

Vision par ordinateur pour la reconnaissance individuelle de girafes

Vincent Miele^{1,2}, Bruno Spataro¹, Lionel Humblot¹,
Dominique Allainé¹, Christophe Bonenfant¹

¹ Laboratoire Biométrie Biologie Evolutive - CNRS - Lyon

² Laboratoire Ecologie Alpine - CNRS - Chambéry

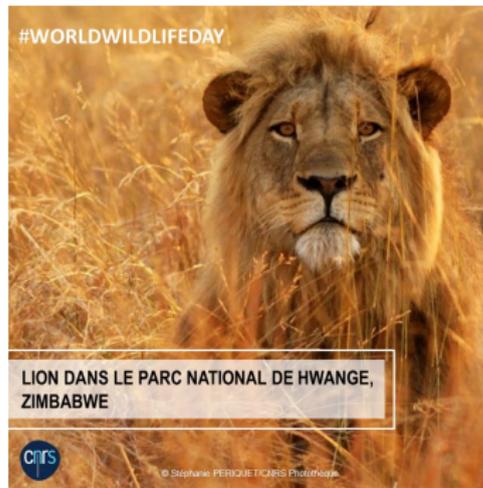
Hwange



LOCALISATION GÉOGRAPHIQUE



Le parc national de Hwange et sa périphérie, au Zimbabwe (Afrique australe)



Hwange - Recherche

- * Ecologie de la conservation
- * Ecologie du comportement
- * **Démographie animale**
- * **Socialité animale**

dans un contexte de changements globaux



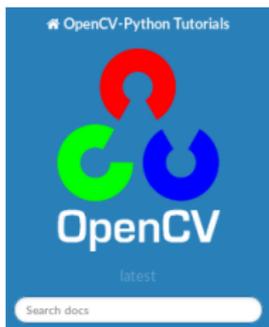
Reconnaissance individuelle de girafe

- * Campagne de photographies pluri-annuelles (saisons sèches et humides)
- * Reconnaissance et suivi des individus au cours du temps



Reconnaissance individuelle de girafe

* SIFT [Lowe (2004)]



Docs » OpenCV-Python Tutorials » Feature Detection and Description »
Introduction to SIFT (Scale-Invariant Feature Transform)

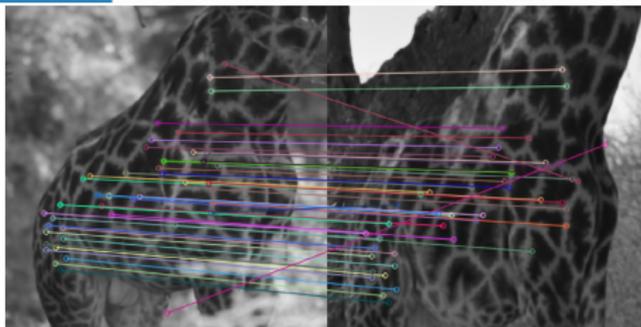
[Edit on GitHub](#)

Introduction to SIFT (Scale-Invariant Feature Transform)

Goal

In this chapter,

- We will learn about the concepts of SIFT algorithm
- We will learn to find SIFT Keypoints and Descriptors.

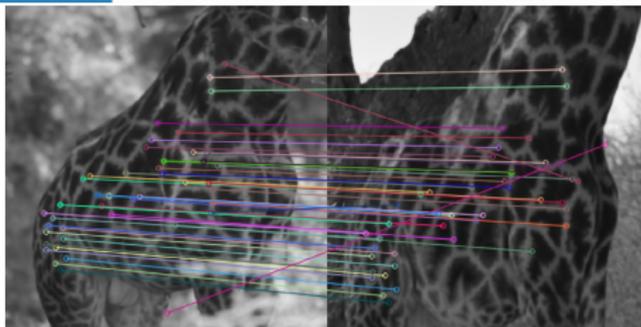


Reconnaissance individuelle de girafe

* SIFT [Lowe (2004)]



The screenshot shows the OpenCV Python Tutorials website. On the left is a blue sidebar with the OpenCV logo (three interlocking circles in red, green, and blue) and the text 'OpenCV latest' and a search bar. The main content area has a breadcrumb trail: 'Docs > OpenCV-Python Tutorials > Feature Detection and Description > Introduction to SIFT (Scale-Invariant Feature Transform)'. There is a link to 'Edit on GitHub'. The title of the page is 'Introduction to SIFT (Scale-Invariant Feature Transform)'. Below the title is the section 'Goal' and a sub-section 'In this chapter,' followed by a bulleted list: '• We will learn about the concepts of SIFT algorithm' and '• We will learn to find SIFT Keypoints and Descriptors.'



.. nécessite l'extraction des flancs de girafes depuis les photos originales

You Only Look Once: Unified, Real-Time Object Detection

Joseph Redmon^{*}, Santosh Divvala^{+†}, Ross Girshick[‡], Ali Farhadi^{+†}

University of Washington^{*}, Allen Institute for AI[†], Facebook AI Research[‡]

<http://pjreddie.com/yolo/>

Abstract

We present YOLO, a new approach to object detection. Prior work on object detection repurposes classifiers to perform detection. Instead, we frame object detection as a regression problem to spatially separated bounding boxes and associated class probabilities. A single neural network predicts bounding boxes and class probabilities directly from full images in one evaluation. Since the whole detection pipeline is a single network, it can be optimized end-to-end directly on detection performance.

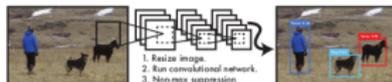


Figure 1: The YOLO Detection System. Processing images with YOLO is simple and straightforward. Our system (1) resizes the input image to 448×448 , (2) runs a single convolutional network on the image, and (3) thresholds the resulting detections by the model's confidence.

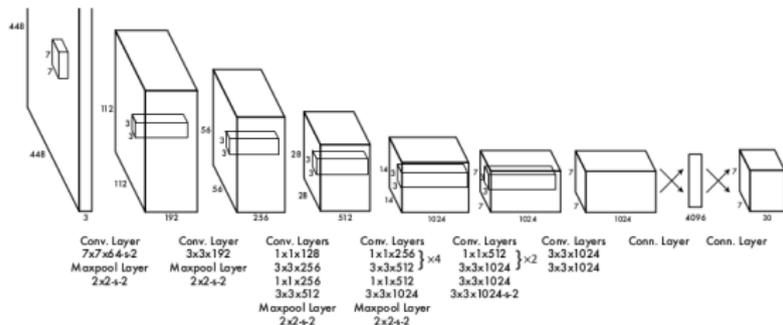


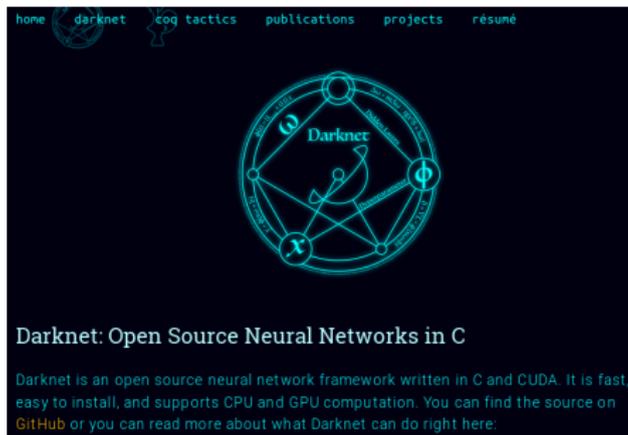
Figure 3: The Architecture. Our detection network has 24 convolutional layers followed by 2 fully connected layers. Alternating 1×1 convolutional layers reduce the features space from preceding layers. We pretrain the convolutional layers on the ImageNet classification task at half the resolution (224×224 input image) and then double the resolution for detection.

YOLO

“Other methods first generate potential bounding boxes in an image and then run a classifier on these proposed boxes”

“We reframe object detection as a single regression problem, straight from image pixels to bounding box coordinates and class probabilities. Using our system, you only look once (YOLO) at an image”

YOLO



C + openMP + Cuda + NVidia cuDNN (+ openCV)

YOLO - Apprentissage

- * 400 images annotées (coordonnées des flancs de girafes)
- * Transfer learning
- * 8 jours avec 32 CPU
- * 80 minutes avec NVIDIA Titan X

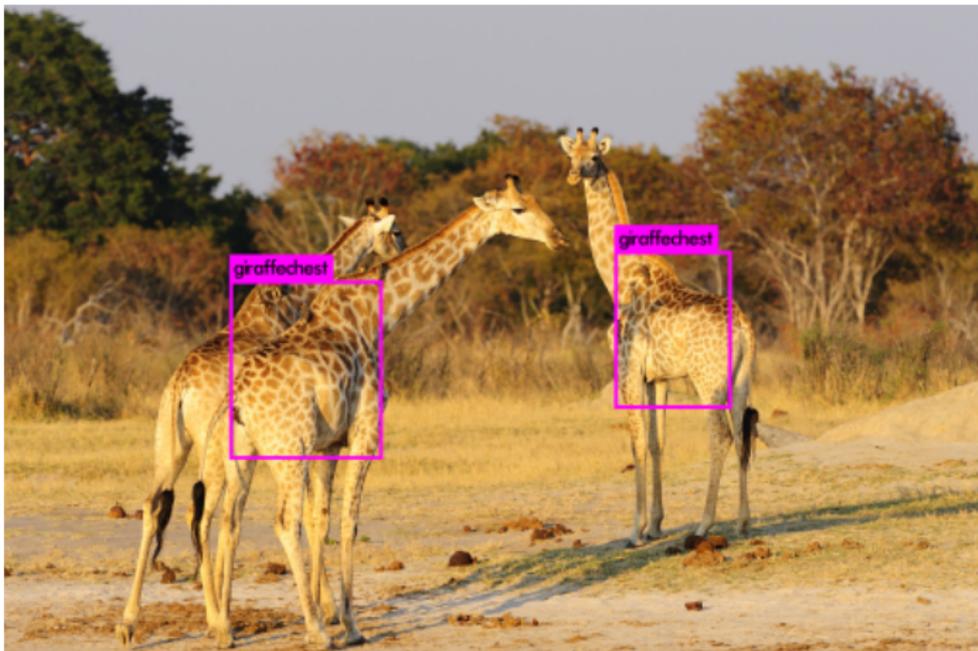
YOLO - Détection

* Quelques secondes par image



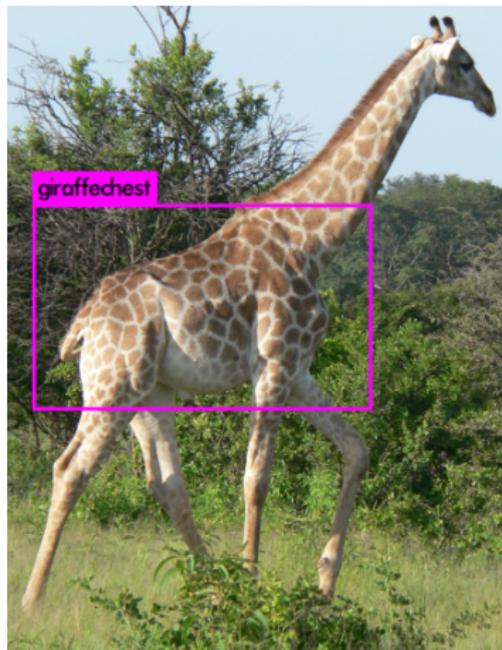
YOLO - Détection

* Quelques secondes par image



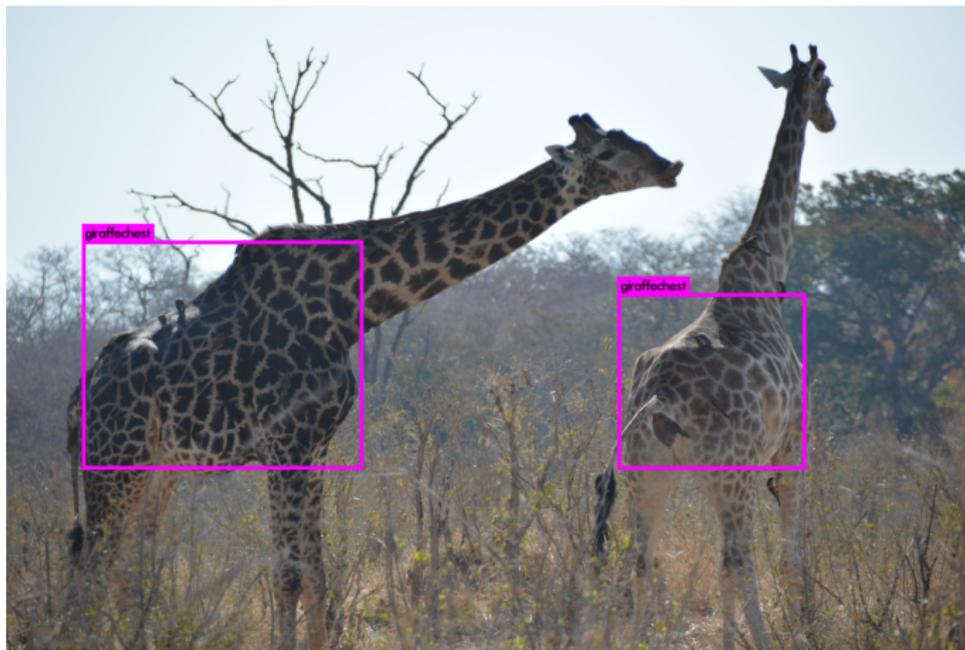
YOLO - Détection

* Quelques secondes par image



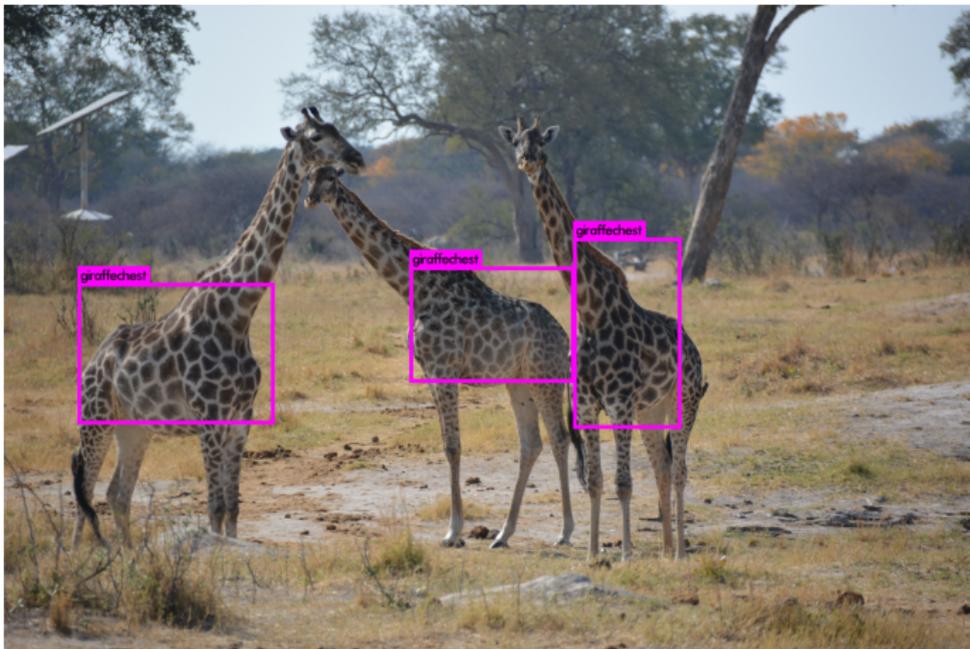
YOLO - Détection

* Quelques secondes par image



YOLO - Détection

* Quelques secondes par image

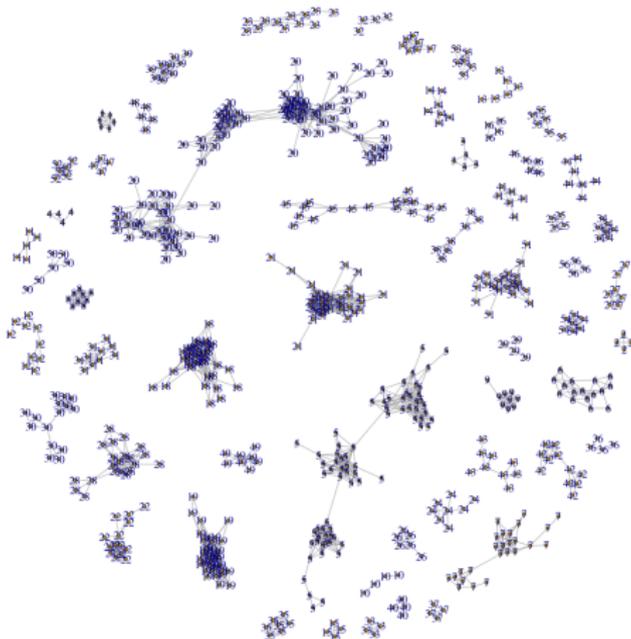


YOLO - Détection

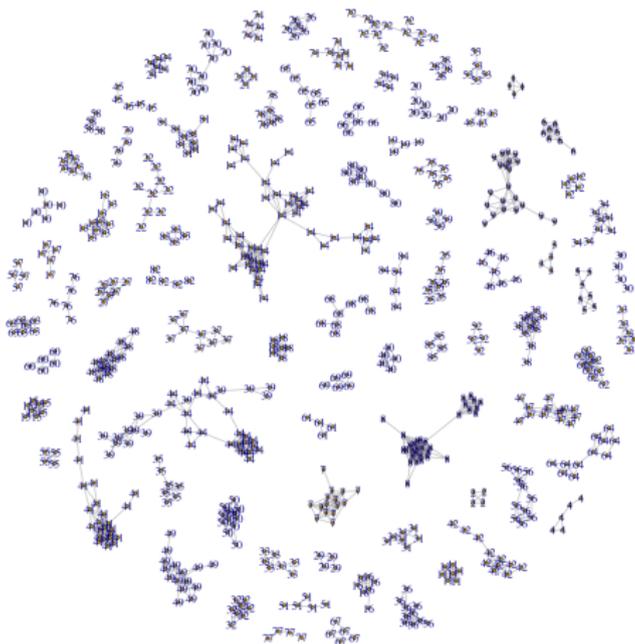
* Quelques secondes par image



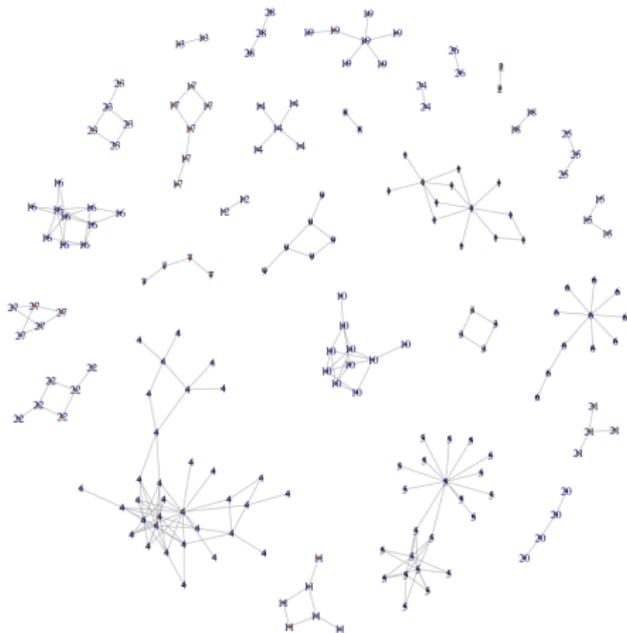
Réseaux de similarité



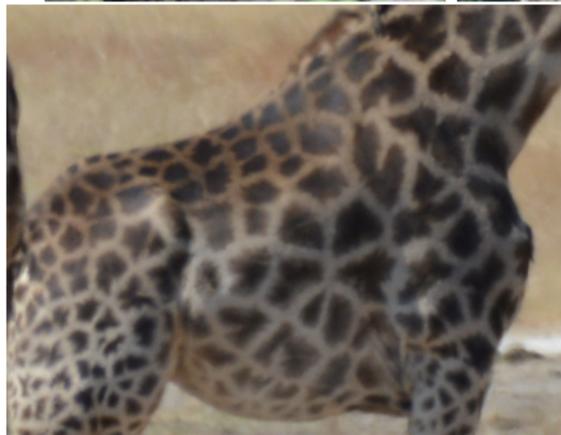
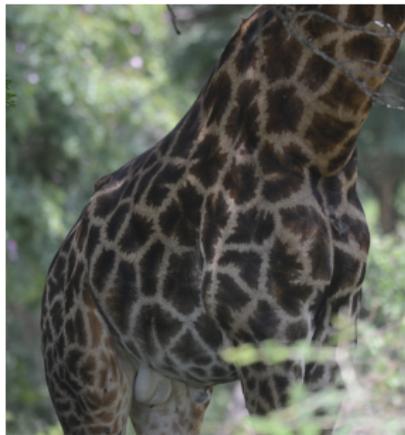
Réseaux de similarité



Réseaux de similarité



Résultats



Et maintenant

* “GPU for Wildlife @ CC” ?

Et maintenant

- * “GPU for Wildlife @ CC” ?
- * GT national “IMage PROcessing en écologie”