection objects using yolo**

**Supervisor**

Dr.Eng. Mohannad Issa

**By**

Firas Barakat

Rasha Asfoor

2022-2021

