

# Understanding the problem statement: COCO keypoints challenge

# COCO keypoint challenge

- **Objective:** To determine person key points from images



# COCO keypoint challenge

- **Objective:** To determine person key points from images



Analytics  
Vidhya

# COCO keypoint challenge

- **Objective:** To determine person key points from images



Ana  
Vid

Pose  
Detector



# COCO keypoint challenge

- **Objective:** To determine person key points from images



Analytics  
Vidhya

# COCO keypoint challenge

- **Objective:** To determine person key points from images



# Understanding the Dataset: COCO keypoint challenge

- 58,945 images



# Understanding the Dataset: COCO keypoint challenge

- 58,945 images
- 156,165 annotated people





# Understanding the Dataset: COCO keypoint challenge

- 58,945 images
- 156,165 annotated people
- 1,710,498 total keypoints



# Understanding the Dataset: COCO keypoint challenge

- 58,945 images
- 156,165 annotated people
- 1,710,498 total keypoints
- 17 types of keypoints

`"nose", "left_eye", "right_eye", "left_ear", "right_ear",  
"left_shoulder", "right_shoulder", "left_elbow", "right_elbow",  
"left_wrist", "right_wrist", "left_hip", "right_hip",  
"left_knee", "right_knee", "left_ankle", "right_ankle"`

# Understanding the Dataset: COCO keypoint challenge

Keypoint: (x, y, v)



# Understanding the Dataset: COCO keypoint challenge

Keypoint: (x, y, v)

- x, y represents the pixel locations



# Understanding the Dataset: COCO keypoint challenge

Keypoint: (x, y, v)

- x, y represents the pixel locations
- v represents the visibility

# Understanding the Dataset: COCO keypoint challenge

Keypoint: (x, y, v)

- x, y represents the pixel locations
- v represents the visibility
  - v=0, not labeled (in which case x=y=0)

# Understanding the Dataset: COCO keypoint challenge

Keypoint: (x, y, v)

- x, y represents the pixel locations
- v represents the visibility
  - v=0, not labeled (in which case x=y=0)
  - v=1, labeled but not visible

# Understanding the Dataset: COCO keypoint challenge

Keypoint: (x, y, v)

- x, y represents the pixel locations
- v represents the visibility
  - v=0, not labeled (in which case x=y=0)
  - v=1, labeled but not visible
  - v=2, labeled and visible



# Understanding the Dataset

[Get Started](#)[Ecosystem](#)[Mobile](#)[Blog](#)[Tutorials](#)[Docs](#)[Resources](#)[GitHub](#)

## Catalyst

Catalyst helps you write compact, but full-featured deep learning and reinforcement learning pipelines with a few lines of code.

## CrypTen

CrypTen is a framework for Privacy Preserving ML. Its goal is to make secure computing techniques accessible to ML practitioners.

## Detectron2

Detectron2 is FAIR's next-generation platform for object detection and segmentation.

## DGL

Deep Graph Library (DGL) is a Python package built for easy implementation of graph neural network model family, on top of PyTorch and other frameworks.

## ELF

ELF is a platform for game research that allows developers to train and test their algorithms in various game environments.



## fastai

fastai is a library that simplifies training fast and accurate neural nets using modern best practices.

# Steps to solve the Pose Detection problem

1. Install Dependencies



# Steps to solve the Pose Detection problem

1. Install Dependencies
2. Load and pre-process the data



# Steps to solve the Pose Detection problem

1. Install Dependencies
2. Load and pre-process the data
3. Loading and training the model



# Steps to solve the Pose Detection problem

1. Install Dependencies
2. Load and pre-process the data
3. Loading and training the model
4. Evaluating model performance



Thank You