

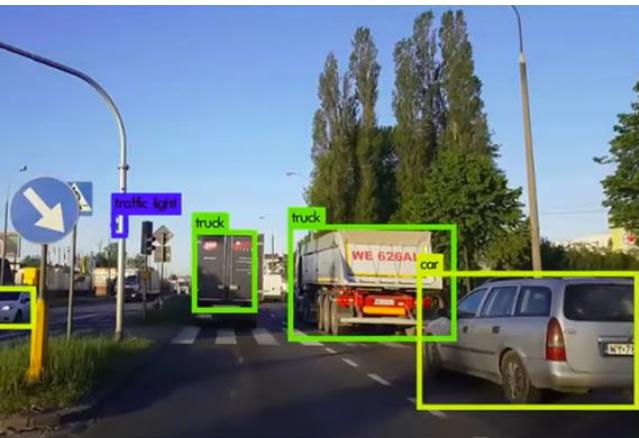
# TensorFlow Object Detection (TFOD)

Tabrej Khan

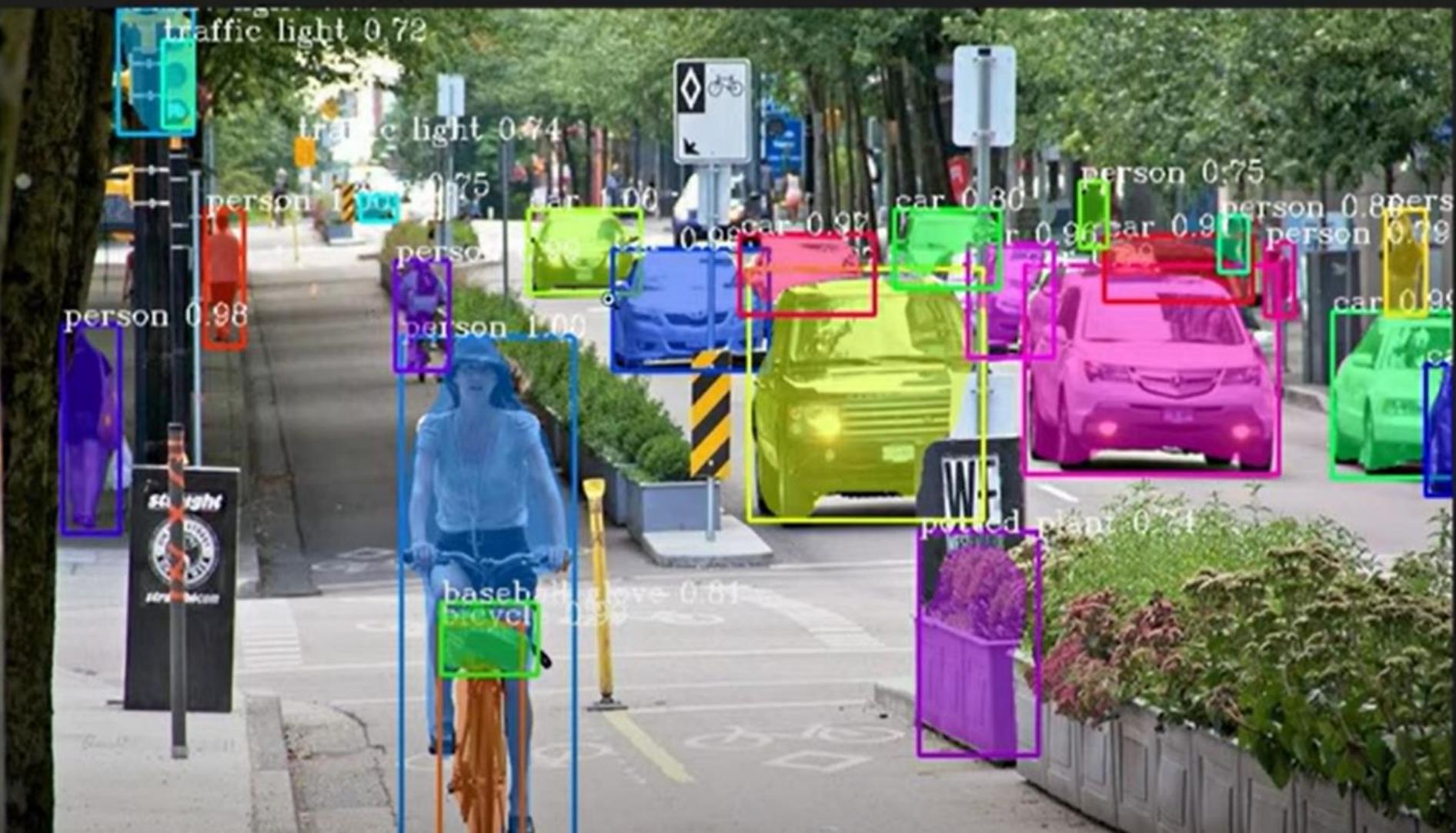
King AbdulAziz University

27<sup>th</sup> March 2022 (12:00-1:00 PM)

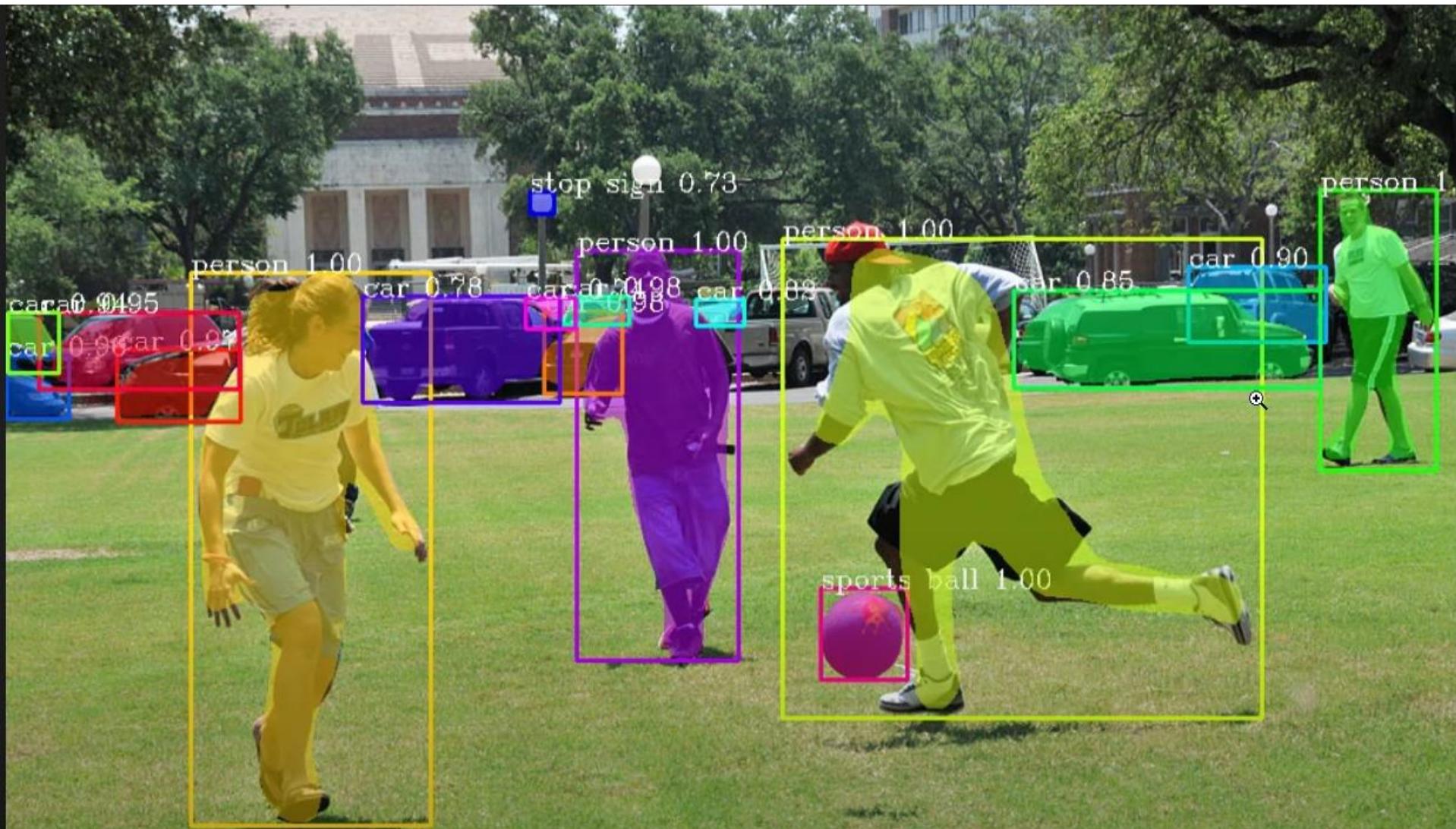
Lab 104



# الدورات العلمية التخصصية ل كلية الحاسوب وتقنيه المعلومات برابغ



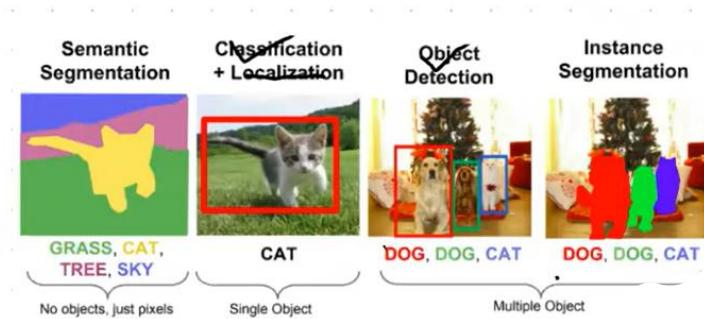
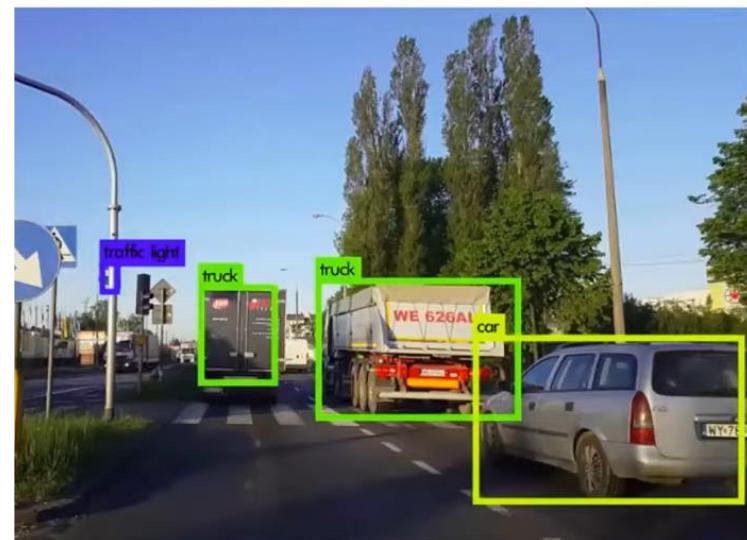
## الدورات العلمية التخصصية ل كلية الحاسوب وتقنية المعلومات برابغ



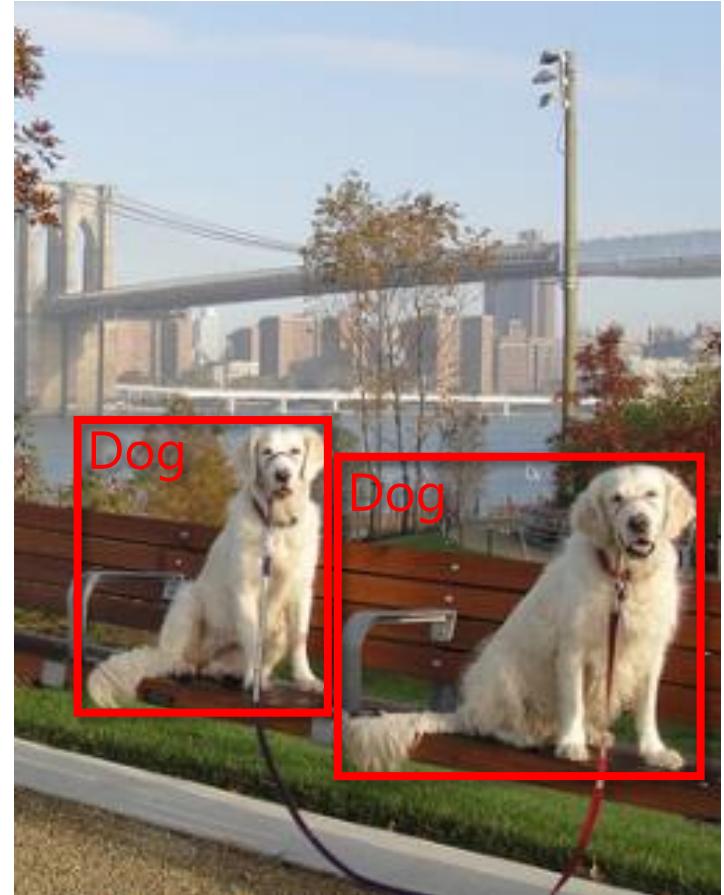
# TFOD

## # Agenda

1. Downloading.
2. Installation tf 1.14
3. Setup Verification
4. Custom training Data
  - 4.1 Annotation/Labelling
  - 4.2 Custom training p  
conversion of xml >>>  
csv >>> tfrecords
  - 4.3 get config file & make cha
  - 4.4 create labelmap.pbtxt
- 4.5 Start training
- 4.6 Stop/resume training
- 4.7 convert ckpt to pb



# Classification vs. Detection

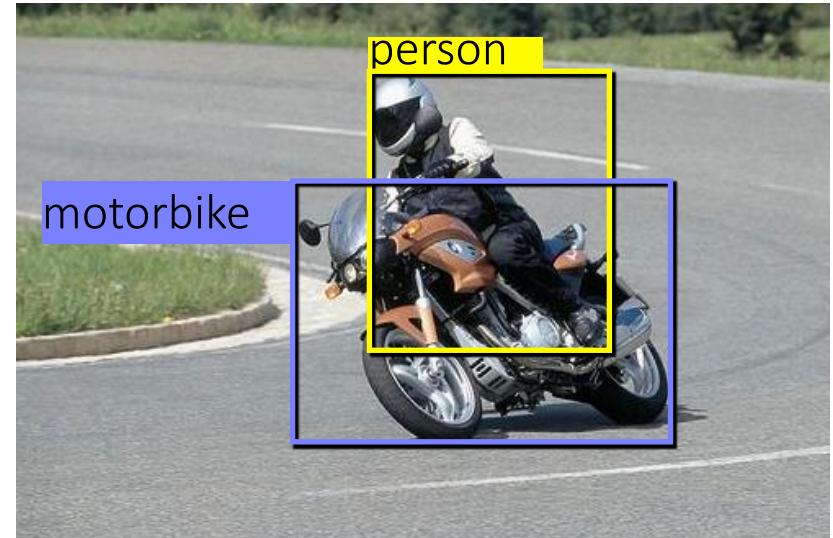


# Problem formulation

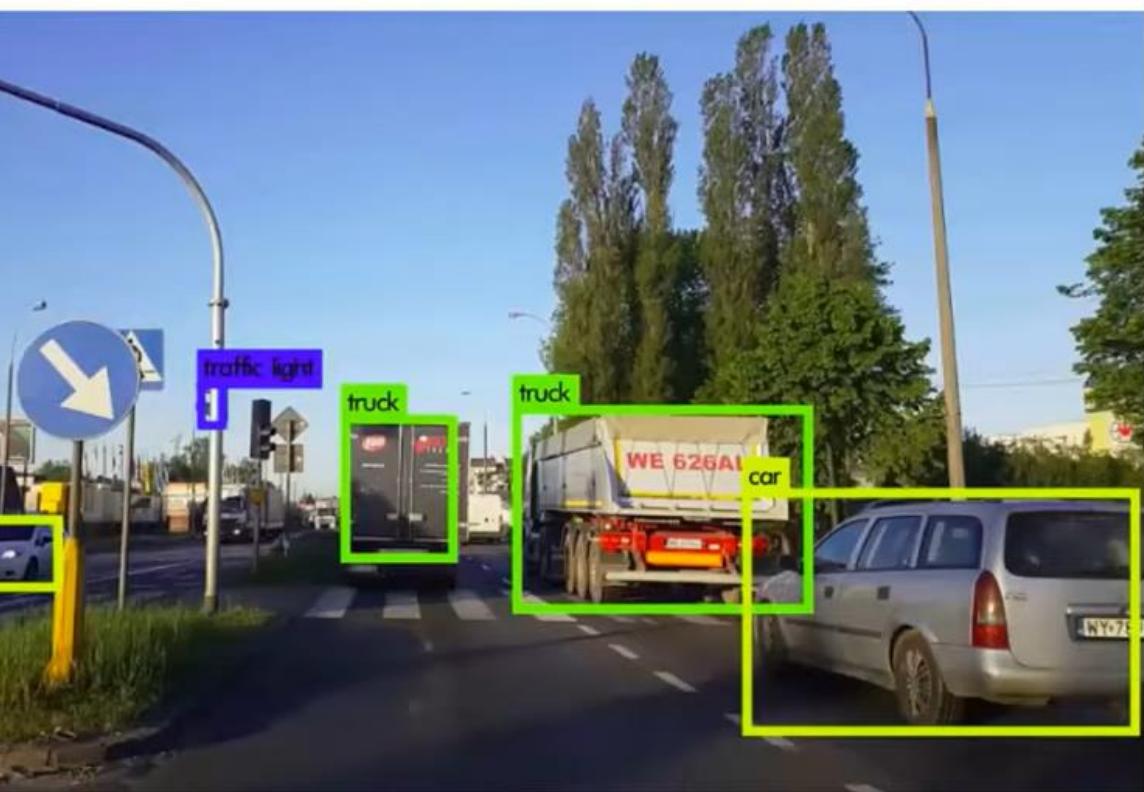
{ airplane, bird, motorbike, person, sofa }



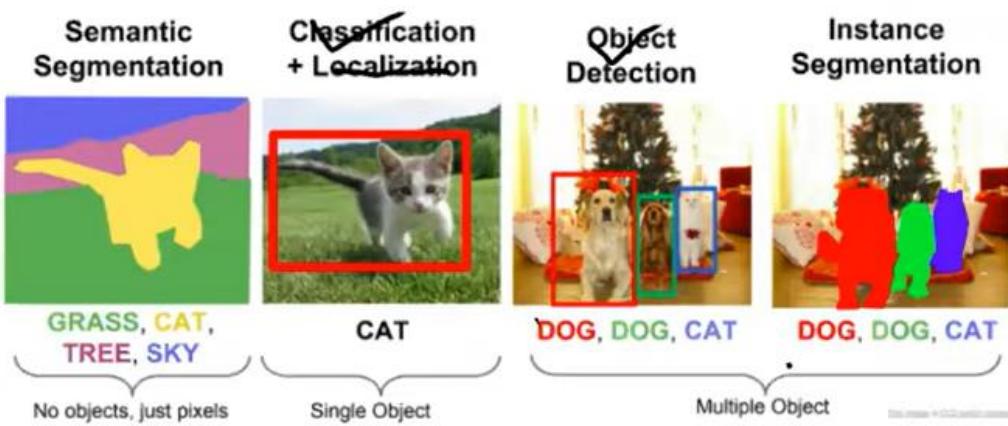
Input



Desired output



Creating accurate machine learning models capable of localizing and identifying multiple objects in a single image remains a core challenge in computer vision.



# Anaconda

- ▶ Anaconda is a Python distribution for large-scale data analytics (provided by Continuum Analytics, Inc.).
- ▶ Provides the tools to analyze large sets of data.
- ▶ Anaconda is a complete scientific stack for Python. It is a cross-platform python distribution for data analytics and scientific computing.
- ▶ Supports Python 2 &3
- ▶ Over 1000 packages
- ▶ Miniconda

# ANACONDA

## ANACONDA NAVIGATOR

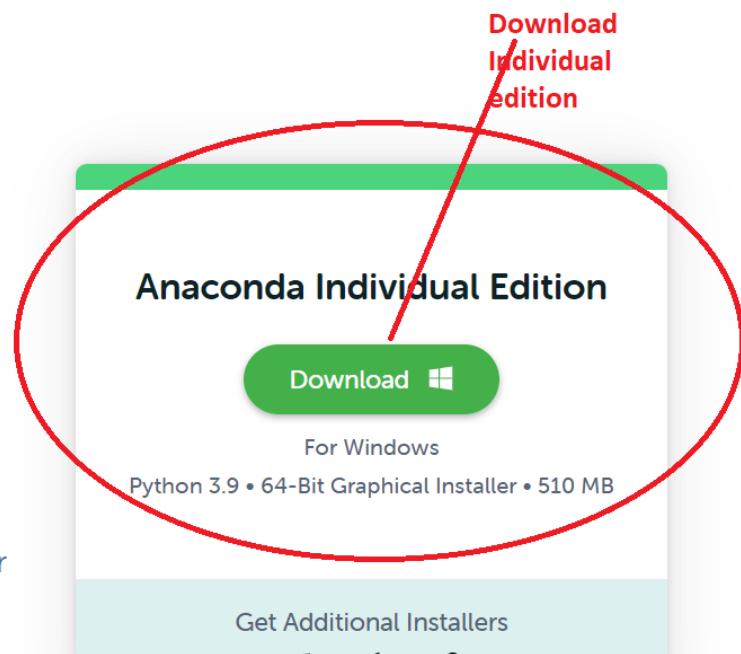
 <b>JupyterLab</b> 1.1.4 <p>An accessible interface for interacting with Jupyter notebooks or anything based on the Jupyter Notebook and its ecosystem.</p> <a href="#">Launch</a>	 <b>Jupyter Notebook</b> 1.1.4 <p>Create Jupyter notebooks containing code, equations, visualizations and rich media alongside text while documenting the data analysis.</p> <a href="#">Launch</a>	 <b>IPython</b> 4.0.1 <p>Run fast clean notebooks like IPynotebooks, proper multi-line editing with syntax highlighting, graphical outputs, and more.</p> <a href="#">Launch</a>
 <b>R</b> 3.4.3 <p>A set of integrated tools designed to help you live closer to your data with its packages and environments.</p> <a href="#">Launch</a>	 <b>Spyder</b> 3.1.3 <p>An intuitive Python IDE that makes scientific computing, development, and data visualization easy with advanced editing, intelligent navigation, code collapsing and communication features.</p> <a href="#">Launch</a>	 <b>Qlucore OmicsView</b> 0.13.0 <p>Qlucore OmicsView helps you analyze complex data. It provides interactive analysis, writing and sharing results directly.</p> <a href="#">Launch</a>
 <b>Orange3</b> 3.4.0 <p>A comprehensive machine learning data mining framework.</p> <a href="#">Launch</a>	 <b>Visual Studio Code</b> 1.33.2 <p>VERSATILE CODE EDITOR WITH SUPPORT FOR MULTIPLE LANGUAGES.</p> <a href="#">Launch</a>	



Individual Edition

# Your data science toolkit

With over 25 million users worldwide, the open-source Individual Edition (Distribution) is the easiest way to perform Python/R data science and machine learning on a single machine. Developed for solo practitioners, it is the toolkit that equips you to work with



Anaconda Individual Edition

Download 

For Windows  
Python 3.9 • 64-Bit Graphical Installer • 510 MB

Get Additional Installers

<https://www.anaconda.com/products/individual>



## PyCharm



- PyCharm is an integrated development environment (IDE) used in computer programming, specifically for the Python programming language.
- PyCharm is cross-platform, with Windows, macOS and Linux versions.
- The Community Edition is released under the Apache License; there is also Professional Edition with extra features
- PyCharm provides an API so that developers can write their own plugins to extend PyCharm features. Several plugins from other JetBrains IDE also work with PyCharm.



3



The project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 625233



Version: 2021.3.2  
Build: 213.6777.50  
31 January 2022

[System requirements](#)  
[Installation instructions](#)  
[Other versions](#)

## Download PyCharm

[Windows](#)[macOS](#)[Linux](#)

### Professional

For both Scientific and Web Python development. With HTML, JS, and SQL support.

[Download](#)[Free trial](#)

### Community

For pure Python development

[Download](#)

Free, built on open-source

[Download Community Edition](#)

<https://www.jetbrains.com/pycharm/download/#section=windows>

# TFOD Setup Steps

The screenshot shows a browser window displaying a GitHub page for "tfod-steps". The URL in the address bar is highlighted with a red box: <https://c17hawke.github.io/tfod-setup/>. The page title is "Configuration steps for TensorFlow object detection-". The main content area contains a list of four steps under the heading "STEP-1 Download the following content-". The steps are numbered 1 through 4 and are underlined. To the right of the content, there is a vertical sidebar with several configuration options and steps:

- for GPU
- for CPU only
- STEP-5 Install protobuf using conda package manager-
- STEP-6 For protobuff to .py conversion download from a tool from here-
  - For Linux or Mac
  - For Windows
- STEP-7 Paste all content present in utils into research folder-
- STEP-8 Paste ssd\_mobilenet\_v1\_coco or any other model downloaded from model zoo into research folder-
- STEP-9 Run the following to generate train and test records-
- STEP-10 Copy from research/object\_detection/sam... YOURMODEL.config file into research/training-
- STEP-11 Update num\_classes, fine\_tune\_checkpoint ,and num\_steps plus update input\_path and label\_map\_path for both train\_input\_reader and eval input reader-

Below the content area, there is a screenshot of a file explorer window showing four compressed files: "labelImg-master.zip", "models-1.13.0.zip", "ssd\_mobilenet\_v1\_coco\_2018\_01\_28.tar.gz", and "utils.zip".

<https://c17hawke.github.io/tfod-setup/>

# Download Repository

The screenshot shows a web browser displaying the GitHub repository page for `tensorflow/models`. The URL in the address bar is highlighted with a green box and contains the path `https://github.com/tensorflow/models/tree/v1.13.0`. A red box highlights this URL. A red arrow points from the `Download ZIP` button in the repository sidebar to the URL in the address bar.

<https://github.com/tensorflow/models/tree/v1.13.0>

github.com/tensorflow/models/tree/v1.13.0

KeepVid: Download... OpenCV with Visual... Video Lectures | Dis... The Opinion Polls N... December-2013 | In... Chicago Elsevier · Owen Bish... Inbox (491) - tabrej... Altmetric it! Altmetric it!

Why GitHub? Team Enterprise Explore Marketplace Pricing

Search Sign in Sign up

tensorflow / models Public Notifications Fork 45.5k Star 72.9k

Code Issues 1.1k Pull requests 171 Actions Projects 5 Wiki Security Insights

v1.13.0 41 branches 23 tags Go to file Code About

goldiegadde and tfboyd Revert "tf\_upgrade\_v2 on resnet and utils folder"

official Revert "tf\_upgrade\_v2 on resnet ar

research [LFADS] Fixes typo in distributions.

samples Fix #5814

tutorials update the calculation of num\_bat

.gitignore Fixed gitignore for mac's ds\_stor

.gitmodules Move the research models into a research subfolder (#2430)

AUTHORS Spatial Transformer model

CODEOWNERS Fix dependency issues (#5815)

CONTRIBUTING.md Fixing small typo

ISSUE TEMPLATE.md Update ISSUE TEMPLATE.md

Clone HTTPS GitHub CLI  
<https://github.com/tensorflow/models.git>

Use Git or checkout with SVN using the web URL.

Open with GitHub Desktop

Download ZIP

Readme View license 72.9k stars 2.9k watching 45.5k forks

Releases 21

TensorFlow Official Models 2.8.0 Latest 22 days ago + 20 releases

<https://github.com/tensorflow/models/tree/v1.13.0>

# Download Deep Learning Model from the model zoo

← → C [github.com/tensorflow/models/blob/master/research/object\\_detection/g3doc/tf1\\_detection\\_zoo.md](https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf1_detection_zoo.md) ↗ ⭐

KV KeepVid: Download... [OpenCV with Visual...](#) [MIT Video Lectures | Dis...](#) [The Opinion Polls N...](#) [December-2013 | In...](#) [Chicago](#) [Elsevier · Owen Bish...](#) [Inbox \(491\) - tabrej...](#) [Altmetric it!](#) [Edit](#)

	198 lines (164 sloc)	20.1 KB	<>	Raw	Blame	Copy	Edit
	<a href="#">ssd_mobilenet_v1_quantized_coco</a> ☆	29	18	Boxes			
	<a href="#">ssd_mobilenet_v1_0.75_depth_quantized_coco</a> ☆	29	16	Boxes			
	<a href="#">ssd_mobilenet_v1_ppn_coco</a> ☆	26	20	Boxes			
	<a href="#">ssd_mobilenet_v1_fpn_coco</a> ☆	56	32	Boxes			
	<a href="#">ssd_resnet_50_fpn_coco</a>	76	35	Boxes			
	<a href="#">ssd_mobilenet_v2_coco</a>	31	22	Boxes			
	<a href="#">ssd_mobilenet_v2_quantized_coco</a>	29	22	Boxes			
	<a href="#">ssdlite_mobilenet_v2_coco</a>	27	22	Boxes			
	<a href="#">ssd_inception_v2_coco</a>	42	24	Boxes			
	<a href="#">faster_rcnn_inception_v2_coco</a>	58	28	Boxes			
	<a href="#">faster_rcnn_resnet50_coco</a>	89	30	Boxes			
	<a href="#">faster_rcnn_resnet50_lowproposals_coco</a>	64		Boxes			
	<a href="#">rfcn_resnet101_coco</a>	92	30	Boxes			
	<a href="#">faster_rcnn_resnet101_coco</a>	106	32	Boxes			
	<a href="#">faster_rcnn_resnet101_lowproposals_coco</a>	82		Boxes			
	<a href="#">faster_rcnn_inception_resnet_v2_atrous_coco</a>	620	37	Boxes			
	<a href="#">faster_rcnn_inception_resnet_v2_atrous_lowproposals_coco</a>	241		Boxes			
	<a href="#">faster_rcnn_nas</a>	1922	12	Boxes			

[http://download.tensorflow.org/models/object\\_detection/faster\\_rcnn\\_inception\\_v2\\_coco\\_2018\\_01\\_28.tar.gz](http://download.tensorflow.org/models/object_detection/faster_rcnn_inception_v2_coco_2018_01_28.tar.gz)

# 1. Download Dataset & utils.

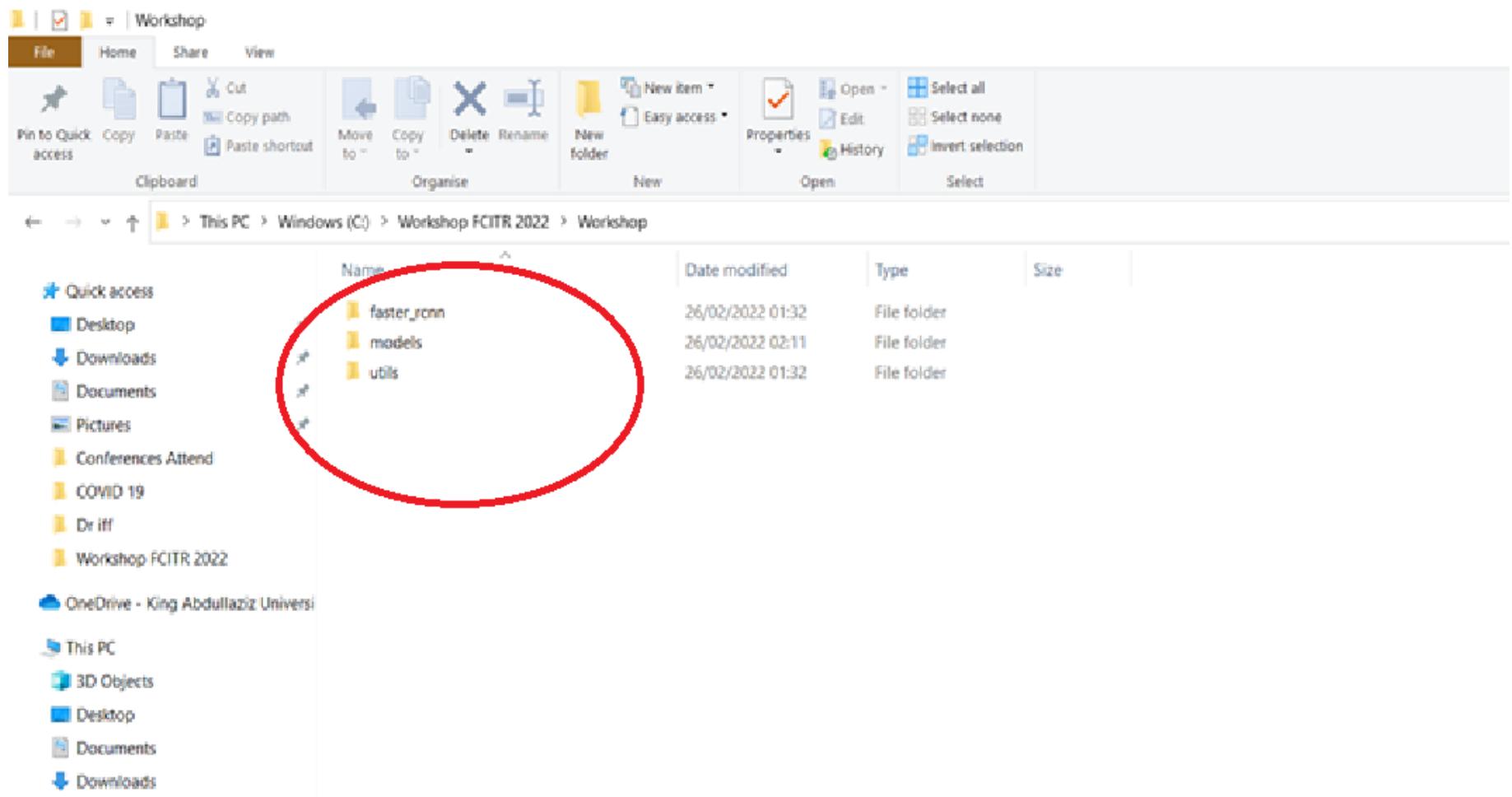
The screenshot shows a Google Drive file viewer for a zip archive named 'utils.zip'. The archive contains one item, 'utils'. The interface includes a header bar with browser controls, a toolbar with various icons, and a main content area displaying the file structure. A red circle highlights the download icon (a downward arrow) in the top right corner of the content area.

utils.zip 1 item

Name	Last modified	File size
utils	-	-

[https://drive.google.com/file/d/12F5oGAuQg7qBM\\_267TCMt\\_rlorV-M7gf/view](https://drive.google.com/file/d/12F5oGAuQg7qBM_267TCMt_rlorV-M7gf/view)

# Unzip All the folder and rename the file



The screenshot shows a Windows File Explorer window with the following details:

**File Explorer Navigation:** Workshop > This PC > Windows (C:) > Workshop FCITR 2022 > Workshop

**Left Sidebar (Quick Access):**

- Quick access
- Desktop
- Downloads
- Documents
- Pictures
- Conferences Attend
- COVID 19
- Dr iff
- Workshop FCITR 2022

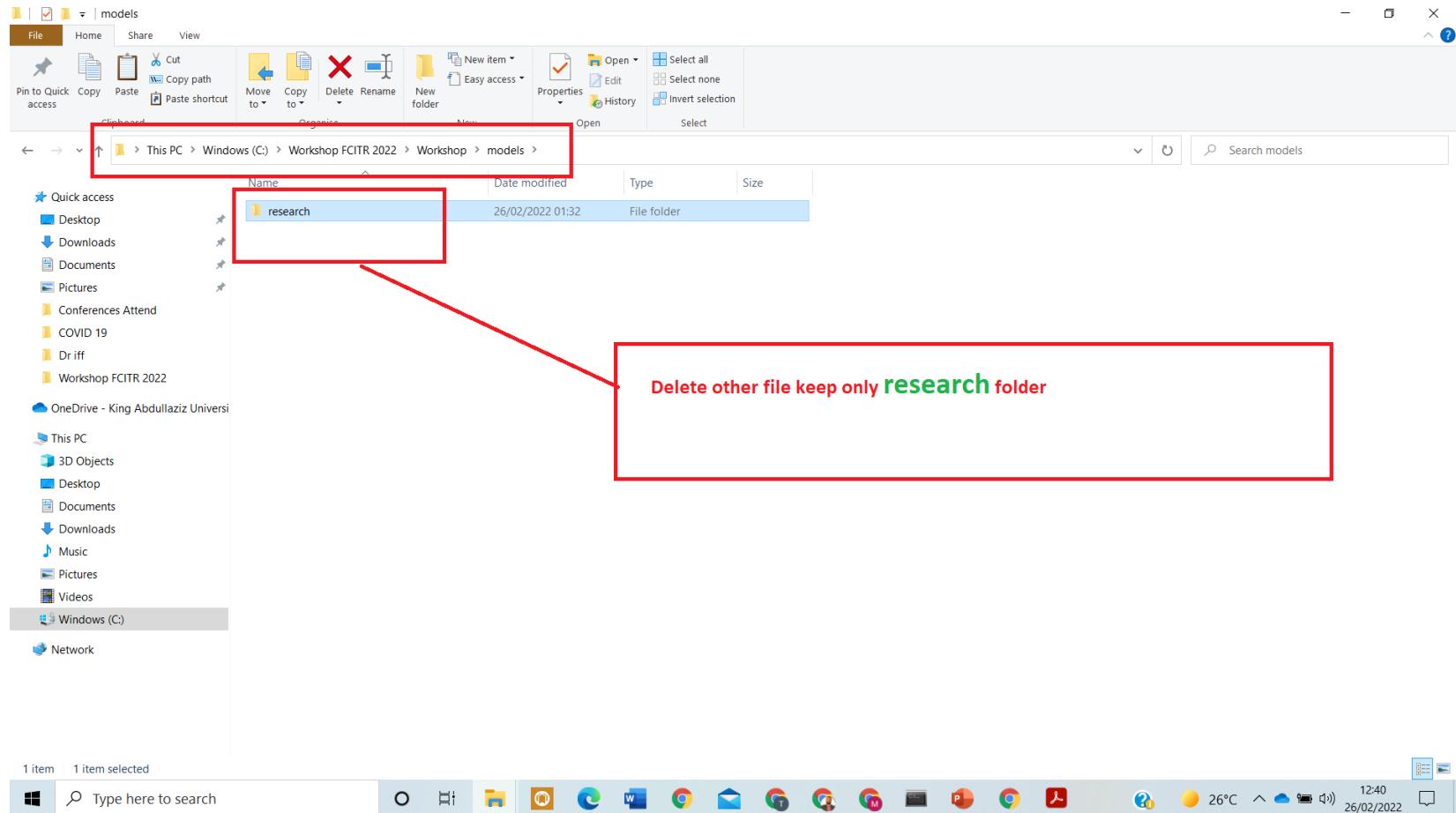
**OneDrive - King Abdullah University of Science and Technology**

**Right Panel (Content View):**

Name	Date modified	Type	Size
faster.jsonn	26/02/2022 01:32	File folder	
models	26/02/2022 02:11	File folder	
utils	26/02/2022 01:32	File folder	

A red oval highlights the three folders: faster.jsonn, models, and utils.

# In Model folder keep only research folder



# Create Virtual Environment

```
conda create -n tfod python=3.6
```

Anaconda Prompt (anaconda3)

```
(base) C:\Users\tabrej>conda create -n tfod1 python=3.6
```

Anaconda Prompt (anaconda3)

```
vc                         conda-forge/win-64::vc-14.2-hb210afc_6
vs2015_runtime      conda-forge/win-64::vs2015_runtime-14.29.30037-h902a5da_6
wheel                     conda-forge/noarch::wheel-0.37.1-pyhd8ed1ab_0
```

```
Proceed ([y]/n)? 
```

```
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate tfod1
#
# To deactivate an active environment, use
#
#     $ conda deactivate
```

```
(base) C:\Users\tabrej>conda activate tfod1
```

```
(tfod1) C:\Users\tabrej>
```

# Install protobuf using conda package manager

conda install -c anaconda protobuf

A screenshot of a Windows desktop environment. At the top, there's a taskbar with icons for File Explorer, Microsoft Edge, Word, and others. Below the taskbar is a system tray with icons for battery, signal, and volume. The main focus is a terminal window titled "Anaconda Prompt (anaconda3)". The window shows the command "conda install -c anaconda protobuf" being run, followed by a large amount of output from the conda package manager. The output includes various package names like cffi, pycparser, and many tensorflow-related packages, along with their versions and download speeds. The terminal window has a red border around the command line. The background of the desktop is a light blue gradient.

```
AutoSave (C) Off | DeepLearning.pptx | Search (Alt+Q) ***** KHAN ***** AMEEN
File Anaconda Prompt (anaconda3)
Paste Collecting cffi>=1.0.1
Using cached cffi-1.15.0-cp36-cp36m-win_amd64.whl (178 kB)
Collecting pycparser
Using cached pycparser-2.21-py2.py3-none-any.whl (118 kB)
Installing collected packages: zipp, typing-extensions, six, ipython-genutils, decorator, traitlets, pywin32, pyrsistent, importlib-metadata, attrs, wcwidth, tornado, pyzmq, python-dateutil, pyparsing, pycparser, parso, nest-asyncio, jupyter-core, jsonschema, entrypoints, webencodings, pygments, prompt-toolkit, pickleshare, packaging, nbformat, MarkupSafe, jupyter-client, jedi, colorama, cffi, backcall, async-generator, testpath, pywinpty, pandocfilters, nbclient, mistune, jupyterlab-pygments, jinja2, ipython, defusedxml, dataclasses, bleach, argon2-cffi-bindings, terminado, Send2Trash, prometheus-client, nbconvert, ipykernel, argon2-cffi, numpy, notebook, cached-property, widgetsnbextension, werkzeug, qtpy, protobuf, markdown, jupyterlab-widgets, h5py, grpcio, absl-py, wrapt, termcolor, tensorflow-estimator, tensorboard, qtconsole, pytz, pillow, opt-einsum, kiwisolver, keras-preprocessing, keras-applications, jupyter-console, iwidgets, google-pasta, gast, cycler, astor, tensorflow, pandas, opencv-python, matplotlib, lxml, jupyter, Cython, contextlib2
Successfully installed Cython-0.29.28 MarkupSafe-2.0.1 Send2Trash-1.8.0 absl-py-1.0.0 argon2-cffi-21.3.0 argon2-cffi-bindings-21.2.0
decorator-0.8.1 async-generator-1.10 attrs-21.4.0 backcall-0.2.0 bleach-4.1.0 cached-property-1.5.2 cffi-1.15.0 colorama-0.4.4 contextlib2-0.6.0
cycler-0.11.0 dataclasses-0.8 decorator-5.1.1 defusedxml-0.7.1 entrypoints-0.4 gast-0.2.2 google-pasta-0.2.0 grpcio-1.44.0 h5py-1.0
importlib-metadata-4.8.3 ipykernel-5.5.6 ipython-7.16.3 ipython-genutils-0.2.0 ipywidgets-7.6.5 jedi-0.17.2 jinja2-3.0.3 jsonschema-3.2.0
jupyter-1.0.0 jupyter-client-7.1.2 jupyter-console-6.4.0 jupyter-core-4.9.2 jupyterlab-pygments-0.1.2 jupyterlab-widgets-1.0.0
keras-applications-1.0.8 keras-preprocessing-1.1.2 kiwisolver-1.3.1 lxml-4.8.0 markdown-3.3.6 matplotlib-3.3.4 mistune-0.8.4 nbclient-5.9
nbconvert-6.0.7 nbformat-5.1.3 nest-asyncio-1.5.4 notebook-6.4.8 numpy-1.19.5 opencv-python-4.5.5.62 opt-einsum-3.3.0 packaging-3.3
pandas-1.1.5 pandocfilters-1.5.0 parso-0.7.1 pickleshare-0.7.5 pillow-8.4.0 prometheus-client-0.13.1 prompt-toolkit-3.0.28 protobuf-19.4
pycparser-2.21 pygments-2.11.2 pyparsing-3.0.7 pyrsistent-0.18.0 python-dateutil-2.8.2 pytz-2021.3 pywin32-303 pywinpty-1.1.6
zmq-22.3.0 qtconsole-5.2.2 qtpy-2.0.1 six-1.16.0 tensorboard-1.15.0 tensorflow-1.15.0 tensorflow-estimator-1.15.1 termcolor-1.1.0 tensorflow-0.12.1
testpath-0.6.0 tornado-6.1 traitlets-4.3.3 typing-extensions-4.1.1 wcwidth-0.2.5 webencodings-0.5.1 werkzeug-2.0.3 widget-extension-3.5.2 wrapt-1.13.3 zipp-3.6.0
(tfod1) C:\Users\tabrej>
```

Slide 18 of 20 20:46 26/02/2022

# Install the following packages in your new environment

**for CPU only**

```
pip install pillow lxml Cython contextlib2 jupyter matplotlib pandas  
opencv-python tensorflow==1.15.0
```

```
Proceed ([y]/n)? y
```

```
Preparing transaction: done  
Verifying transaction: done  
Executing transaction: done  
#  
# To activate this environment, use  
#  
#     $ conda activate tfod1  
#  
# To deactivate an active environment, use  
#  
#     $ conda deactivate
```

```
(base) C:\Users\tabrej>conda activate tfod1
```

```
(tfod1) C:\Users\tabrej>pip install pillow lxml Cython contextlib2 jupyter matplotlib pandas opencv-python tensorflow==1.15.0
```

These are Proto files

The screenshot shows a Windows File Explorer window with the following details:

- Address Bar:** This PC > Windows (C:) > Workshop FCITR 2022 > Workshop > models > research > object\_detection > protos
- File Explorer View:** The folder contains 30 items, primarily proto files. A red circle highlights the following files in the second row:
  - image\_resizer.proto
  - input\_reader.proto
  - keypoint\_box\_coder.proto
  - losses.proto
  - matcher.proto
  - mean\_stddev\_box\_coder.proto
  - model.proto
  - multiscale\_anchored\_generator.proto
  - optimizer.proto
  - pipeline.proto
  - post\_processing.proto
  - preprocessor.proto
- Red Box:** A red box highlights the entire folder path in the address bar.
- Sidebar:** Shows quick access links and recent locations.
- Taskbar:** Shows the Start button, search bar, and various pinned application icons.

# Install protobuf using conda package manager

conda install protobuf

```
Anaconda Prompt (anaconda3) - conda install protobuf

(tfod2) C:\Users\tabrejconda install protobuf
C:\Users\tabrej\anaconda3\lib\site-packages\requests\__init__.py:89: RequestsDependencyWarning: urllib3 (1.26.7) or chardet (3.0.4) doesn't match a supported version!
  warnings.warn("urllib3 ({}) or chardet ({}) doesn't match a supported "
Collecting package metadata (current_repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 4.9.2
  latest version: 4.11.0

Please update conda by running

$ conda update -n base -c defaults conda

## Package Plan ##

environment location: C:\Users\tabrej\anaconda3\envs\tfod2

added / updated specs:
- protobuf

The following packages will be downloaded:

  package           | build
  -----           |
libprotobuf-3.19.4 | h7755175_0    2.4 MB  conda-forge
```

# Move to your research directory using cd Command (cmd)

```
research
Anaconda Prompt (anaconda3)
python=3.6 -> vs2015_runtime[version='>=14.16.27012|>=14.16.27012,<15.0a0']
protobuf -> vs2015_runtime[version='>=14.16.27012|>=14.16.27012,<15.0a0|>=14.16.27033']
python_abi -> python=3.10 -> vs2015_runtime[version='>=14.16.27012,<15.0a0|>=14.16.27012|>=14.16.27033']
python=3.6 -> vc[version='>=14.1,<15.0a0'] -> vs2015_runtime[version='>=14.0.25123,<15.0a0|>=14.0.25420|>=14.15.26706|>=14.28.29325|>=14.29.30037|>=14.16.27033|>=14.27.29016']
vc -> vs2015_runtime[version='>=14.0.25123,<15.0a0|>=14.0.25420|>=14.15.26706|>=14.16.27012|>=14.28.29325|>=14.29.30037|>=14.16.27033|>=14.27.29016']
protobuf -> vc[version='>=14.1,<15.0a0'] -> vs2015_runtime[version='>=14.0.25123,<15.0a0|>=14.0.25420|>=14.15.26706|>=14.28.29325|>=14.29.30037|>=14.27.29016']
wheel -> python -> vs2015_runtime[version='>=14.16.27012|>=14.16.27012,<15.0a0|>=14.16.27033']

Package ucrt conflicts for:
protobuf -> vs2015_runtime[version='>=14.16.27012,<15.0a0'] -> ucrt[version='>=10.0.20348.0']
ucrt
python=3.6 -> vs2015_runtime[version='>=14.16.27012,<15.0a0'] -> ucrt[version='>=10.0.20348.0']
vc -> vs2015_runtime[version='>=14.15.26706'] -> ucrt[version='>=10.0.20348.0']

Package pyparsing conflicts for:
wheel -> packaging[version='>=20.2'] -> pyparsing[version='<3,>=2.0.2|>=2.0.2|>=2.0.2,!>=3.0.5|>=2.0.2,<3']
pip -> packaging -> pyparsing[version='<3,>=2.0.2|>=2.0.2|>=2.0.2,!>=3.0.5|>=2.0.2,<3']
Note that strict channel priority may have removed packages required for satisfiability.

(tfod1) C:\Users\tabrej>cd C:\Workshop FCITR 2022\Workshop\models\research
```

Move to research folder

# For protobuff to .py conversion

```
protoc object_detection/protos/*.proto --python_out=.
```

```
Anaconda Prompt (anaconda3) - conda install protobuf  
(tfod2) C:\Users\tabrej>cd C:\Workshop FCITR 2022\Workshop\models\research  
(tfod2) C:\Workshop FCITR 2022\Workshop\models\research>protoc object_detection/protos/*.proto --python_out=.  
(tfod2) C:\Workshop FCITR 2022\Workshop\models\research>
```

# Visualization inspection

The screenshot shows a Windows File Explorer window displaying a folder named 'protos' located at 'C:\Workshop FCITR 2022\Workshop\models\research\object\_detection\protos'. The window contains 59 items, primarily Python source files (.py) and protocol buffer files (.proto). The files are organized into several rows:

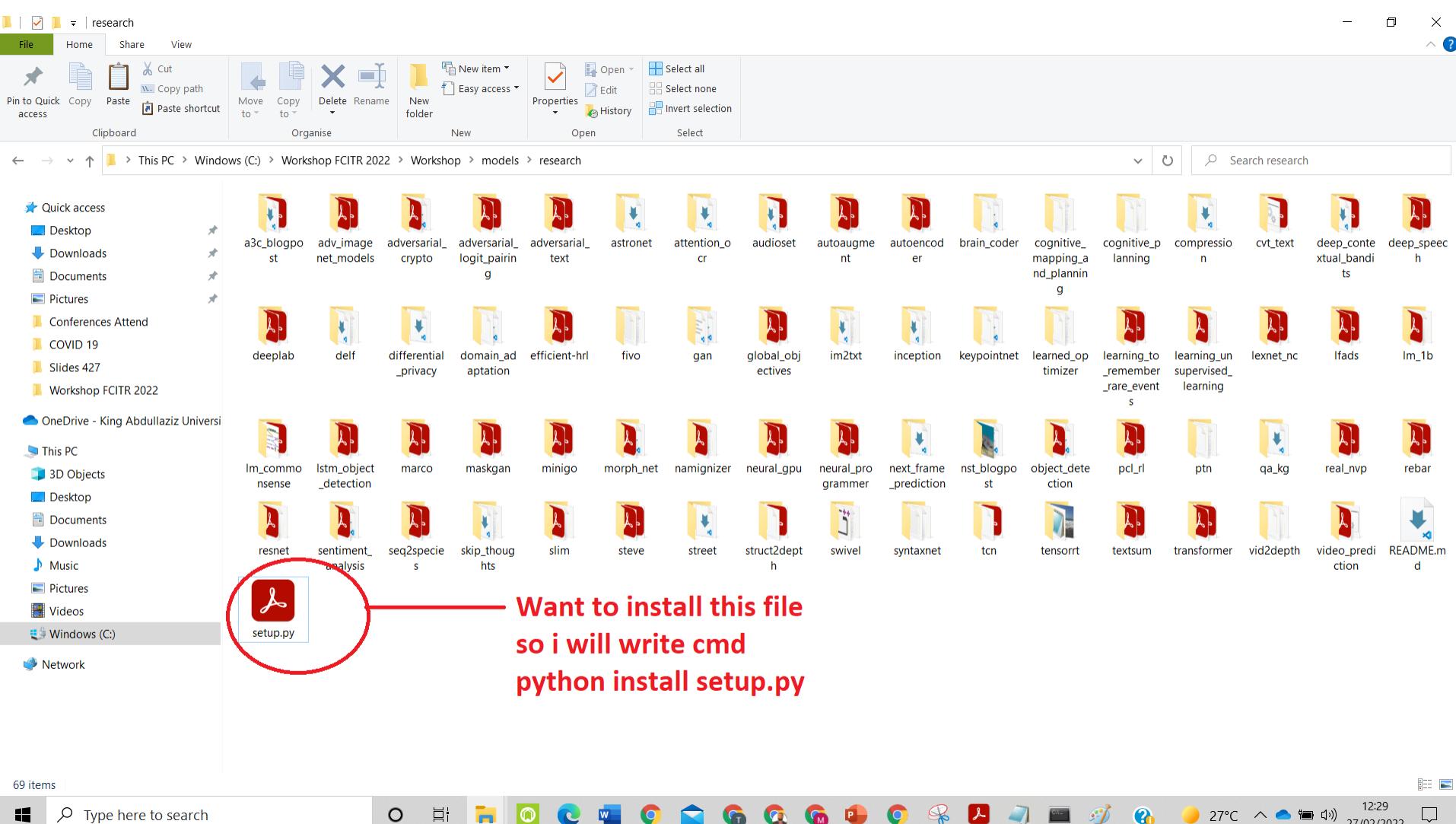
- Row 1:** \_init\_.py, anchor\_generator.proto, anchor\_generator\_pb2.py, argmax\_matcher.proto, argmax\_matcher\_pb2.py, bipartite\_matcher.proto, bipartite\_matcher\_pb2.py, box\_coder.proto, box\_coder\_pb2.py, box\_predictor.proto, box\_predictor\_pb2.py, eval.proto, eval\_pb2.py, faster\_rcnn.proto, faster\_rcnn\_box\_coder.proto, faster\_rcnn\_box\_coder\_pb2.py, faster\_rcnn\_pb2.py.
- Row 2:** graph\_rewriter.proto, graph\_rewriter\_pb2.py, grid\_anchor\_generator.proto, grid\_anchor\_generator\_pb2.py, hyperparameters.proto, hyperparameters\_pb2.py, image\_resizer.proto, image\_resizer\_pb2.py, input\_reader.proto, input\_reader\_pb2.py, keypoint\_box\_coder.proto, keypoint\_box\_coder\_pb2.py, losses.proto, losses\_pb2.py, matcher.proto, matcher\_pb2.py, mean\_stddev\_box\_coder.proto.
- Row 3:** mean\_stddev\_box\_coder\_pb2.py, model.proto, model\_pb2.py, multiscale\_anchor\_generator.proto, multiscale\_anchor\_generator\_pb2.py, optimizer.proto, optimizer\_pb2.py, pipeline.proto, pipeline\_pb2.py, post\_processing.proto, post\_processing\_pb2.py, preprocess.proto, preprocess\_pb2.py, region\_similarity\_calculator.proto, region\_similarity\_calculator\_pb2.py, square\_box\_coder.proto, square\_box\_coder\_pb2.py.
- Row 4:** ssd.proto, ssd\_anchor\_generator.proto, ssd\_anchor\_generator\_pb2.py, ssd\_anchor\_generator\_pb2.py, string\_int\_label\_map.proto, string\_int\_label\_map\_pb2.py, train.proto, train\_pb2.py.

Several files are circled in red, highlighting them for inspection:

- \_init\_.py
- argmax\_matcher.proto
- box\_predictor.proto
- eval.proto
- image\_resizer.proto
- keypoint\_box\_coder.proto
- losses.proto
- post\_processing.proto
- region\_similarity\_calculator.proto
- square\_box\_coder.proto

The left sidebar shows the user's library, including Quick access, Desktop, Downloads, Documents, Pictures, Conferences Attend, COVID 19, Slides 427, Workshop FCITR 2022, OneDrive - King Abdullah University of Science and Technology, This PC, 3D Objects, Desktop, Documents, Downloads, Music, Pictures, Videos, and Network. The bottom taskbar includes icons for Start, Search, Task View, File Explorer, Edge, Word, Excel, Google Chrome, Mail, and other system icons. The status bar at the bottom right shows the date (27/02/2022), time (12:19), temperature (27°C), battery level, and signal strength.

# Setup.py file



# To install setup.py file

Anaconda Prompt (anaconda3) - conda install protobuf

```
26/02/2022 01:32 <DIR>      namigner
26/02/2022 01:32 <DIR>      neural_gpu
26/02/2022 01:32 <DIR>      neural_programmer
26/02/2022 01:32 <DIR>      next_frame_prediction
26/02/2022 01:32 <DIR>      nst_blogpost
26/02/2022 01:32 <DIR>      object_detection
26/02/2022 01:32 <DIR>      pcl_rl
26/02/2022 01:32 <DIR>      ptn
26/02/2022 01:32 <DIR>      qa_kg
06/02/2019 22:49      5,008 README.md
26/02/2022 01:32 <DIR>      real_nvp
26/02/2022 01:32 <DIR>      rebar
26/02/2022 01:32 <DIR>      resnet
26/02/2022 01:32 <DIR>      sentiment_analysis
26/02/2022 01:32 <DIR>      seq2species
06/02/2019 22:49      446 setup.py
26/02/2022 01:32 <DIR>      skip_thoughts
26/02/2022 01:32 <DIR>      slim
26/02/2022 01:32 <DIR>      steve
26/02/2022 01:32 <DIR>      street
26/02/2022 01:32 <DIR>      struct2depth
26/02/2022 01:32 <DIR>      swivel
26/02/2022 01:32 <DIR>      syntaxnet
26/02/2022 01:32 <DIR>      tcn
26/02/2022 01:32 <DIR>      tensorrt
26/02/2022 01:32 <DIR>      textsum
26/02/2022 01:32 <DIR>      transformer
26/02/2022 01:32 <DIR>      vid2depth
26/02/2022 01:32 <DIR>      video_prediction
2 File(s)          5,454 bytes
69 Dir(s)  40,665,108,480 bytes free
```

(tfod2) C:\Workshop FCITR 2022\Workshop\models\research>python setup.py install

Type here to search



12:30

27/02/2022

27°C

# To verify everything install correct or not

```
Anaconda Prompt (anaconda3) - conda install protobuf
Adding fonttools 4.29.1 to easy-install.pth file
Installing fonttools-script.py script to C:\Users\tabrej\anaconda3\envs\tfod2\Scripts
Installing fonttools.exe script to C:\Users\tabrej\anaconda3\envs\tfod2\Scripts
Installing pyftmerge-script.py script to C:\Users\tabrej\anaconda3\envs\tfod2\Scripts
Installing pyftmerge.exe script to C:\Users\tabrej\anaconda3\envs\tfod2\Scripts
Installing pyftsubset-script.py script to C:\Users\tabrej\anaconda3\envs\tfod2\Scripts
Installing pyftsubset.exe script to C:\Users\tabrej\anaconda3\envs\tfod2\Scripts
Installing ttx-script.py script to C:\Users\tabrej\anaconda3\envs\tfod2\Scripts
Installing ttx.exe script to C:\Users\tabrej\anaconda3\envs\tfod2\Scripts

Using c:\users\tabrej\anaconda3\envs\tfod2\lib\site-packages
Searching for cycler==0.11.0
Best match: cycler 0.11.0
Adding cycler 0.11.0 to easy-install.pth file

Using c:\users\tabrej\anaconda3\envs\tfod2\lib\site-packages
Searching for six==1.16.0
Best match: six 1.16.0
Adding six 1.16.0 to easy-install.pth file

Using c:\users\tabrej\anaconda3\envs\tfod2\lib\site-packages
Finished processing dependencies for object-detection==0.1

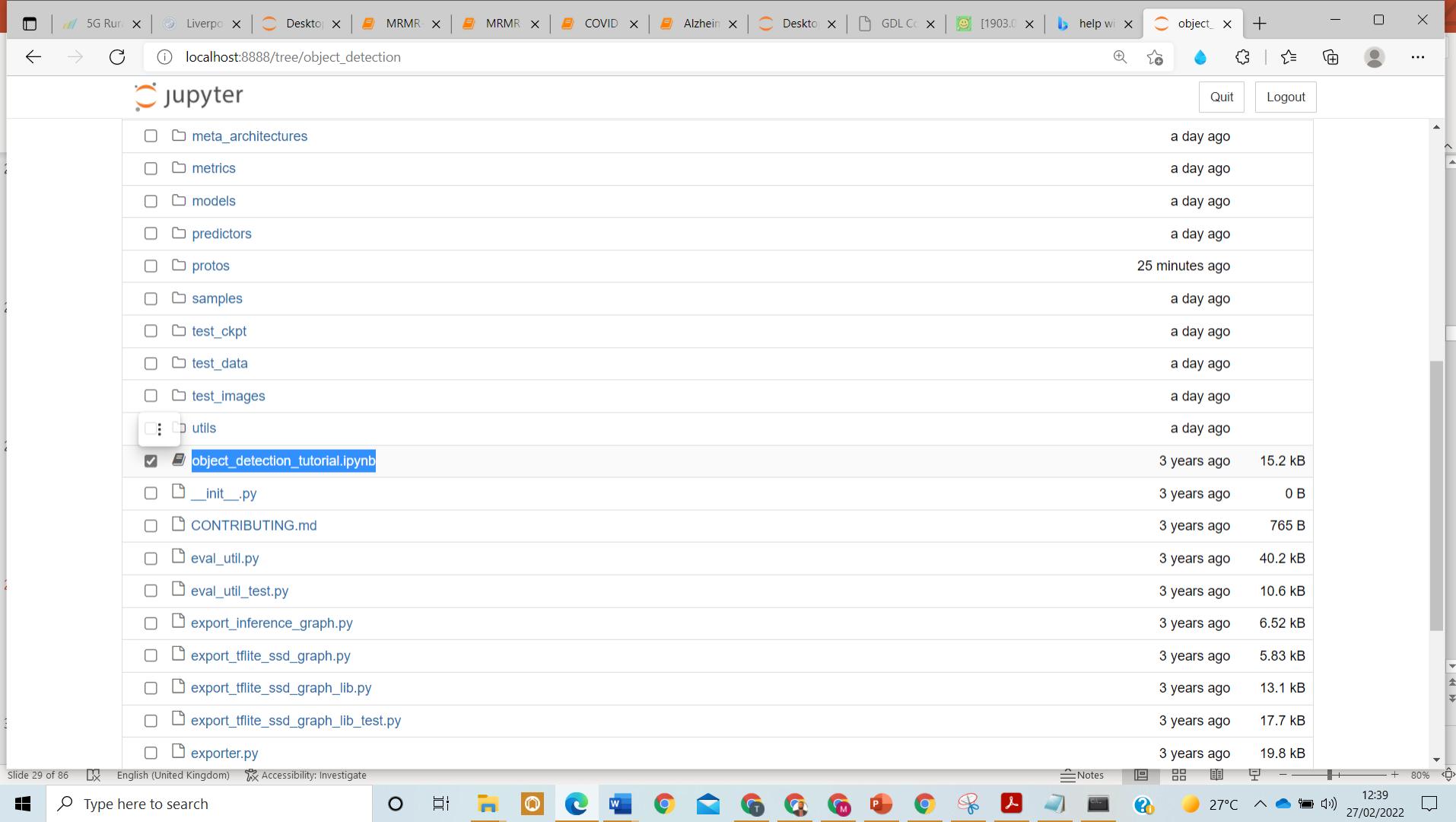
(tfod2) C:\Workshop FCITR 2022\Workshop\models\research>jupyter notebook
```



# Jupyter notebook will open and there is folder Object\_Detection

The screenshot shows a Windows desktop environment. At the top, a browser window is open to `localhost:8888/tree`, displaying a Jupyter Notebook file tree. The 'object\_detection' folder is selected, indicated by a blue border and a checked checkbox icon. The file tree includes various other projects like 'lstm\_object\_detection', 'marco', 'maskgan', etc. Below the browser is a taskbar with several pinned icons, including File Explorer, Edge, and various Google services. A search bar at the bottom left says 'Type here to search'. The system tray at the bottom right shows the date (27/02/2022), time (12:38), and weather (27°C).

File/Folder	Last Modified
lstm_object_detection	a day ago
marco	a day ago
maskgan	a day ago
minigo	a day ago
morph_net	a day ago
namigner	a day ago
neural_gpu	a day ago
neural_programmer	a day ago
next_frame_prediction	a day ago
... nst_blogpost	a day ago
<input checked="" type="checkbox"/> object_detection	a day ago
object_detection.egg-info	6 minutes ago
pcl_rl	a day ago
ptn	a day ago
qa_kg	a day ago
real_nvp	a day ago
rebar	a day ago
resnet	a day ago
sentiment_analysis	a day ago
seq2species	a day ago



# Verification Part is done

Screenshot of a Jupyter Notebook session titled "object\_detection\_tutorial" running on Python 3 (ipykernel). The notebook displays code for plotting an image and its corresponding object detection results.

The code in the cell shows:

```
plt.figure(figsize=IMAGE_SIZE)
plt.imshow(image_np)
```

The output of the cell shows the image with detected objects and their confidence levels:

```
In [14]: %matplotlib inline
plt.figure(figsize=(50,50))
plt.imshow(image_np)

Out[14]: <matplotlib.image.AxesImage at 0x2d51c795148>
```

The image shows a beach scene with several people and kites. Bounding boxes and confidence percentages are displayed for each detected object:

- A kite in the sky is labeled "kite: 76%".
- A kite in the sky is labeled "kite: 75%".
- A kite in the sky is labeled "kite: 82%" and another is labeled "kite: 60%".
- Two people on the beach are labeled "person: 51%" and "person: 91%".
- Three people on the beach are labeled "person: 77%", "person: 63%", and "person: 58%".

The Jupyter interface includes a toolbar, a Trusted button, and a Logout link. The browser address bar shows the URL: localhost:8888/notebooks/object\_detection/object\_detection\_tutorial.ipynb. The bottom of the screen shows the Windows taskbar with various application icons and the system clock indicating 12:46 on 27/02/2022.

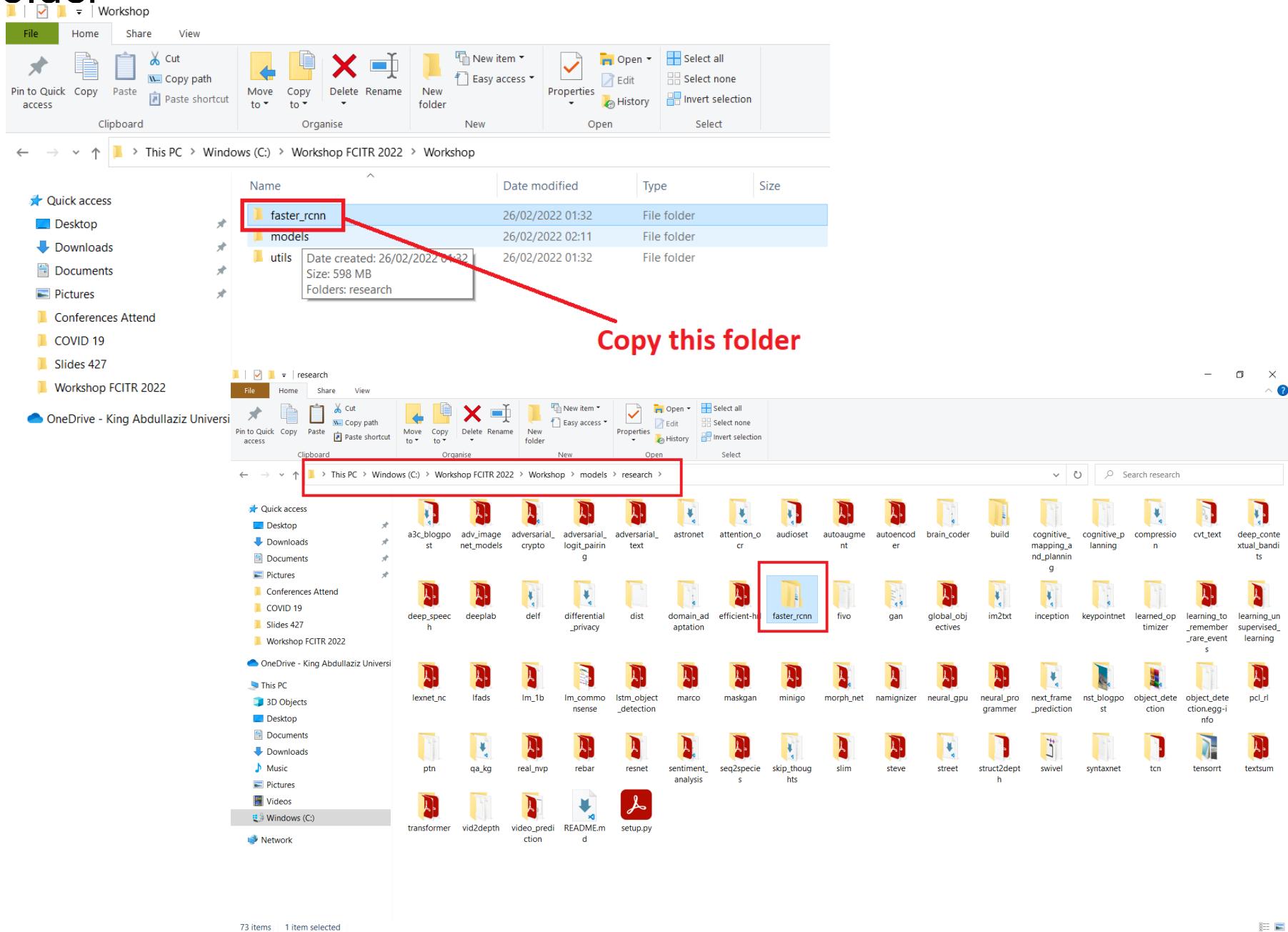


الدورات العملية التخصصية ل كلية الحاسوب وتقنية  
المعلومات برابغ

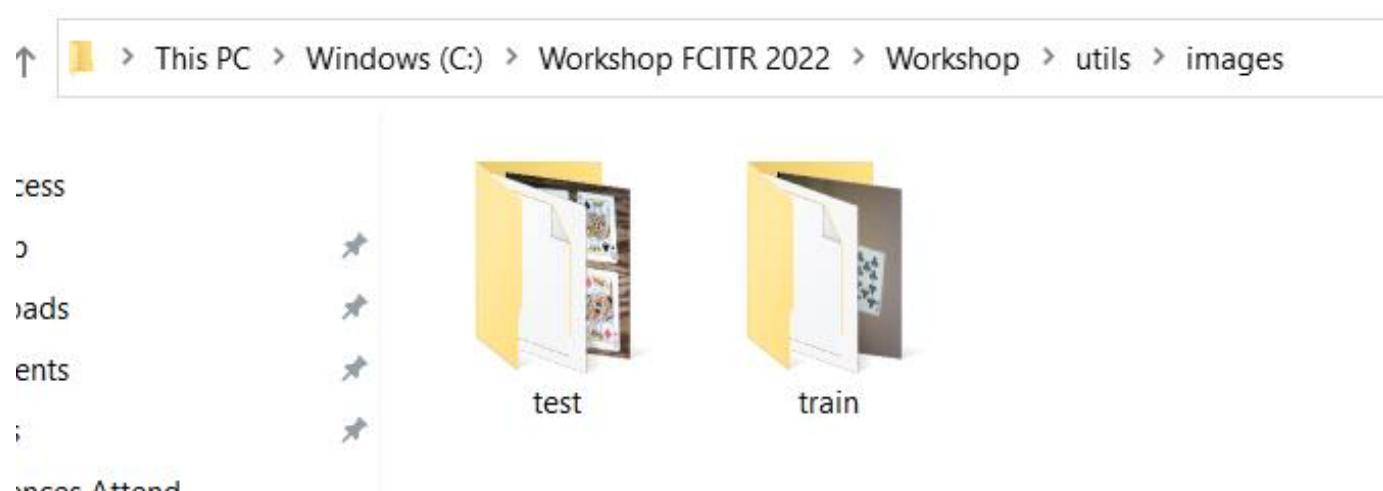
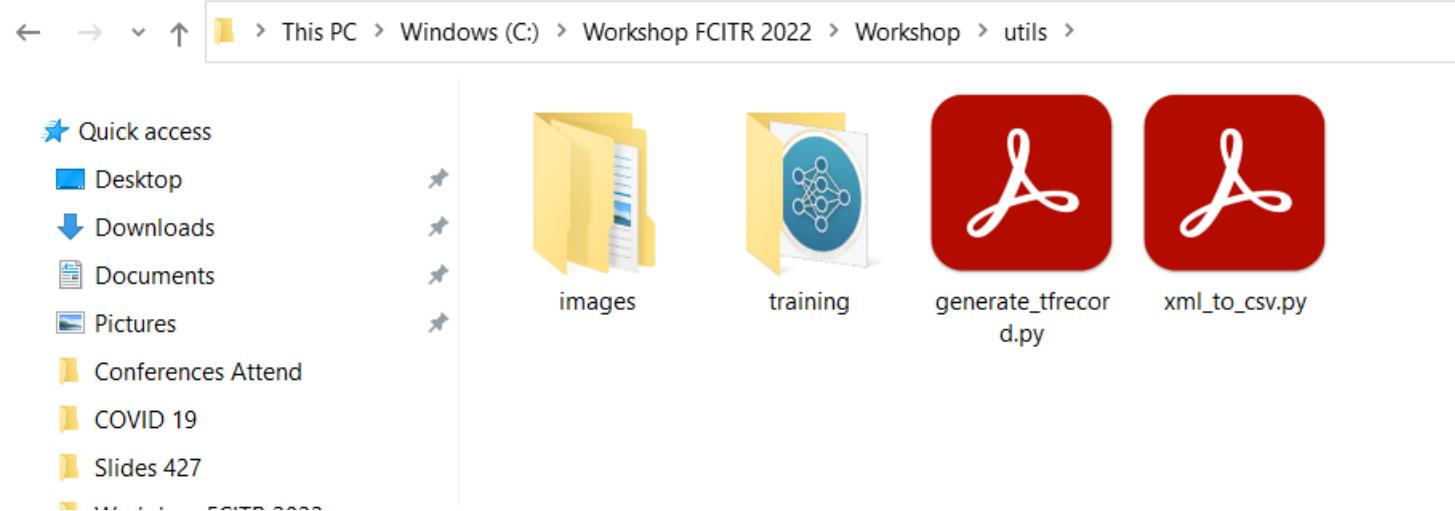


# Custom Model

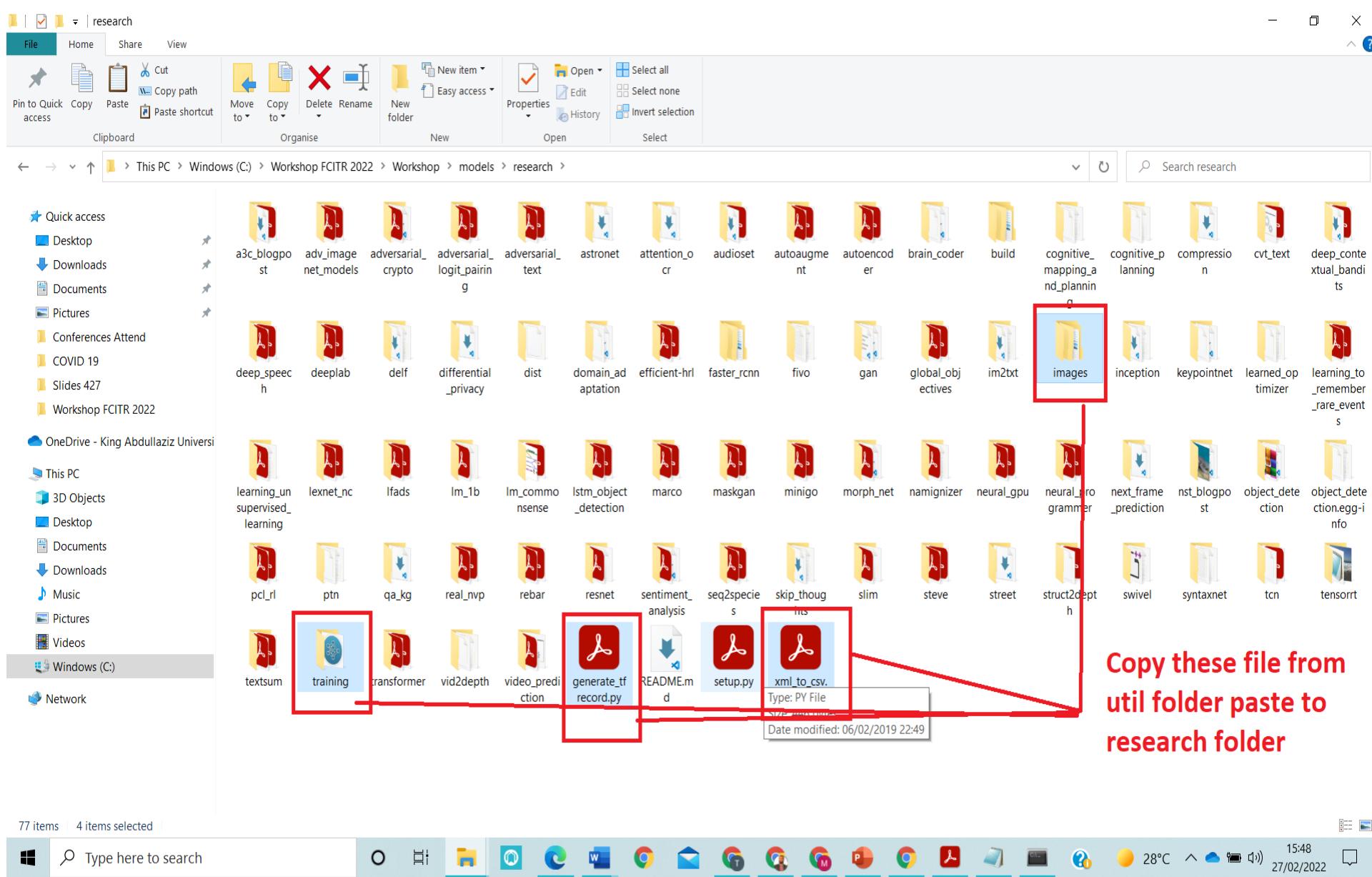
# Step 1: Copy the folder **faster\_rcnn** and paste to inside **research** folder



# Overview of Util file



## Step 2: Copy the file from util folder and paste to inside research folder



# Install LabelImg

← → ⌂ [github.com/tzutalin/labelImg](https://github.com/tzutalin/labelImg)

KV KeepVid: Download... W OpenCV with Visual... MIT Video Lectures | Dis... S S The Opinion Polls N... PKP December-2013 | In... Chicago Elsevier · Owen Bish... M Inbox (

README.rst

Annotations are saved as XML files in PASCAL VOC format, the format used by [ImageNet](#). Besides, it also supports YOLO and CreateML formats.

The screenshot shows the LabelImg application interface. On the left, there's a sidebar with icons for Open, Open Dir, Next Image, Prev Image, and Save. The main area displays a soccer match between Brazil and Argentina. A bounding box is drawn around a Brazilian player in a yellow jersey. A small window is open, showing a list of labels: dog, person, cat, tv, and car. The word 'person' is selected. To the right, a panel titled 'Box Labels' shows two entries, both labeled 'person'. Below this is a 'File List' containing a series of identical paths: /Users/rflynn/src/labelImg/dem.

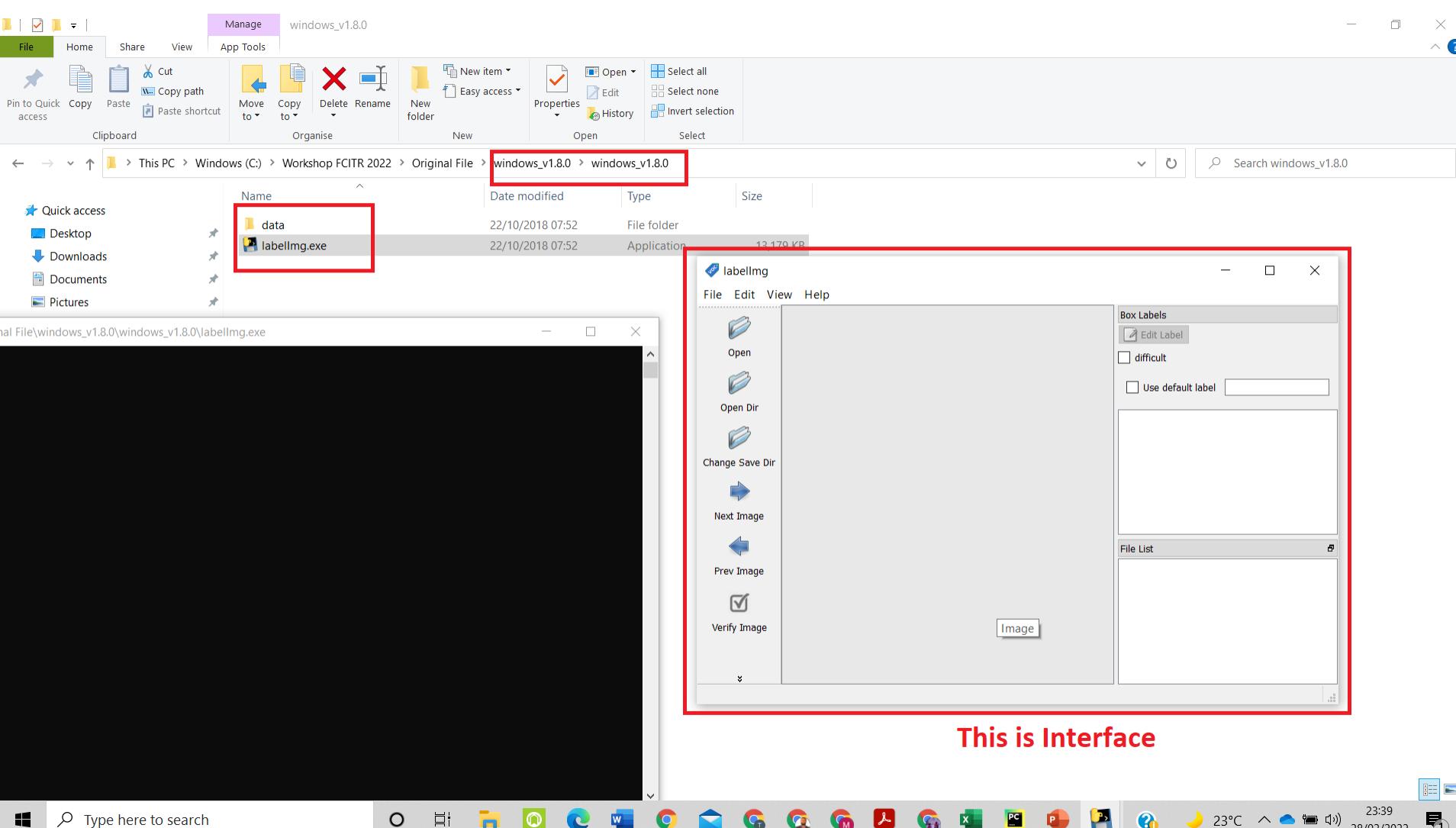
# Install LabelImg

<https://tzutalin.github.io/labelImg/>

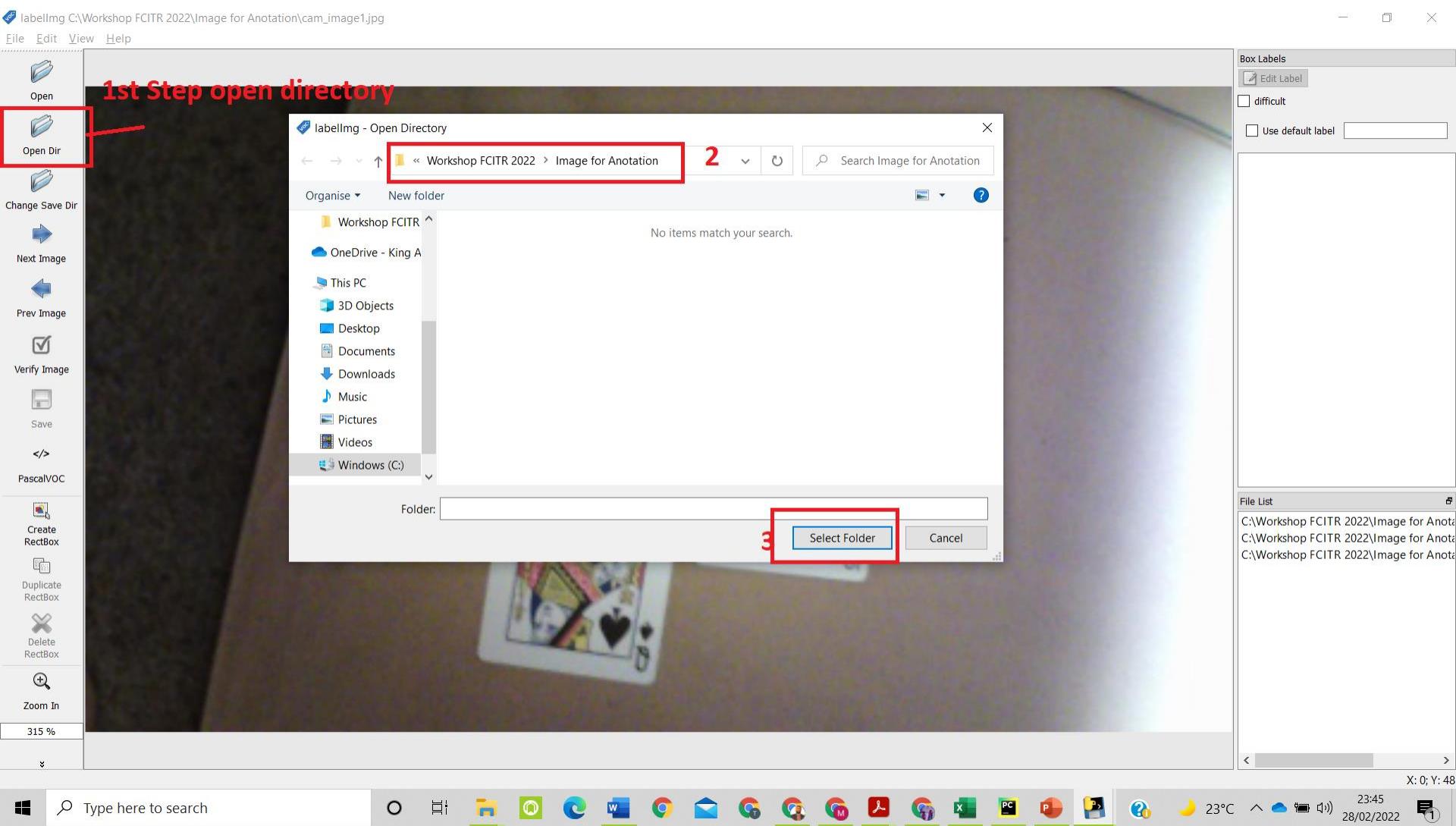
A screenshot of a web browser displaying a list of LabelImg releases for Windows. The browser's address bar shows the URL <https://tzutalin.github.io/labelImg/>. The page content lists twelve release links, each preceded by a blue bullet point. The releases are: Windows\_v1.3.3, Windows\_v1.3.4, Linux\_v1.3.3, Linux\_v1.3.4, Windows\_v1.4.0, Linux\_v1.4.0, Windows\_v1.4.3, Linux\_v1.4.3, Windows\_v1.5.0, Windows\_v1.5.1, Windows\_v1.5.2, Windows\_v1.6.0, Windows\_v1.6.1, Windows\_v1.7.0, and Windows\_v1.8.0. The last two items, Windows\_v1.7.0 and Windows\_v1.8.0, are circled with a red oval. A red arrow originates from the word "Download" and points towards this circled area.

- [Windows\\_v1.3.3](#)
- [Windows\\_v1.3.4](#)
- [Linux\\_v1.3.3](#)
- [Linux\\_v1.3.4](#)
- [Windows\\_v1.4.0](#)
- [Linux\\_v1.4.0](#)
- [Windows\\_v1.4.3](#)
- [Linux\\_v1.4.3](#)
- [Windows\\_v1.5.0](#)
- [Windows\\_v1.5.1](#)
- [Windows\\_v1.5.2](#)
- [Windows\\_v1.6.0](#)
- [Windows\\_v1.6.1](#)
- [Windows\\_v1.7.0](#)
- [Windows\\_v1.8.0](#)

# Install LabelImg



# Open Images directory for annotation LabelImg



# Annotation of images

labelImg C:\Workshop FCITR 2022\Image for Anotation\cam\_image1.jpg

File Edit View Help

Open  
Open Dir  
Change Save Dir  
Next Image  
Prev Image  
Verify Image  
Save  
</>  
PascalVOC  
Create RectBox  
Duplicate RectBox  
Delete RectBox  
Zoom In  
315 %  
▼

Change save format

1

this format we need

Box Labels

Edit Label  
 difficult  
 Use default label

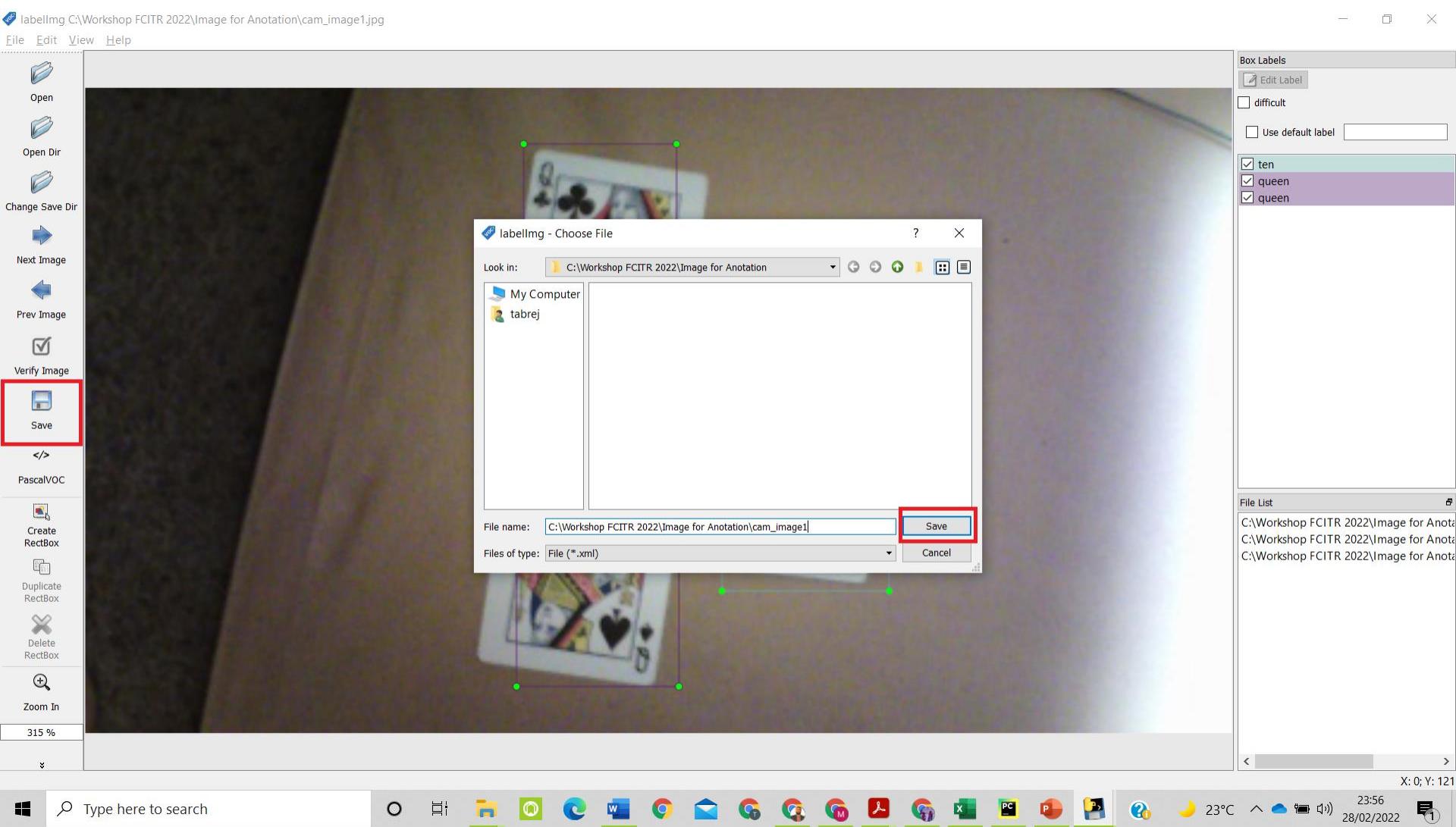
ten  
 queen  
 queen

File List

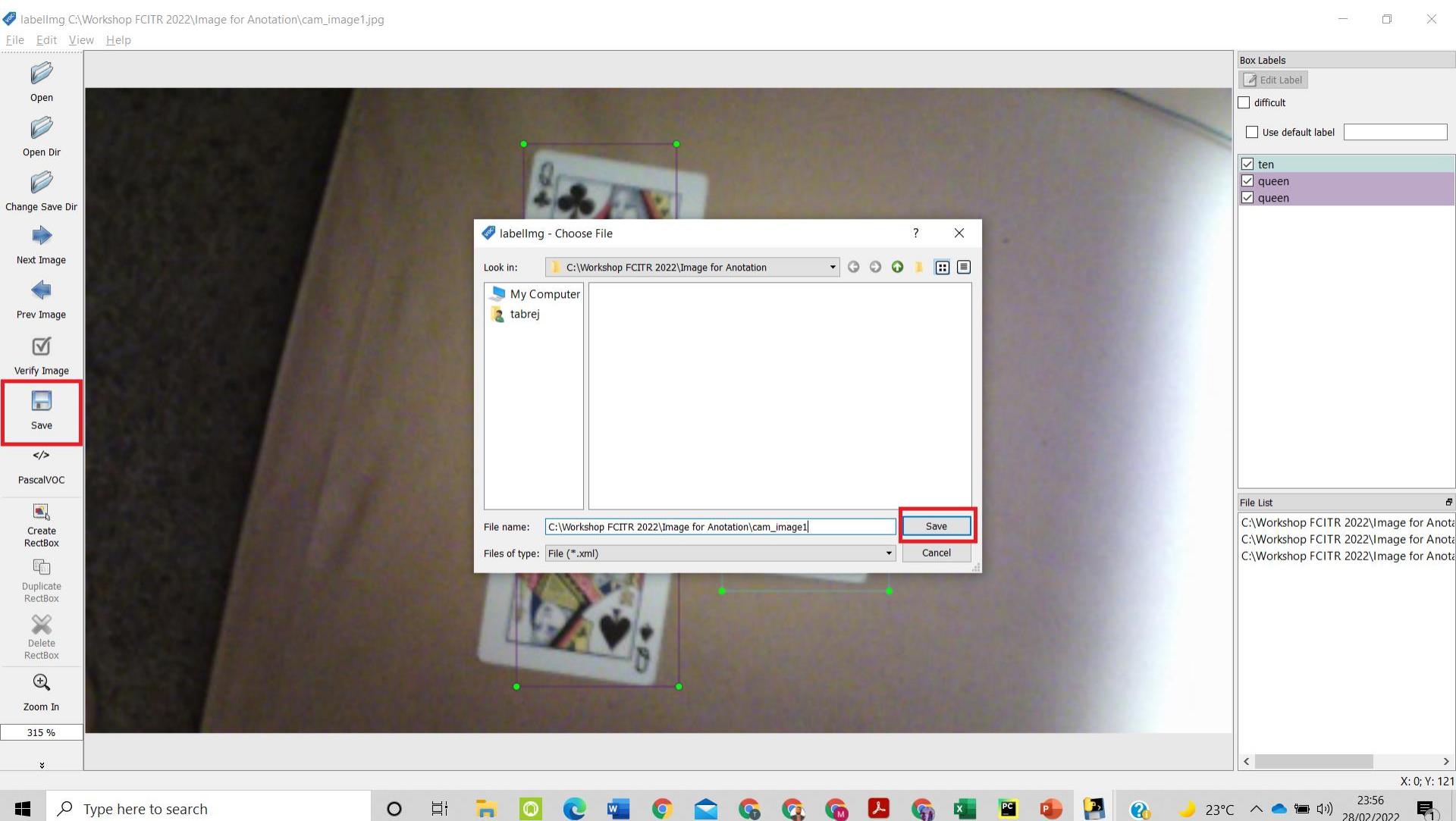
C:\Workshop FCITR 2022\Image for Anota  
C:\Workshop FCITR 2022\Image for Anota  
C:\Workshop FCITR 2022\Image for Anota

X: 0 Y: 162 23:50 23°C 28/02/2022

# Annotation of images



# Annotation of images

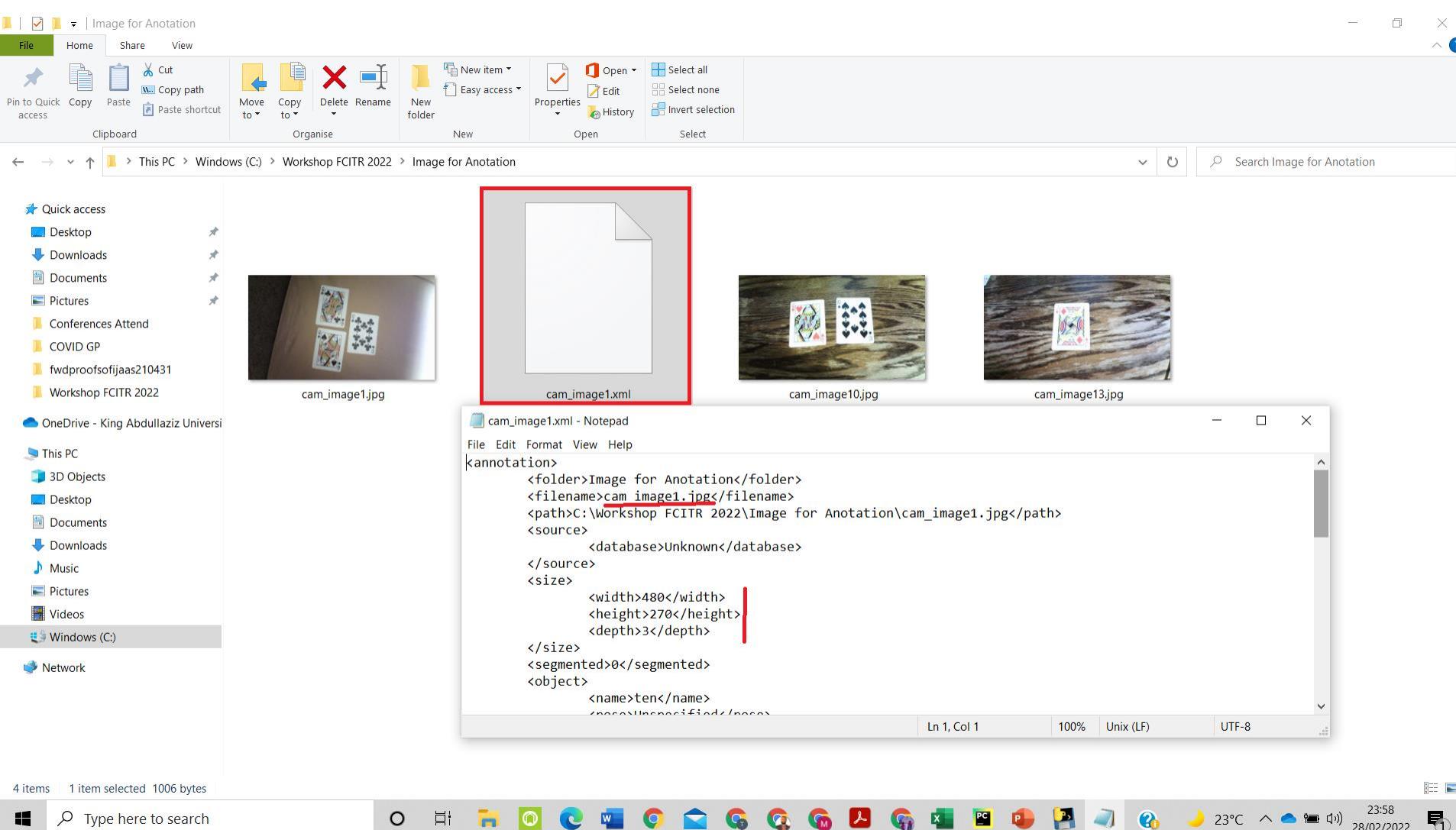


Type here to search



23:56  
23°C  
28/02/2022

# Annotation of images



# Annotation File

cam\_image1.xml - Notepad

File Edit Format View Help

```
</size>
<segmented>0</segmented>
<object>
    <name>ten</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <bndbox>
        <xmin>266</xmin>
        <ymin>96</ymin>
        <xmax>336</xmax>
        <ymax>210</ymax>
    </bndbox>
</object>
<object>
    <name>queen</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <bndbox>
        <xmin>180</xmin>
        <ymin>138</ymin>
        <xmax>248</xmax>
        <ymax>250</ymax>
    </bndbox>
</object>
<object>
    <name>queen</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <bndbox>
        <xmin>183</xmin>
        <ymin>23</ymin>
        <xmax>247</xmax>
        <ymax>132</ymax>
    </bndbox>
</object>
```

Ln 1, Col 1

100% Unix (LF)

UTF-8

# Check These file

Screenshot of a Windows File Explorer window showing a folder structure under "research".

The folder path is: This PC > Windows (C:) > Workshop FCITR 2022 > Workshop > models > research >

The "research" folder contains 77 items, displayed in a grid view:

- Items starting with "a": a3c\_blogpost, adv\_image\_net\_models, adversarial\_crypto, adversarial\_logit\_pairing, adversarial\_text, astronet, attention\_cr, audioset, autoaugment, autoencoder, brain\_coder, build, cognitive\_mapping\_and\_planning, cognitive\_planning, compression, cvt\_text, deep\_contextual\_bandits
- Items starting with "d": deep\_speech, deeplab, delf, differential\_privacy, dist, domain\_adaptation, efficient-hrl, faster\_rcnn, fivo, gan, global\_obj ectives, im2txt, images, inception, keypointnet, learned\_optimizer, learning\_to\_remember\_rare\_event s
- Items starting with "l": learning\_unsupervised\_learning, lexnet\_nc, lfads, lm\_1b, lm\_commo nsense, lstm\_object \_detection, marco, maskgan, minigo, morph\_net, namigner, neural\_gpu, neural\_pro grammmer, next\_frame \_prediction, nst\_blogpost, object\_dete ction, object\_dete ction.egg-i nfo, tensorrt
- Items ending in ".py": pcl\_rl, ptn, qa\_kg, real\_nvp, rebar, resnet, sentiment\_analysis, seq2specie s, skip\_thoug ht, slim, steve, street, struct2dept h, swivel, syntaxnet, tcn
- Other files: textsum, training, transformer, vid2depth, video\_predi ction, generate\_tf record.py, README.m d, setup.py, xml\_to\_csv. py

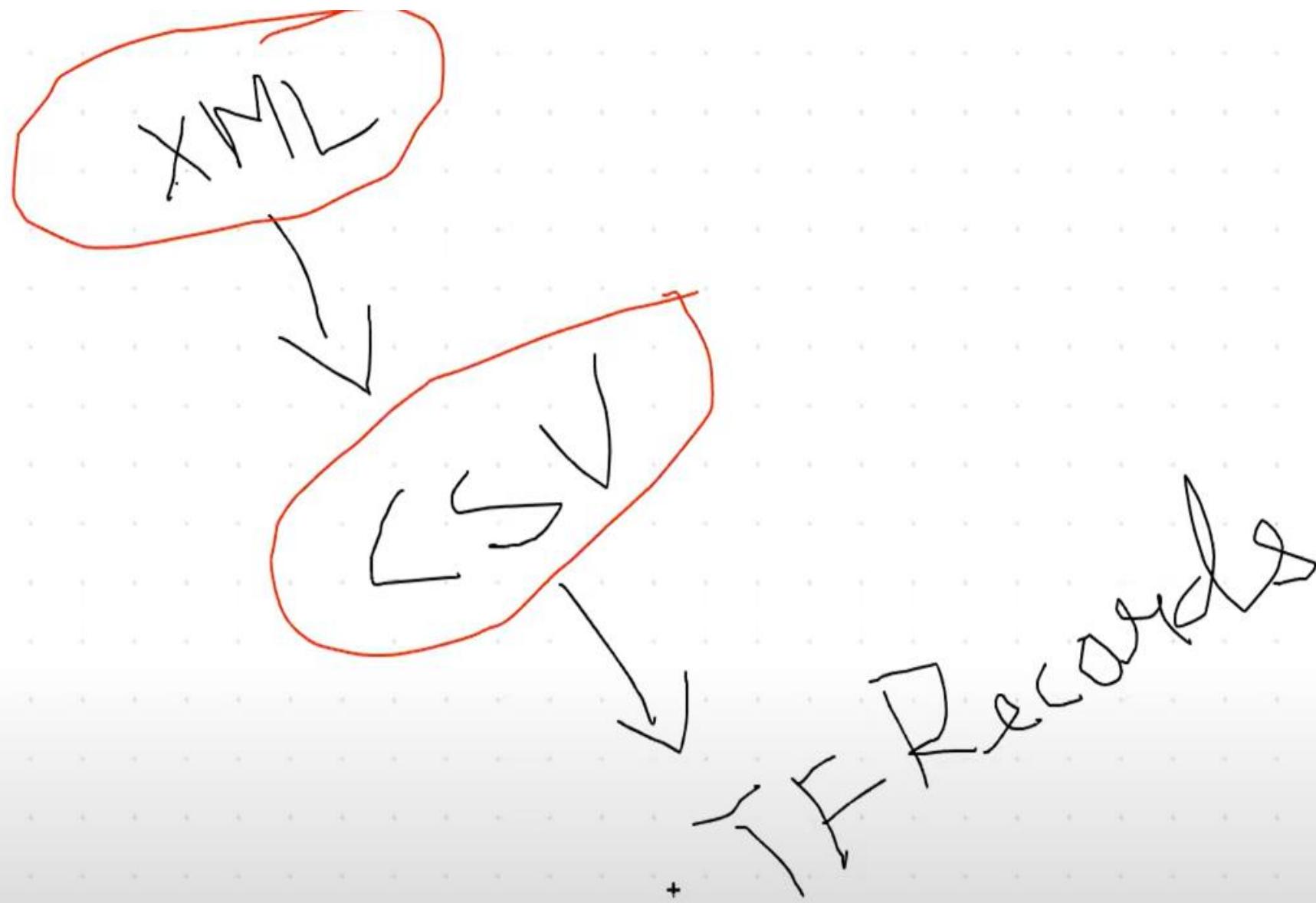
Two specific files are highlighted with red boxes: "faster\_rcnn" and "images".

File Explorer ribbon tabs: File, Home, Share, View.

File Explorer toolbar: Pin to Quick access, Copy, Paste, Cut, Copy path, Paste shortcut, Move to, Copy to, Delete, Rename, New folder, New item, Open, Properties, Select all, Select none, Invert selection.

File Explorer status bar: 77 items, Type here to search, 00:05, 01/03/2022, 23°C.

# Conversion of XML to CSV



# Conversion of XML to CSV

Anaconda Prompt (anaconda3)

```
06/02/2019  22:49      5,008 README.md
26/02/2022  01:32      <DIR>    real_nvp
26/02/2022  01:32      <DIR>    rebar
26/02/2022  01:32      <DIR>    resnet
26/02/2022  01:32      <DIR>    sentiment_analysis
26/02/2022  01:32      <DIR>    seq2species
06/02/2019  22:49      446 setup.py
26/02/2022  01:32      <DIR>    skip_thoughts
26/02/2022  01:32      <DIR>    slim
26/02/2022  01:32      <DIR>    steve
26/02/2022  01:32      <DIR>    street
26/02/2022  01:32      <DIR>    struct2depth
26/02/2022  01:32      <DIR>    swivel
26/02/2022  01:32      <DIR>    syntaxnet
26/02/2022  01:32      <DIR>    tcn
26/02/2022  01:32      <DIR>    tensorrt
26/02/2022  01:32      <DIR>    textsum
27/02/2022  15:48      <DIR>    training
26/02/2022  01:32      <DIR>    transformer
26/02/2022  01:32      <DIR>    vid2depth
26/02/2022  01:32      <DIR>    video_prediction
25/02/2020  01:49      1,164 xml_to_csv.py
        4 File(s)      10,307 bytes
       75 Dir(s)  46,854,643,712 bytes free
```

(tfod2) C:\Workshop FCITR 2022\Workshop\models\research>python xml\_to\_csv.py

# Conversion of XML to CSV

```
26/02/2022 01:32 <DIR>      slim
26/02/2022 01:32 <DIR>      steve
26/02/2022 01:32 <DIR>      street
26/02/2022 01:32 <DIR>      struct2depth
26/02/2022 01:32 <DIR>      swivel
26/02/2022 01:32 <DIR>      syntaxnet
26/02/2022 01:32 <DIR>      tcn
26/02/2022 01:32 <DIR>      tensorrt
26/02/2022 01:32 <DIR>      textsum
27/02/2022 15:48 <DIR>      training
26/02/2022 01:32 <DIR>      transformer
26/02/2022 01:32 <DIR>      vid2depth
26/02/2022 01:32 <DIR>      video_prediction
25/02/2020 01:49           1,164 xml_to_csv.py
                           4 File(s)      10,307 bytes
                           75 Dir(s)  46,854,643,712 bytes free
```

```
(tfod2) C:\Workshop FCITR 2022\Workshop\models\research>python xml_to_csv.py
Successfully converted xml to csv.
Successfully converted xml to csv.
```

```
(tfod2) C:\Workshop FCITR 2022\Workshop\models\research>
```

# Conversion of XML to CSV (Visual Confirmation)

File Home Share View

Pin to Quick access Copy Paste Cut Copy path Move to Copy to Delete Rename New folder New item Easy access Properties Open Select all Select none Invert selection

Clipboard Organise New Open Select AutoSave Off train\_labels.csv

File Home Insert Page Layout Formulas Data Review View Help

Quick access Desktop Downloads Documents Pictures Bandar form Tabrej Tabrej Documents Workshop FCITR 2022 OneDrive - King Abdullah University of Science and Technology This PC 3D Objects Desktop Documents Downloads Music Pictures Videos Windows (C) Network

test train test\_labels.csv train\_labels.csv

These F

	A	B	C	D	E	F	G	H	I
1	filename	width	height	class	xmin	ymin	xmax	ymax	
2	cam_imag	480	270	queen	173	24	260	137	
3	cam_imag	480	270	queen	165	135	253	251	
4	cam_imag	480	270	ten	255	96	337	208	
5	cam_imag	960	540	ten	501	116	700	353	
6	cam_imag	960	540	queen	261	124	453	370	
7	cam_imag	960	540	nine	225	96	490	396	
8	cam_imag	960	540	king	362	149	560	389	
9	cam_imag	960	540	jack	349	142	550	388	
10	cam_imag	960	540	jack	297	167	512	420	
11	cam_imag	960	540	ace	367	181	589	457	
12	cam_imag	960	540	ace	303	155	525	456	
13	cam_imag	960	540	ace	316	125	547	451	
14	cam_imag	960	540	ace	390	86	605	365	
15	cam_imag	960	540	jack	357	97	578	379	
16	cam_imag	960	540	queen	291	119	542	422	
17	cam_imag	960	540	queen	319	54	556	346	
18	cam_imag	960	540	queen	296	94	522	370	
19	cam_imag	960	540	queen	286	108	517	389	
20	cam_imag	960	540	king	444	113	685	411	
21	cam_imag	960	540	king	329	94	584	410	
22	cam_imag	960	540	king	350	89	587	383	

# Run the following to generate train and test records(.csv to TFRecord)

```
python generate_tfrecord.py --csv_input=images/train_labels.csv --  
image_dir=images/train --output_path=train.record
```

```
python generate_tfrecord.py --csv_input=images/test_labels.csv --  
image_dir=images/test --output_path=test.record
```

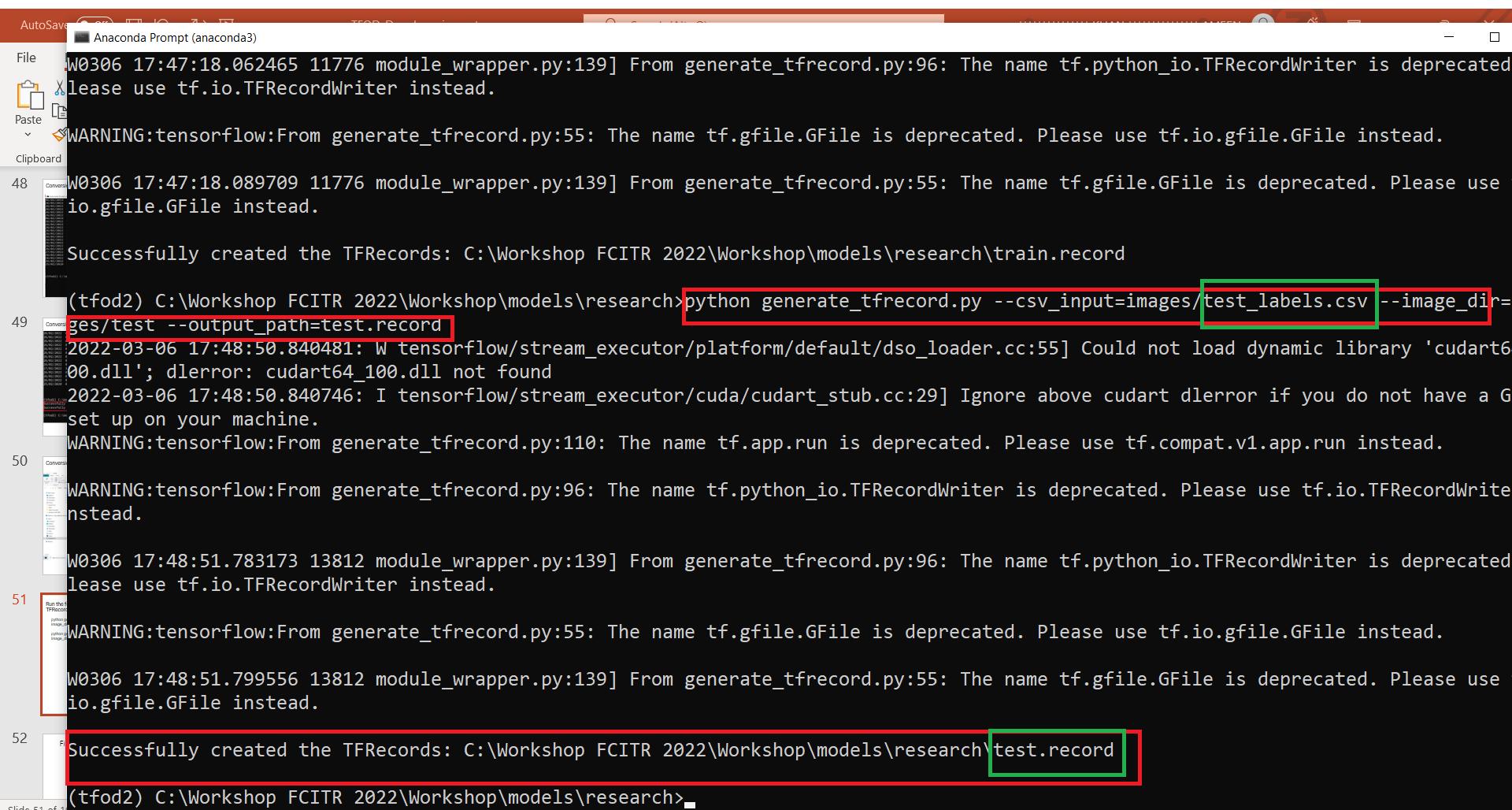
The screenshot shows a Windows taskbar at the bottom with various icons. Above it is a window titled "Anaconda Prompt (anaconda3)". The command line shows the following sequence of operations:

- Execution of `xml_to_csv.py` which successfully converts XML to CSV.
- Execution of `generate_tfrecord.py` with arguments: `--csv_input=images/train_labels.csv --image_dir=images/train --output_path=train.record`. This command generates TFRecords and outputs the path: `C:\Workshop FCITR 2022\Workshop\models\research\train.record`.
- Execution of `generate_tfrecord.py` with arguments: `--csv_input=images/test_labels.csv --image_dir=images/test --output_path=test.record`. This command generates TFRecords and outputs the path: `C:\Workshop FCITR 2022\Workshop\models\research\test.record`.

Red boxes highlight the command line arguments and the output paths from the terminal output.

# Run the following to generate train and test records(.csv to TFRecord)

```
python generate_tfrecord.py --csv_input=images/test_labels.csv --  
image_dir=images/test --output_path=test.record
```



The screenshot shows a terminal window titled "Anaconda Prompt (anaconda3)". The command entered was:

```
(tfod2) C:\Workshop FCITR 2022\Workshop\models\research>python generate_tfrecord.py --csv_input=images/test_labels.csv --image_dir=images/test --output_path=test.record
```

The output indicates that the command was successful:

```
W0306 17:47:18.062465 11776 module_wrapper.py:139] From generate_tfrecord.py:96: The name tf.python_io.TFRecordWriter is deprecated.  
Please use tf.io.TFRecordWriter instead.  
W0306 17:47:18.089709 11776 module_wrapper.py:139] From generate_tfrecord.py:55: The name tf.gfile.GFile is deprecated. Please use tf.io.gfile.GFile instead.  
Successfully created the TFRecords: C:\Workshop FCITR 2022\Workshop\models\research\train.record  
W0306 17:48:51.783173 13812 module_wrapper.py:139] From generate_tfrecord.py:96: The name tf.python_io.TFRecordWriter is deprecated.  
Please use tf.io.TFRecordWriter instead.  
W0306 17:48:51.799556 13812 module_wrapper.py:139] From generate_tfrecord.py:55: The name tf.gfile.GFile is deprecated. Please use tf.io.gfile.GFile instead.  
Successfully created the TFRecords: C:\Workshop FCITR 2022\Workshop\models\research\test.record
```

Annotations highlight the command line arguments: `--csv_input=images/test_labels.csv`, `--image_dir=images/test`, and `--output_path=test.record`. The output message "Successfully created the TFRecords" is also highlighted.

# generate\_tfrecord.py file

C:\Workshop FCITR 2022\Workshop\models\research\generate\_tfrecord.py - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

generate\_tfrecord.py

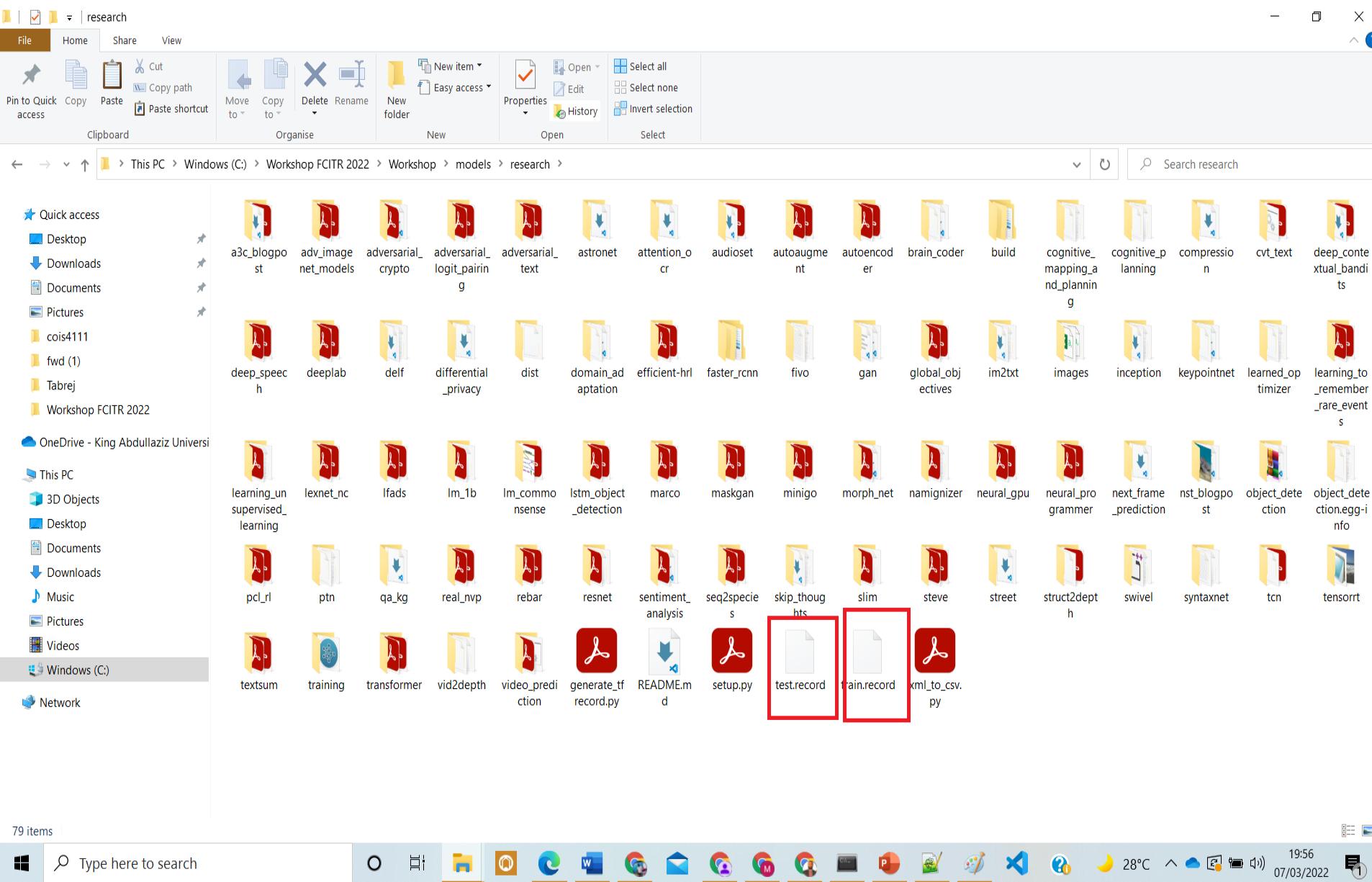
```
7 # Create test data:
8 python generate_tfrecord.py --csv_input=images/test_labels.csv --image_dir=images/test --output_path=test.record
9 """
10 from __future__ import division
11 from __future__ import print_function
12 from __future__ import absolute_import
13
14 import os
15 import io
16 import pandas as pd
17 import tensorflow as tf
18
19 from PIL import Image
20 from object_detection.utils import dataset_util
21 from collections import namedtuple, OrderedDict
22
23 flags = tf.app.flags
24 flags.DEFINE_string('csv_input', '', 'Path to the CSV input')
25 flags.DEFINE_string('image_dir', '', 'Path to the image directory')
26 flags.DEFINE_string('output_path', '', 'Path to output TFRecord')
27 FLAGS = flags.FLAGS
28
29
30 # TO-DO replace this with label map
31 def class_text_to_int(row_label):
32     if row_label == 'nine':
33         return 1
34     elif row_label == 'ten':
35         return 2
36     elif row_label == 'jack':
37         return 3
38     elif row_label == 'queen':
39         return 4
40     elif row_label == 'king':
41         return 5
42     elif row_label == 'ace':
43         return 6
44     else:
45         None
46
47
48 def split(df, group):
```

If you use custom images mentioned the name of class name (like car, Bus)

length : 3,689 lines : 111 Ln: 18 Col: 1 Pos: 498 Unix (LF) UTF-8 07/03/2022 19:45

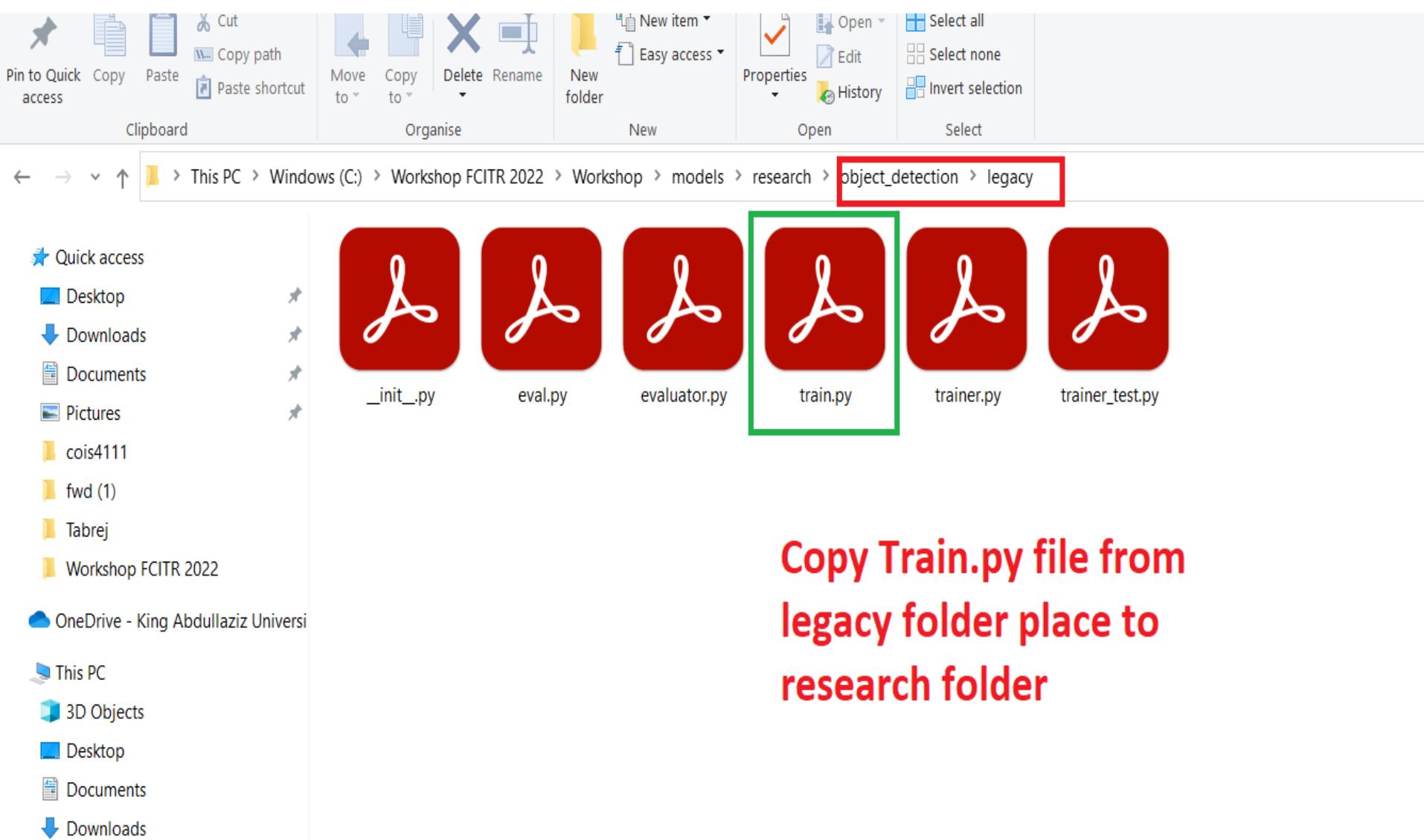
Type here to search

# test.record, train.record file generated in directory



# Start the Training Process

Copy train.py file from object\_detection>legacy folder place to research folder



# Explore the pretrained model as we install **faster\_rcnn**

[Detection/detection\\_model\\_zoo.md at master · librafacebook/Detection · GitHub](https://github.com/librafacebook/Detection/blob/master/object_detection/g3doc/detection_model_zoo.md)

The screenshot shows a web browser window with multiple tabs open. The active tab is displaying the GitHub page for the 'detection\_model\_zoo.md' file. The page title is 'COCO-trained models'. A table lists 15 different models, each with its name, speed in milliseconds, COCO mAP score, and the type of output (Boxes). The 'faster\_rcnn' model is listed at the bottom of the table.

Model name	Speed (ms)	COCO mAP <sup>[1]</sup>	Outputs
<a href="#">ssd_mobilenet_v1_coco</a>	30	21	Boxes
<a href="#">ssd_mobilenet_v1_0.75_depth_coco</a> ☆	26	18	Boxes
<a href="#">ssd_mobilenet_v1_quantized_coco</a> ☆	29	18	Boxes
<a href="#">ssd_mobilenet_v1_0.75_depth_quantized_coco</a> ☆	29	16	Boxes
<a href="#">ssd_mobilenet_v1_ppn_coco</a> ☆	26	20	Boxes
<a href="#">ssd_mobilenet_v1_fpn_coco</a> ☆	56	32	Boxes
<a href="#">ssd_resnet_50_fpn_coco</a> ☆	76	35	Boxes
<a href="#">ssd_mobilenet_v2_coco</a>	31	22	Boxes
<a href="#">ssd_mobilenet_v2_quantized_coco</a>	29	22	Boxes
<a href="#">ssdlite_mobilenet_v2_coco</a>	27	22	Boxes
<a href="#">ssd_inception_v2_coco</a>	42	24	Boxes
<a href="#">faster_rcnn_inception_v2_coco</a>	58	28	Boxes
<a href="#">faster_rcnn_resnet50_coco</a>	89	30	Boxes
<a href="#">faster_rcnn_resnet50_lowproposals_coco</a>	64		Boxes
<a href="#">rfcn_resnet101_coco</a>	92	30	Boxes

Slide 56 of 113    English (United Kingdom)    Accessibility: Investigate    Notes    21:48    07/03/2022

# These pretrained model is trained on COCO datasets (90 classes)

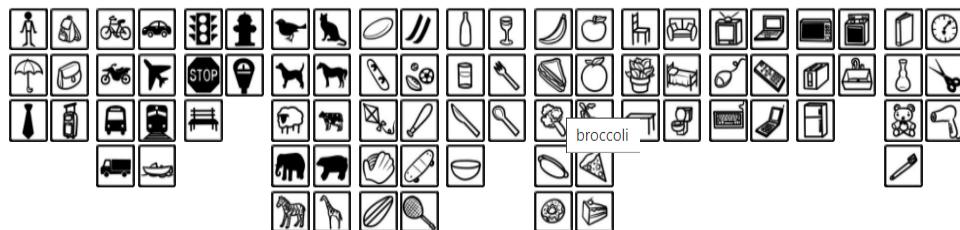
Screenshot of a web browser showing the COCO Explorer interface.

The browser address bar shows: [cocodataset.org/#explore](https://cocodataset.org/#explore)

The page header includes the COCO logo, navigation links (Home, People, Dataset, Tasks, Evaluate), and an email link: [info@cocodataset.org](mailto:info@cocodataset.org).

## COCO Explorer

COCO 2017 train/val browser (123,287 images, 886,284 instances). Crowd labels not shown.



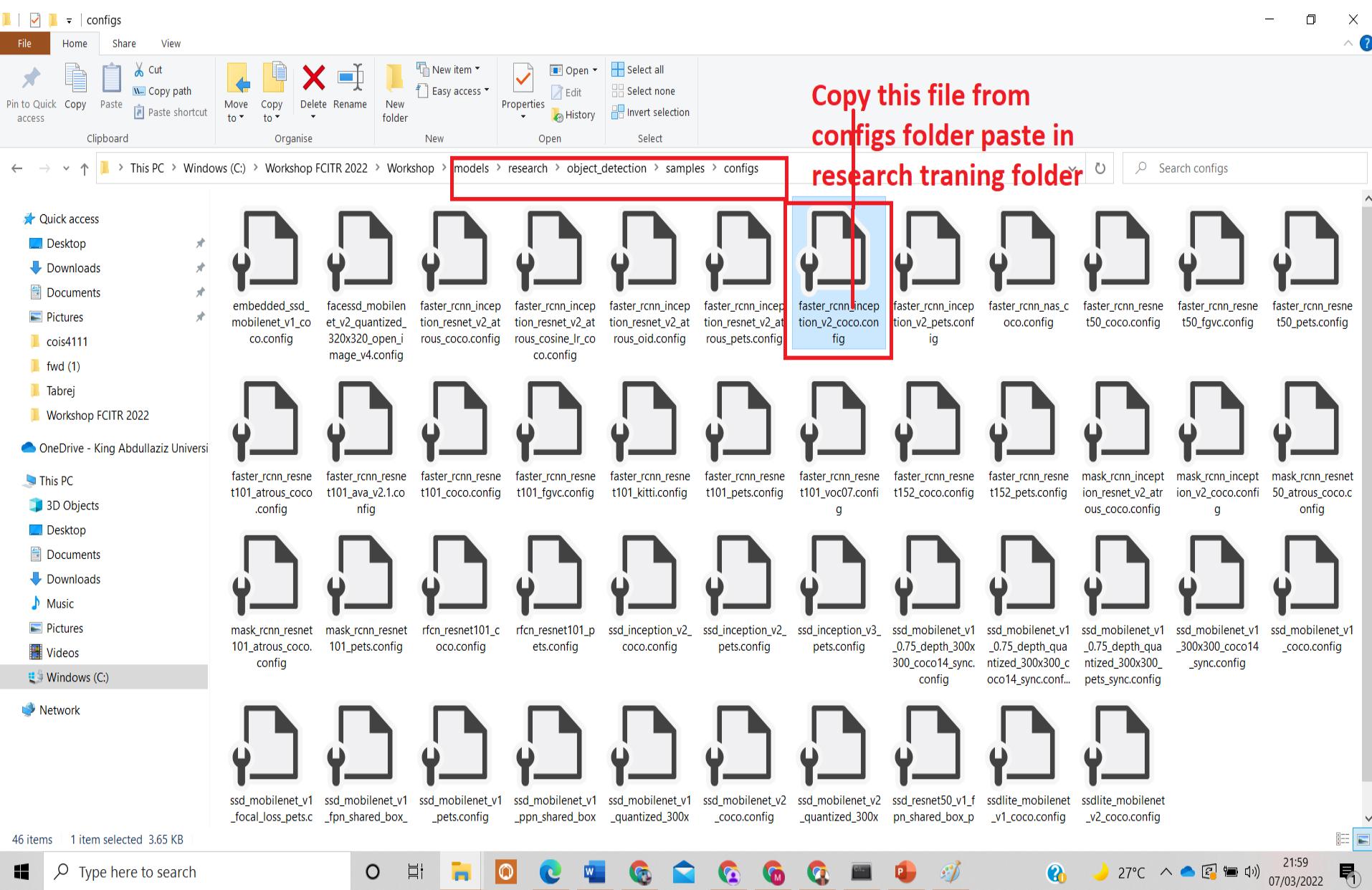
A search bar with a "search" button is located below the icon grid.

Page footer links: [Github Page Source](#), [Terms of Use](#).

Windows taskbar at the bottom:

- Type here to search
- Icons for File Explorer, Edge, File Manager, Word, Chrome, Mail, People, Paint 3D, and File History.
- System tray icons: Battery (21:49, 07/03/2022), Network, Volume, and Task View.

# Copy the config file from object\_detection>samples>configs folder



# Paste file in research>training folder

The screenshot shows a Windows File Explorer window with the following directory path:

- File
- Home
- Share
- View
- Pin to Quick access
- Copy
- Paste
- Move to
- Copy to
- Delete
- Rename
- New folder
- New item
- Easy access
- Properties
- Open
- Select all
- Select none
- Invert selection

Clipboard

Organise

New

Open

Search training

Name Date modified Type Size

Name	Date modified	Type	Size
faster_rcnn_inception_v2_coco.config	06/02/2019 23:49	XML Configuration...	4 KB
labelmap.pbtxt	25/02/2020 01:49	PBTXT File	1 KB

Quick access

- Desktop
- Downloads
- Documents
- Pictures
- cois4111
- fwd (1)
- Tabrej
- Workshop FCITR 2022

OneDrive - King Abdullah University of Science and Technology

This PC

- 3D Objects
- Desktop
- Documents
- Downloads
- Music
- Pictures
- Videos

Windows (C)

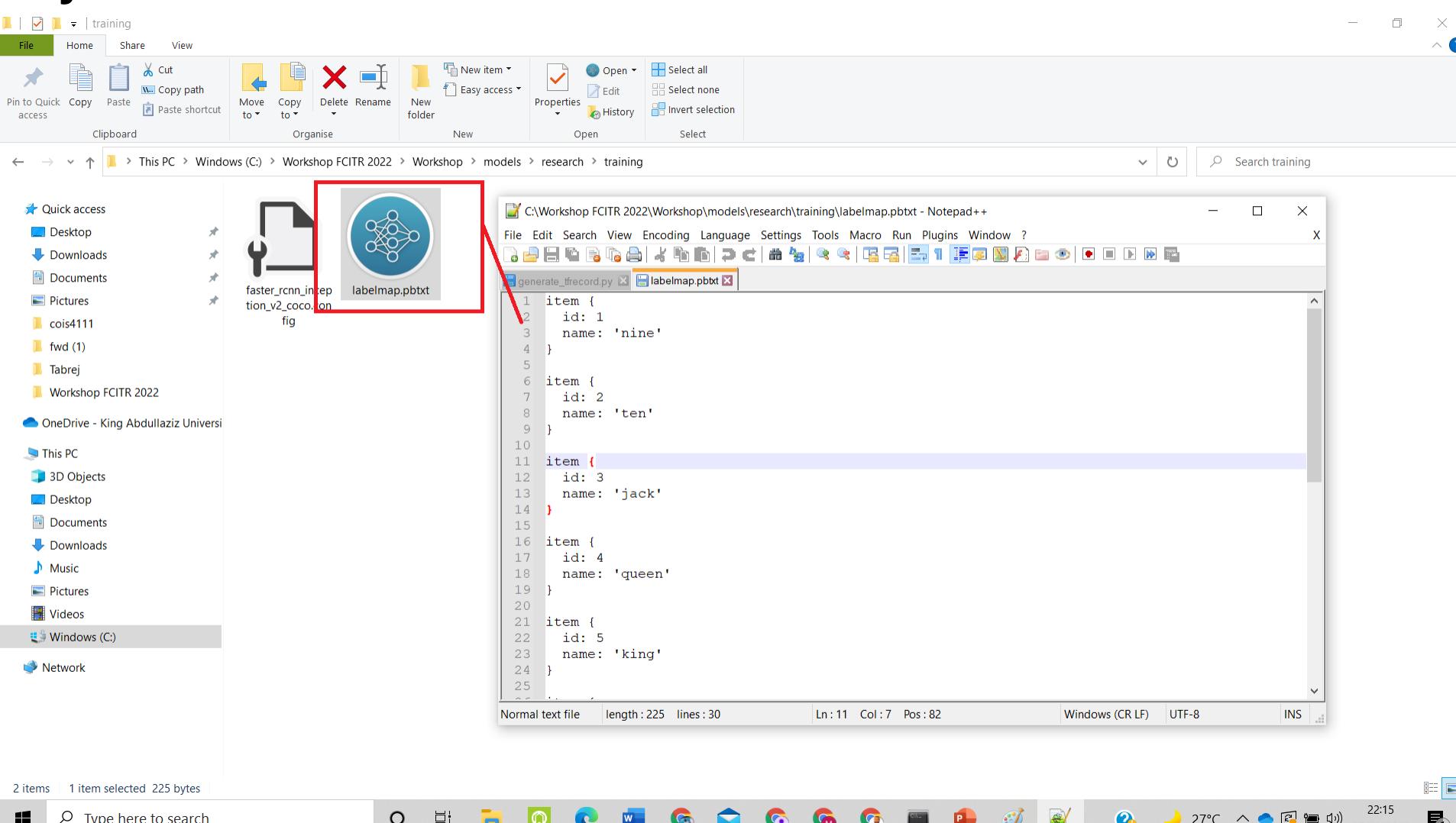
Network

2 items 1 item selected 3.65 KB

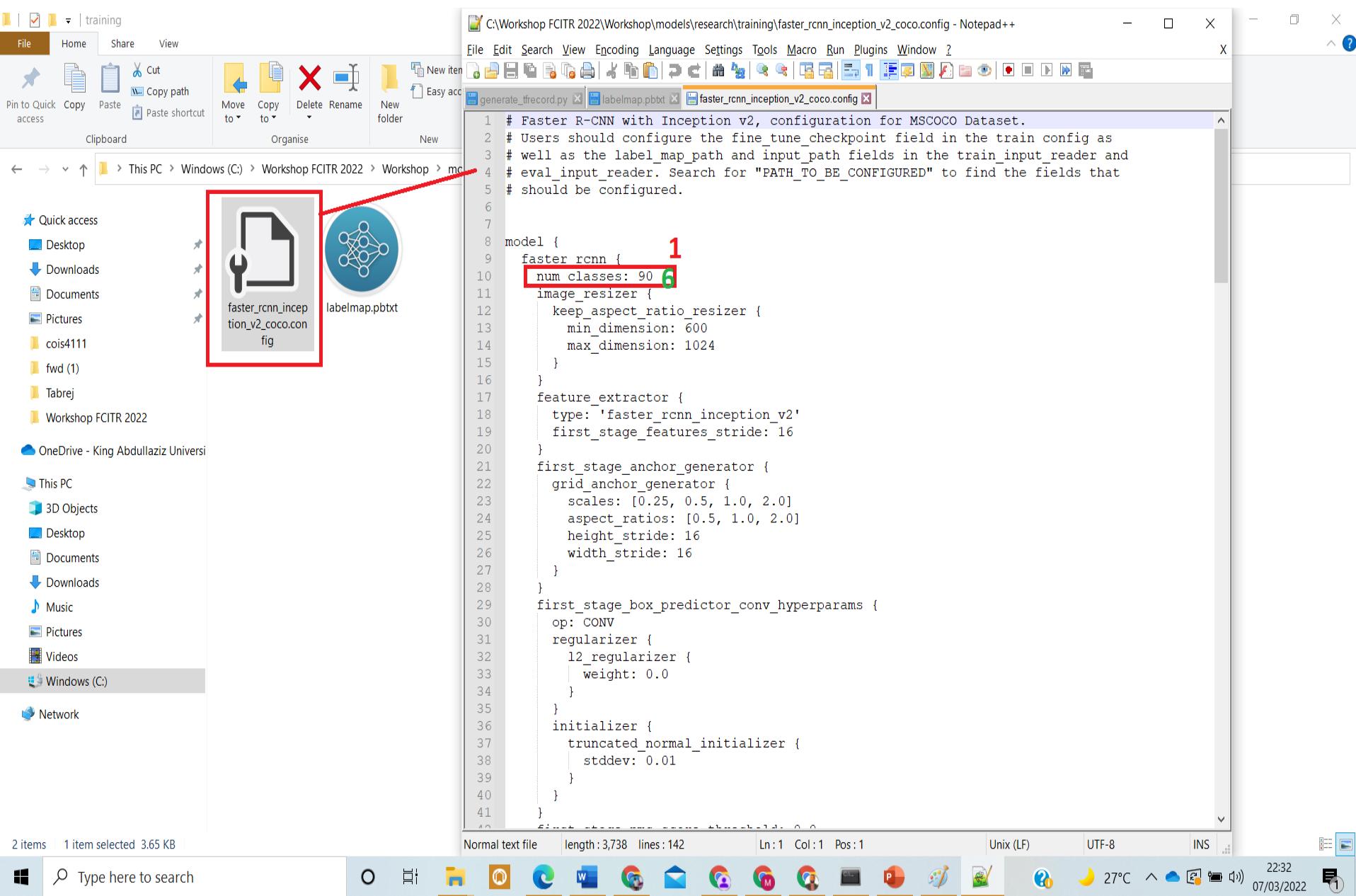
Paste this file in training folder

22:01 07/03/2022

# Labelmap.pbtxt file have all classes similarly file you can find in object\_detection>data



# Changes 6 places faster\_rcnn\_inception\_v2\_coco.config in file



The screenshot shows a Windows File Explorer window on the left and a Notepad++ window on the right. The File Explorer window displays two files: 'faster\_rcnn\_inception\_v2\_coco.config' and 'labelmap.pbtxt'. The 'faster\_rcnn\_inception\_v2\_coco.config' file is highlighted with a red box and has a red arrow pointing to the Notepad++ window. The Notepad++ window shows the contents of the configuration file. Line 10 contains the line 'num\_classes: 90'. This line is also highlighted with a red box. Above the opening brace of the 'model' block, the number '1' is displayed. Inside the red box around 'num\_classes: 90', the number '6' is displayed.

```
# Faster R-CNN with Inception v2, configuration for MSCOCO Dataset.  
# Users should configure the fine_tune_checkpoint field in the train config as  
# well as the label_map_path and input_path fields in the train_input_reader and  
# eval_input_reader. Search for "PATH_TO_BE_CONFIGURED" to find the fields that  
# should be configured.  
  
model {  
    faster_rcnn {  
        num_classes: 90 1  
        image_resizer {  
            keep_aspect_ratio_resizer {  
                min_dimension: 600  
                max_dimension: 1024  
            }  
        }  
        feature_extractor {  
            type: 'faster_rcnn_inception_v2'  
            first_stage_features_stride: 16  
        }  
        first_stage_anchor_generator {  
            grid_anchor_generator {  
                scales: [0.25, 0.5, 1.0, 2.0]  
                aspect_ratios: [0.5, 1.0, 2.0]  
                height_stride: 16  
                width_stride: 16  
            }  
        }  
        first_stage_box_predictor_conv_hyperparams {  
            op: CONV  
            regularizer {  
                l2_regularizer {  
                    weight: 0.0  
                }  
            }  
            initializer {  
                truncated_normal_initializer {  
                    stddev: 0.01  
                }  
            }  
        }  
    }  
}
```

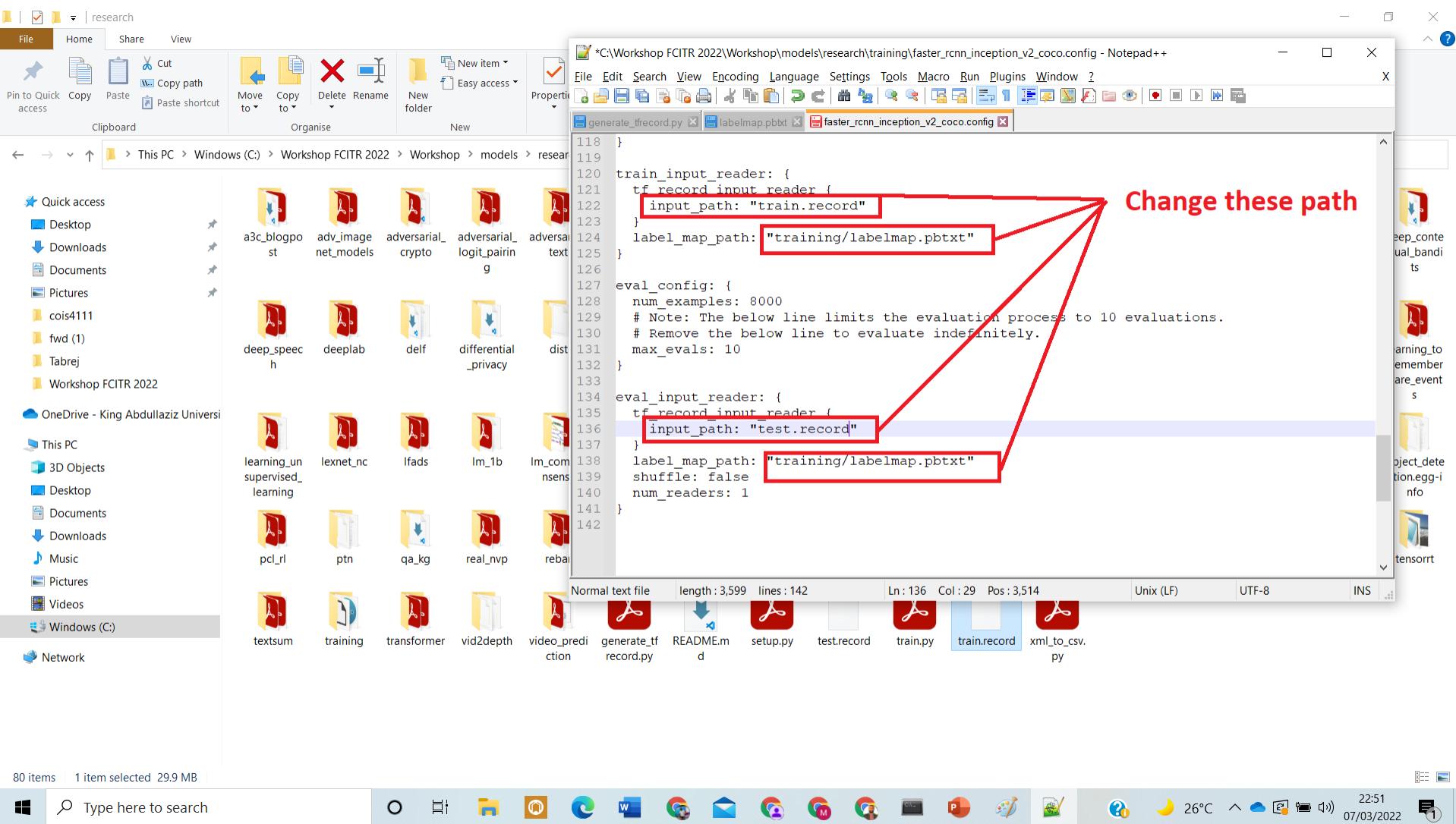
# Change line 100 faster\_rcnn folder name as path

The screenshot shows a Windows desktop environment. On the left, a File Explorer window is open with the path 'This PC > Windows (C) > Workshop FCITR 2022 > Workshop > mode'. Inside the 'mode' folder, there is a subfolder named 'faster\_rcnn' containing several files: 'saved\_model', 'checkpoint', 'frozen\_inference\_graph.pb', 'model.ckpt.data-00000-of-00001', 'model.ckpt.index', 'model.ckpt.meta', and 'pipeline.config'. To the right of the File Explorer is a Notepad++ application window. The title bar of Notepad++ says '\*C:\Workshop FCITR 2022\Workshop\models\research\training\faster\_rcnn\_inception\_v2\_coco.config - Notepad++'. The main text area of Notepad++ displays a configuration file with multiple lines of code. Line 107 of the file is highlighted with a red circle and a red rectangle around the text 'faster\_rcnn/model.ckpt'. The Notepad++ status bar at the bottom shows 'Normal text file length: 3,728 lines : 142 Ln : 107 Col : 37 Pos : 2,693 Unix (LF) UTF-8 INS'.

```
momentum_optimizer: {
    learning_rate: {
        manual_step_learning_rate {
            initial_learning_rate: 0.0002
            schedule {
                step: 900000
                learning_rate: .00002
            }
            schedule {
                step: 1200000
                learning_rate: .000002
            }
        }
        momentum_optimizer_value: 0.9
    }
    use_moving_average: false
}
gradient_clipping_by_norm: 10.0
107 fine_tune_checkpoint: "faster_rcnn/model.ckpt"
from_detection_checkpoint: true
# Note: The below line limits the training process to 200K steps, which we
# empirically found to be sufficient enough to train the COCO dataset. This
# effectively bypasses the learning rate schedule (the learning rate will
# never decay). Remove the below line to train indefinitely.
num_steps: 200000
data_augmentation_options {
    random_horizontal_flip {
```

File Home Share View  
Pin to Quick access Cut Copy path Paste Move to Copy to Delete Rename New folder New item Open Select all  
Clipboard Organise New  
File Edit Search Encoding Language Settings Tools Macro Run Plugins Window ?  
generate\_tfrecord.py labelmap.pbtxt faster\_rcnn\_inception\_v2\_coco.config  
Normal text file length: 3,728 lines : 142 Ln : 107 Col : 37 Pos : 2,693 Unix (LF) UTF-8 INS  
Type here to search 22:46 26°C 07/03/2022

# Change path of train.record and label path also test.record



# Change the number of steps for less training but generally do for 2000

The screenshot shows a Windows desktop environment. On the left is a file explorer window showing a folder structure under 'Windows (C)'. In the center is a Notepad++ window displaying a configuration file named 'faster\_rcnn\_inception\_v2\_coco.config'. The file contains several learning rate schedules and optimizer settings. A specific line, 'num\_steps: 250', is highlighted with a red box. This line is part of a section that limits the training process to 200K steps. The Notepad++ status bar indicates the file is a 'Normal text file' with a length of 3,596 bytes, 142 lines, and line 113.

```
initial_learning_rate: 0.0002
schedule {
    step: 900000
    learning_rate: .00002
}
schedule {
    step: 1200000
    learning_rate: .000002
}
momentum_optimizer_value: 0.9
use_moving_average: false
}
gradient_clipping_by_norm: 10.0
fine_tune_checkpoint: "faster_rcnn/model.ckpt"
from_detection_checkpoint: true
# Note: The below line limits the training process to 200K steps, which we
# empirically found to be sufficient enough to train the COCO dataset. This
# effectively bypasses the learning rate schedule (the learning rate will
# never decay). Remove the below line to train indefinitely.
num_steps: 250
data_augmentation_options {
    random_horizontal_flip {
}
```

At the bottom of the screen is the taskbar, which includes a search bar, pinned application icons (File Explorer, Edge, Word, Excel, etc.), and system status indicators (date, time, battery level).

# NOW Run the following command from the research folder. This will start the training in your local system

```
python train.py --logtostderr --train_dir=training/ --  
pipeline_config_path=training/faster_rcnn_inception_v2_coco.config
```

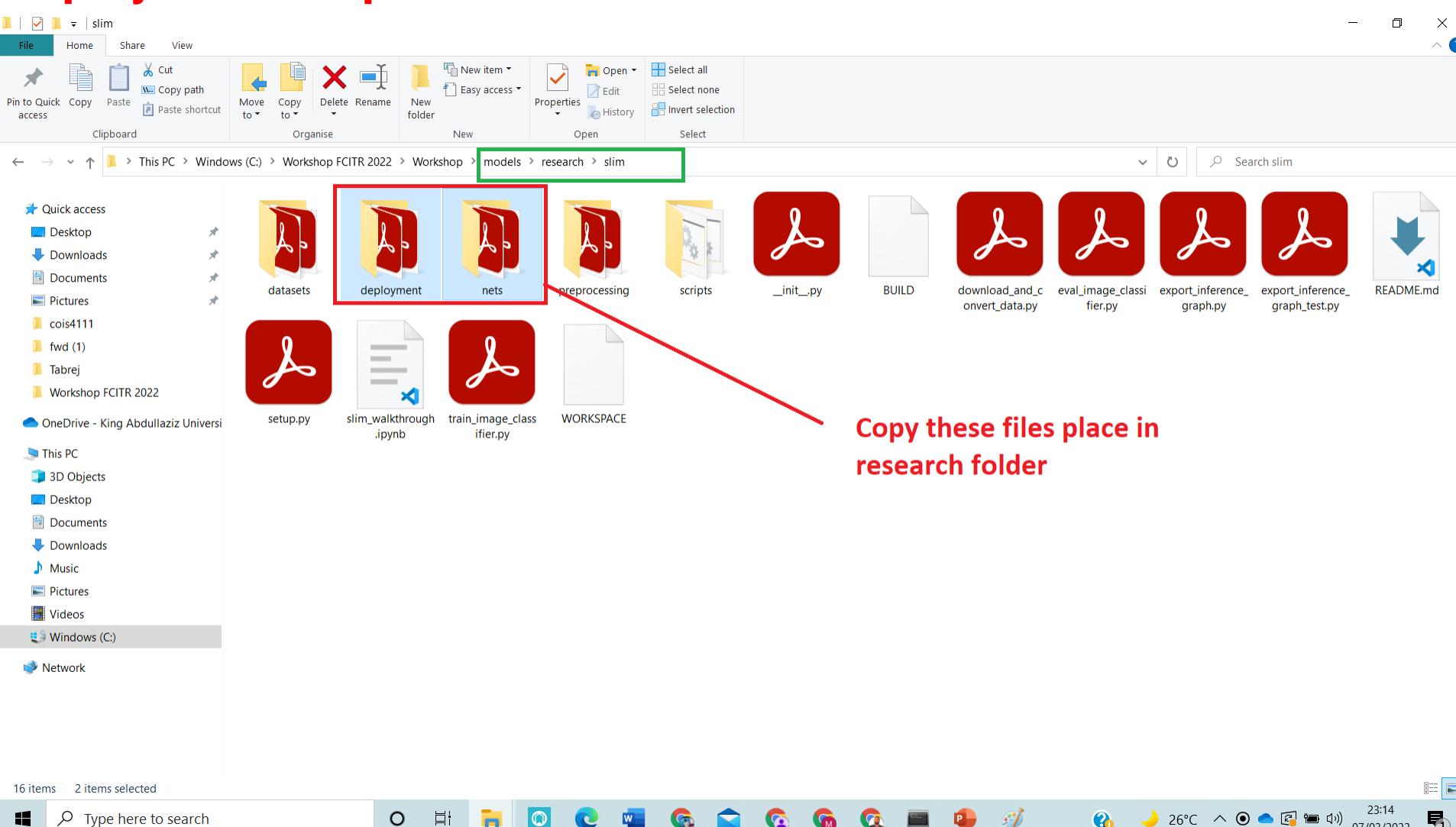
```
Anaconda Prompt (anaconda3)  
WARNING:tensorflow:From generate_tfrecord.py:55: The name tf.gfile.GFile is deprecated. Please use tf.io.gfile.GFile instead.  
W0306 17:48:51.799556 13812 module_wrapper.py:139] From generate_tfrecord.py:55: The name tf.gfile.GFile is deprecated. Please use tf.io.gfile.GFile instead.  
Successfully created the TFRecords: C:\Workshop FCITR 2022\Workshop\models\research\test.record  
(tfod2) C:\Workshop FCITR 2022\Workshop\models\research>python train.py --logtostderr --train_dir=training/ --pipeline_config_path=tra  
ining/faster_rcnn_inception_v2_coco.config  
2022-03-07 23:09:00.811291: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not load dynamic library 'cudart64_1  
00.dll'; dlerror: cudart64_100.dll not found  
2022-03-07 23:09:00.811433: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you do not have a GPU  
set up on your machine.  
WARNING:tensorflow:  
The TensorFlow contrib module will not be included in TensorFlow 2.0.  
For more information, please see:  
* https://github.com/tensorflow/community/blob/master/rfcs/20180907-contrib-sunset.md  
* https://github.com/tensorflow/addons  
* https://github.com/tensorflow/io \(for I/O related ops\)  
If you depend on functionality not listed there, please file an issue. This error will get  
  
Traceback (most recent call last):  
  File "train.py", line 51, in <module>  
    from object_detection.builders import model_builder  
  File "C:\Workshop FCITR 2022\Workshop\models\research\object_detection\builders\model_builder.py", line 35, in <module>  
    from object_detection.models import faster_rcnn_inception_resnet_v2_feature_extractor as frcnn_inc_res  
  File "C:\Workshop FCITR 2022\Workshop\models\research\object_detection\models\faster_rcnn_inception_resnet_v2_feature_extractor.py",  
line 28, in <module>  
    from nets import inception_resnet_v2  
ModuleNotFoundError: No module named 'nets'  
(tfod2) C:\Workshop FCITR 2022\Workshop\models\research>
```

run this cmd for training

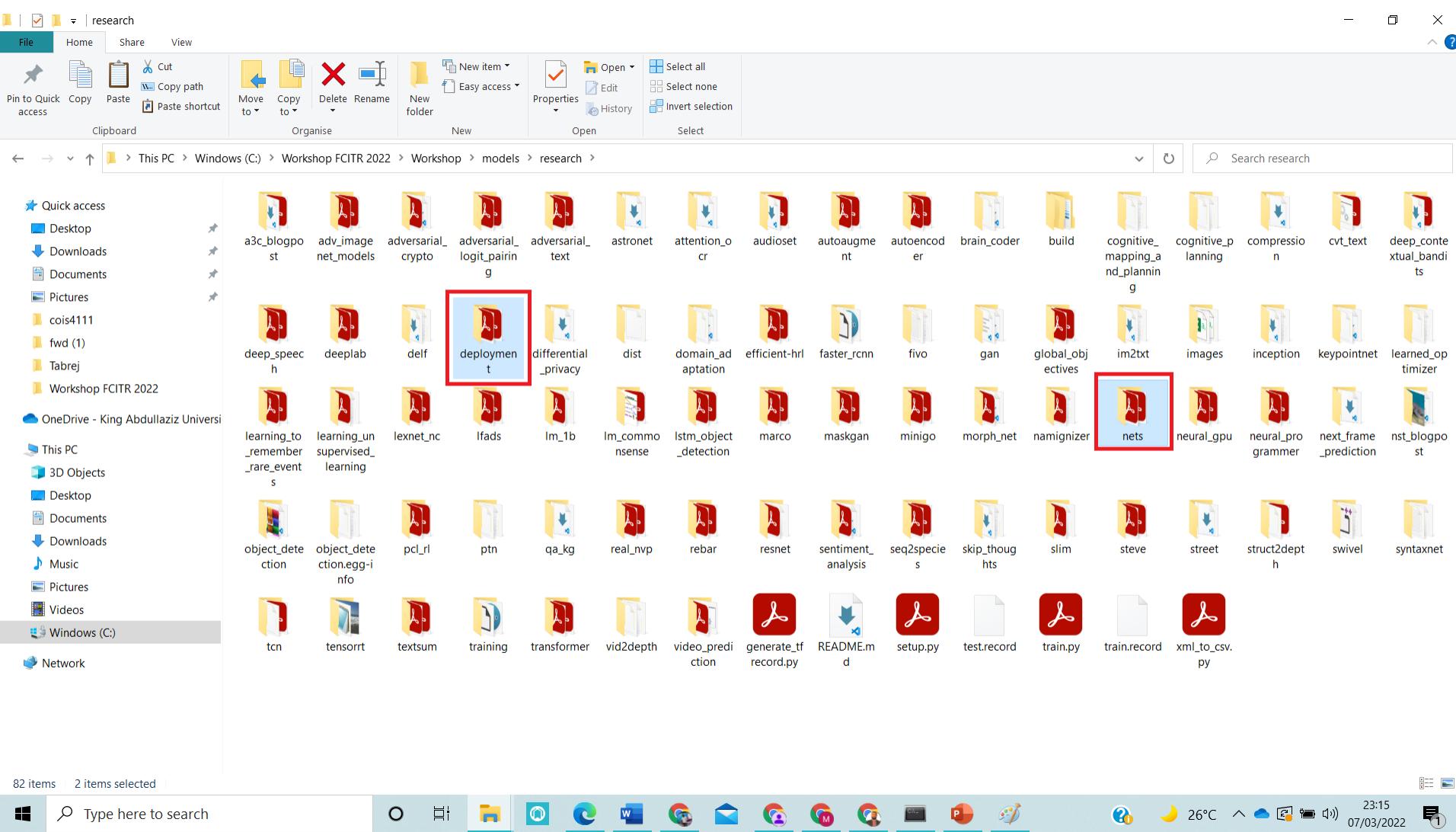
This error will get



# To solve this error no module name net copy nets file and deployment file paste in research folder



# File in research folder



# Error will come due to numpy version

stackoverflow.com/questions/58479556/notimplementederror-cannot-convert-a-symbolic-tensor-2nd-target0-to-a-numpy

NotImplementedError: Cannot convert a symbolic Tensor (2nd\_target:0) to a numpy array

Asked 2 years, 4 months ago Active 1 month ago Viewed 124k times

I try to pass 2 loss functions to a model as [Keras allows that](#).

loss: String (name of objective function) or objective function or Loss instance. See losses. If the model has multiple outputs, you can use a different loss on each output by **passing a dictionary or a list of losses**. The loss value that will be minimized by the model will then be the sum of all individual losses.

The two loss functions:

```
def l_2nd(beta):
    def loss_2nd(y_true, y_pred):
        ...
        return K.mean(t)

    return loss_2nd
```

and

```
def l_1st(alpha):
    def loss_1st(y_true, y_pred):
        ...
```

may be this error will come

The Overflow Blog

- Stop aggregating away the signal in your data

Featured on Meta

- Stack Exchange Q&A access will not be restricted in Russia
- Planned maintenance scheduled for Friday, March 18th, 00:30-2:00 UTC...
- Should we burninate the [high-resolution] tag?
- Improving the first-time asker experience - What was asking your first...

Linked

Tensorflow 2 Object Detection API: Numpy Version Errors

# Solve to run this cmd

stackoverflow.com/questions/58479556/notimplementederror-cannot-convert-a-symbolic-tensor-2nd-target0-to-a-numpy

KeepVid: Download... OpenCV with Visual... Video Lectures | Dis... The Opinion Polls N... December-2013 | In... Chicago Elsevier · Owen Bish... Inbox (491) - tabrej... Altmetric it! Altmetric it!

Log in Sign up

Home PUBLIC Questions Tags Users COLLECTIVES Explore Collectives FIND A JOB Jobs Companies TEAMS Stack Overflow for Teams – Collaborate and share knowledge with a private group. Free

2 Even I am facing the same issue, and it works perfectly when I disable eager execution. – Siddhant Nov 3, 2019 at 6:33  
@Siddhant did you find an alternative without having to disable eager execution? Disabling it seems to fix the issue, but I am no longer benefiting from the other functionalities of eager execution. – Pleastry Jun 3, 2020 at 13:35

Add a comment

5 Answers

Active Oldest Votes

For me, the issue occurred when upgrading from `numpy 1.19` to `1.20` and using `ray`'s RLLib, which uses `tensorflow 2.2` internally. Simply downgrading with `pip install numpy==1.19.5` solved the problem; the error did not occur anymore.

Run this command to solve error

Update (comment by @codeananda): You can also update to a newer TensorFlow (2.6+) version now that resolves the problem (`pip install -U tensorflow`).

Share Follow edited Jan 26 at 14:00 answered Feb 1, 2021 at 13:43 CGFoX 3,860 5 38 61

11 This was my problem too and the solution worked, thanks – Marc Laugharn Feb 2, 2021 at 8:53

2 Why is the lower one the accepted version? It just says: Don't use numpy. But when you're dependent on it, it's impossible to implement that solution. This here is the only right answer. – Lukas Fink Feb 15, 2021 at

Join Stack Overflow to learn, share knowledge, and build your career.

Sign up with email G Sign up with Google GitHub Facebook

Type here to search

23:21 07/03/2022

# Finally start the training of model

```
Anaconda Prompt (anaconda3) - python train.py --logtostderr --train_dir=training/ --pipeline_config_path=training/faster_rcnn_inception_v2_coco.config
I0307 23:19:21.801887 14992 learning.py:754] Starting Session.
INFO:tensorflow:Saving checkpoint to path training/model.ckpt
I0307 23:19:21.990761 18848 supervisor.py:1117] Saving checkpoint to path training/model.ckpt
INFO:tensorflow:Starting Queues.
I0307 23:19:21.990761 14992 learning.py:768] Starting Queues.
INFO:tensorflow:global_step/sec: 0
I0307 23:19:26.555567 17644 supervisor.py:1099] global_step/sec: 0
INFO:tensorflow:Recording summary at step 0.
I0307 23:19:27.716749 24428 supervisor.py:1050] Recording summary at step 0.
INFO:tensorflow:global step 1: loss = 4.0087 (7.256 sec/step)
I0307 23:19:29.404736 14992 learning.py:507] global step 1: loss = 4.0087 (7.256 sec/step)
INFO:tensorflow:global step 2: loss = 2.8635 (1.644 sec/step)
I0307 23:19:31.246026 14992 learning.py:507] global step 2: loss = 2.8635 (1.644 sec/step)
INFO:tensorflow:global step 3: loss = 3.7928 (1.796 sec/step)
I0307 23:19:33.057495 14992 learning.py:507] global step 3: loss = 3.7928 (1.796 sec/step)
INFO:tensorflow:global step 4: loss = 3.0844 (1.625 sec/step)
I0307 23:19:34.682565 14992 learning.py:507] global step 4: loss = 3.0844 (1.625 sec/step)
INFO:tensorflow:global step 5: loss = 2.8251 (1.606 sec/step)
I0307 23:19:36.288189 14992 learning.py:507] global step 5: loss = 2.8251 (1.606 sec/step)
INFO:tensorflow:global step 6: loss = 2.0988 (1.844 sec/step)
I0307 23:19:38.147427 14992 learning.py:507] global step 6: loss = 2.0988 (1.844 sec/step)
INFO:tensorflow:global step 7: loss = 2.7426 (1.682 sec/step)
I0307 23:19:39.828987 14992 learning.py:507] global step 7: loss = 2.7426 (1.682 sec/step)
INFO:tensorflow:global step 8: loss = 2.8295 (1.658 sec/step)
I0307 23:19:41.503159 14992 learning.py:507] global step 8: loss = 2.8295 (1.658 sec/step)
INFO:tensorflow:global step 9: loss = 2.1845 (1.701 sec/step)
I0307 23:19:43.203932 14992 learning.py:507] global step 9: loss = 2.1845 (1.701 sec/step)
INFO:tensorflow:global step 10: loss = 1.9383 (1.823 sec/step)
I0307 23:19:45.027604 14992 learning.py:507] global step 10: loss = 1.9383 (1.823 sec/step)
INFO:tensorflow:global step 11: loss = 2.1950 (1.892 sec/step)
I0307 23:19:46.921714 14992 learning.py:507] global step 11: loss = 2.1950 (1.892 sec/step)
```

# For prediction we need to convert file .ckpt file to .pb file

Copy file from object\_detection file

**export\_inference\_graph.py**

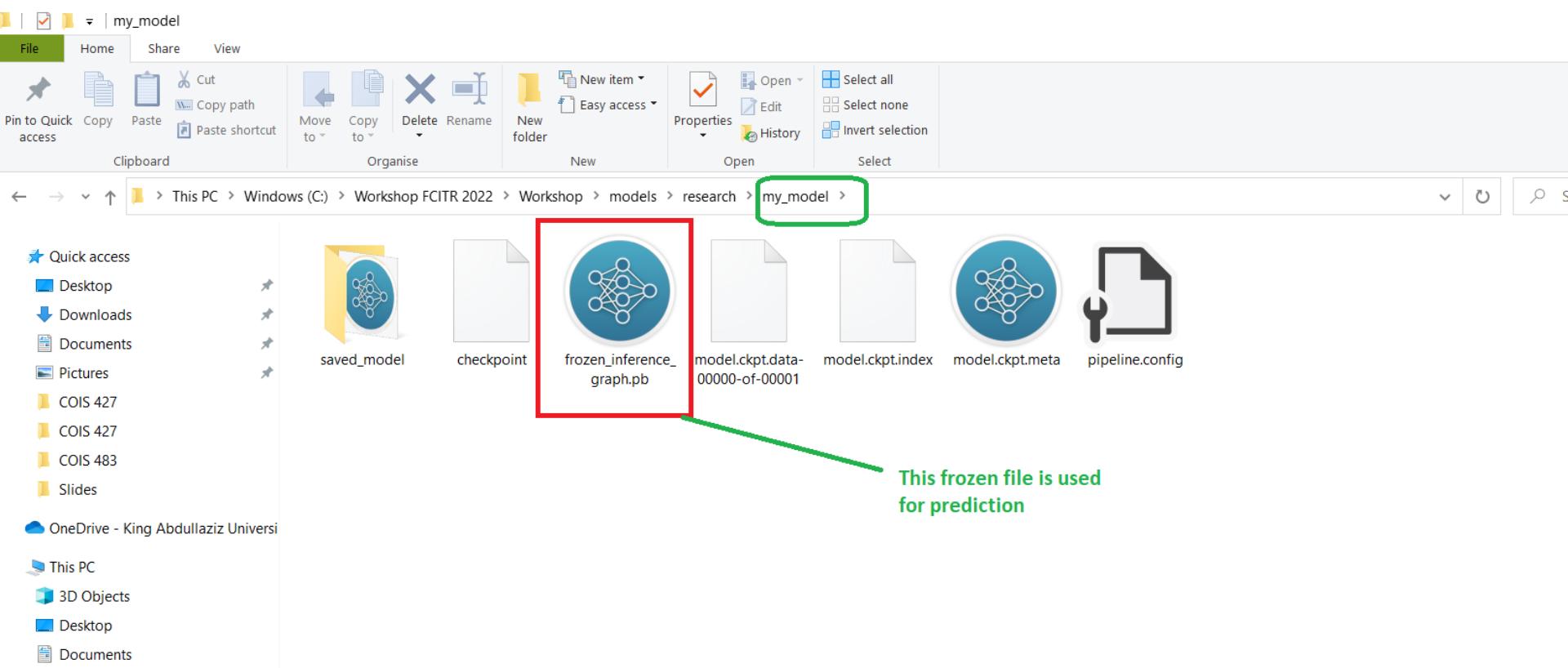
and paste into research

## Command

```
python export_inference_graph.py --input_type image_tensor --pipeline_config_path  
training/faster_rcnn_inception_v2_coco.config --trained_checkpoint_prefix  
training/model.ckpt-250 --output_directory my_model
```

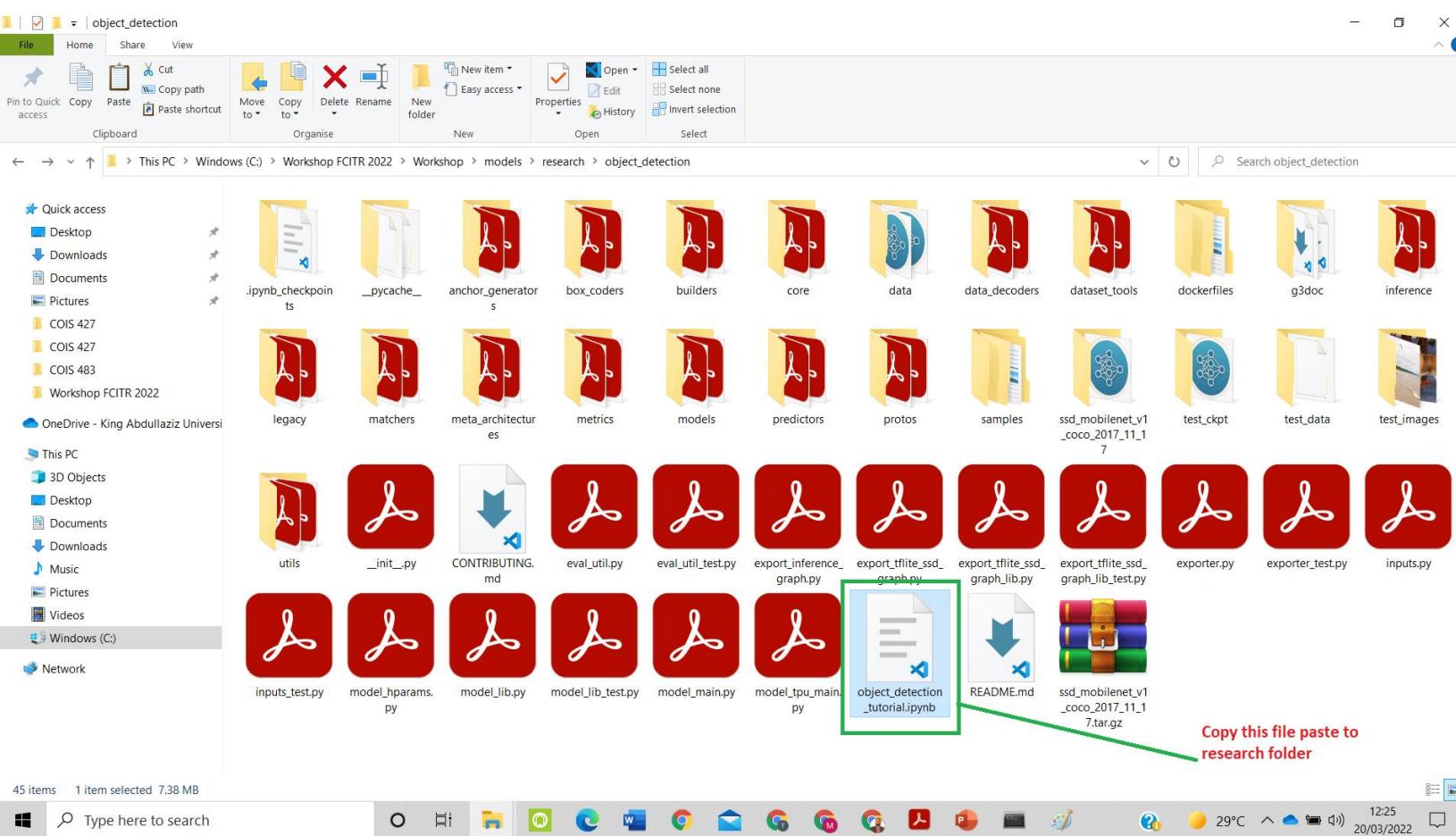
```
I0320 12:09:08.818146 12976 builder_impl.py:640] No assets to save.  
INFO:tensorflow:No assets to write.  
I0320 12:09:08.819141 12976 builder_impl.py:460] No assets to write.  
INFO:tensorflow:SavedModel written to: my_model\saved_model\saved_model.pb  
I0320 12:09:09.198475 12976 builder_impl.py:425] SavedModel written to: my_model\saved_model\saved_model.pb  
WARNING:tensorflow:From C:\Workshop FCITR 2022\Workshop\models\research\object_detection\utils\config_util.py:180: The name tf.gfile.Open is deprecated. Please use tf.io.gfile.GFile instead.  
  
W0320 12:09:09.218655 12976 module_wrapper.py:139] From C:\Workshop FCITR 2022\Workshop\models\research\object_detection\utils\config_util.py:180: The name tf.gfile.Open is deprecated. Please use tf.io.gfile.GFile instead.  
  
INFO:tensorflow:Writing pipeline config file to my_model\pipeline.config  
I0320 12:09:09.220160 12976 config_util.py:182] Writing pipeline config file to my_model\pipeline.config  
  
(tfod2) C:\Workshop FCITR 2022\Workshop\models\research>python export_inference_graph.py --input_type image_tensor --pipeline_config_path training/faster_rcnn_inception_v2_coco.config --trained_checkpoint_prefix training/model.ckpt-250 --output_directory my_model.
```

# frozen\_inference\_graph.pb will use for prediction



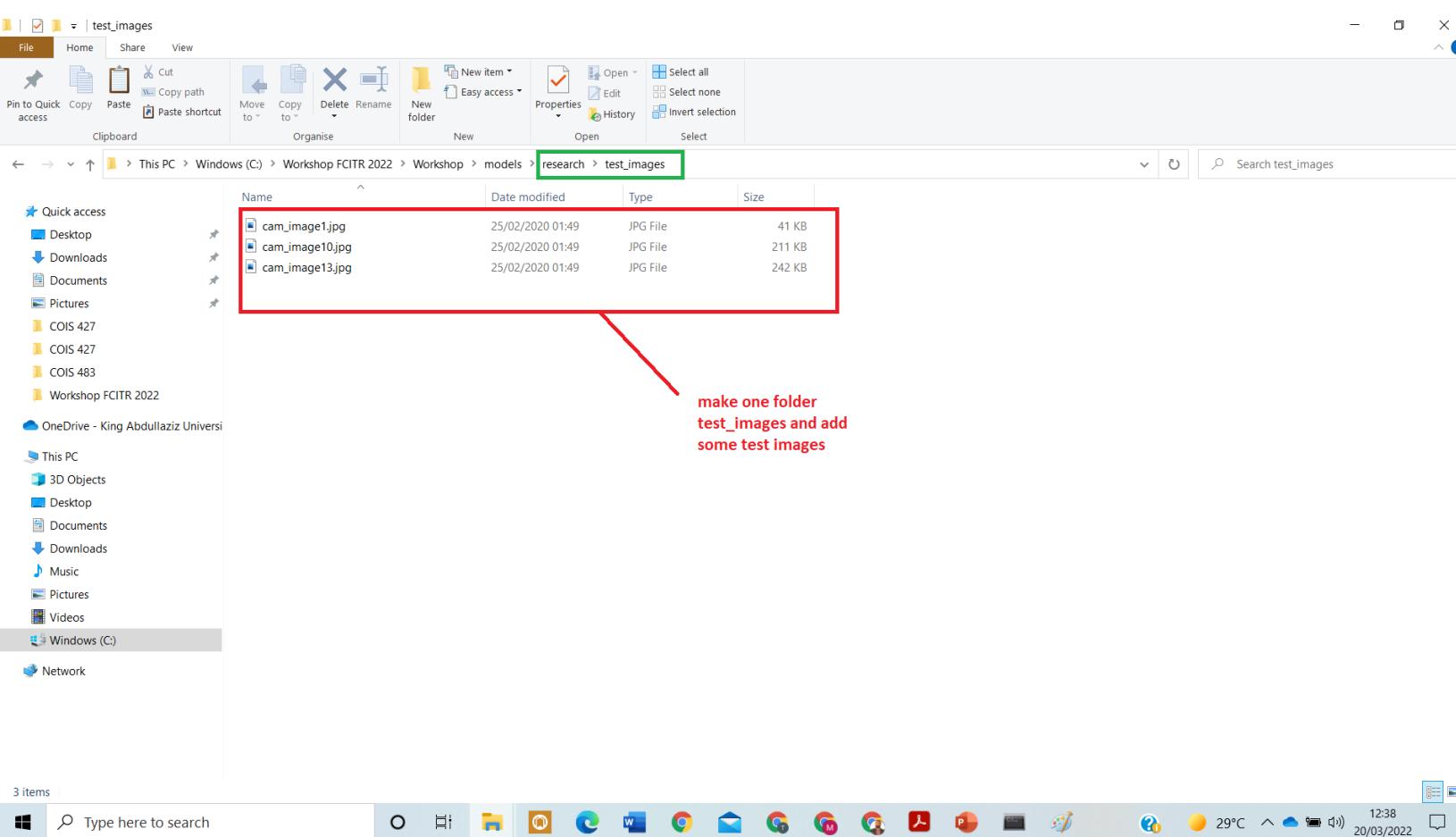
# copy file form object\_detection and past in research folder

## object\_detection\_tutorial.ipynb



after that create folder test\_images

put some images



then open jupiter notebook and open object tutorials

```
Anaconda Prompt (anaconda3)
ls.build_signature_def instead.

WARNING:tensorflow:From C:\Workshop FCITR 2022\Workshop\models\research\object_detection\exporter.py:274: The name tf.saved_model.tag_constants.SERVING is deprecated. Please use tf.saved_model.SERVING instead.

W0320 12:09:08.817150 12976 module_wrapper.py:139] From C:\Workshop FCITR 2022\Workshop\models\research\object_detection\exporter.py:274: The name tf.saved_model.tag_constants.SERVING is deprecated. Please use tf.saved_model.SERVING instead.

INFO:tensorflow:No assets to save.
I0320 12:09:08.818146 12976 builder_impl.py:640] No assets to save.
INFO:tensorflow:No assets to write.
I0320 12:09:08.819141 12976 builder_impl.py:460] No assets to write.
INFO:tensorflow:SavedModel written to: my_model\saved_model\saved_model.pb
I0320 12:09:09.198475 12976 builder_impl.py:425] SavedModel written to: my_model\saved_model\saved_model.pb
WARNING:tensorflow:From C:\Workshop FCITR 2022\Workshop\models\research\object_detection\utils\config_util.py:180: The name tf.gfile.Open is deprecated. Please use tf.io.gfile.GFile instead.

W0320 12:09:09.218655 12976 module_wrapper.py:139] From C:\Workshop FCITR 2022\Workshop\models\research\object_detection\utils\config_util.py:180: The name tf.gfile.Open is deprecated. Please use tf.io.gfile.GFile instead.

INFO:tensorflow:Writing pipeline config file to my_model\pipeline.config
I0320 12:09:09.220160 12976 config_util.py:182] Writing pipeline config file to my_model\pipeline.config

(tfod2) C:\Workshop FCITR 2022\Workshop\models\research>jupyter notebook
```



then open jupyter notebook and open object tutorials

A screenshot of a Windows desktop environment. At the top, a browser window titled "Home Page - Select or create a notebook" shows the URL "localhost:8888/tree". The main content of the browser is a Jupyter Notebook file tree under the path "jupyter/skip\_thoughts". The tree lists several sub-directories and files, many of which are marked with a "22 days ago" timestamp, indicating they were last modified recently. The files listed include "slim", "steve", "street", "struct2depth", "swivel", "syntaxnet", "tcn", "tensorrt", "test\_images" (last modified "4 minutes ago"), "textsum", "training", "transformer", "vid2depth", "video\_prediction", "object\_detection\_tutorial.ipynb" (last modified "21 days ago, 7.74 MB"), "export\_inference\_graph.py", "generate\_tfrecord.py", "README.md", "setup.py", "test.record", "train.py", and "train.record" (last modified "14 days ago, 31.4 MB"). To the right of the browser window, there are standard Windows control buttons for minimize, maximize, and close. Below the browser, the Windows taskbar is visible, featuring the Start button, a search bar with the placeholder "Type here to search", and a row of pinned application icons. The taskbar also displays system information such as the date (20/03/2022), time (12:42), and weather (29°C). The overall interface is a blend of the classic Windows aesthetic and modern web-based tools.

File/Folder	Last Modified	Size
slim	22 days ago	
steve	22 days ago	
street	22 days ago	
struct2depth	22 days ago	
swivel	22 days ago	
syntaxnet	22 days ago	
tcn	22 days ago	
tensorrt	22 days ago	
test_images	4 minutes ago	
textsum	22 days ago	
training	an hour ago	
transformer	22 days ago	
vid2depth	22 days ago	
video_prediction	22 days ago	
object_detection_tutorial.ipynb	21 days ago	7.74 MB
export_inference_graph.py	3 years ago	6.52 kB
generate_tfrecord.py	2 years ago	3.69 kB
README.md	3 years ago	5.01 kB
setup.py	3 years ago	446 B
test.record	14 days ago	6.93 MB
train.py	3 years ago	6.83 kB
train.record	14 days ago	31.4 MB

then open jupyter notebook and open object tutorials

The screenshot shows a Jupyter Notebook interface running in a browser window. The title bar indicates the notebook is titled "object\_detection\_tutorial" and is located at "localhost:8888/notebooks/object\_detectionTutorial.ipynb". The toolbar includes standard browser controls like back, forward, and search, along with Jupyter-specific icons for file operations, cell execution, and help.

The main content area displays two code cells:

**Imports**

```
In [1]: import numpy as np
import os
import six.moves.urllib as urllib
import sys
import tarfile
import tensorflow as tf
import zipfile

from distutils.version import StrictVersion
from collections import defaultdict
from io import StringIO
from matplotlib import pyplot as plt
from PIL import Image

# This is needed since the notebook is stored in the object_detection folder.
sys.path.append("..")
from object_detection.utils import ops as utils_ops

if StrictVersion(tf.__version__) < StrictVersion('1.9.0'):
    raise ImportError('Please upgrade your TensorFlow installation to v1.9.* or later!')
```

A red warning box highlights a deprecation warning from the IPython kernel launcher:

C:\Users\tabrez\anaconda3\envs\tfod2\lib\site-packages\ipykernel\_launcher.py:19: DeprecationWarning: distutils Version classes are deprecated. Use packaging.version instead.

**Env setup**

```
In [2]: # This is needed to display the images.
%matplotlib inline
```

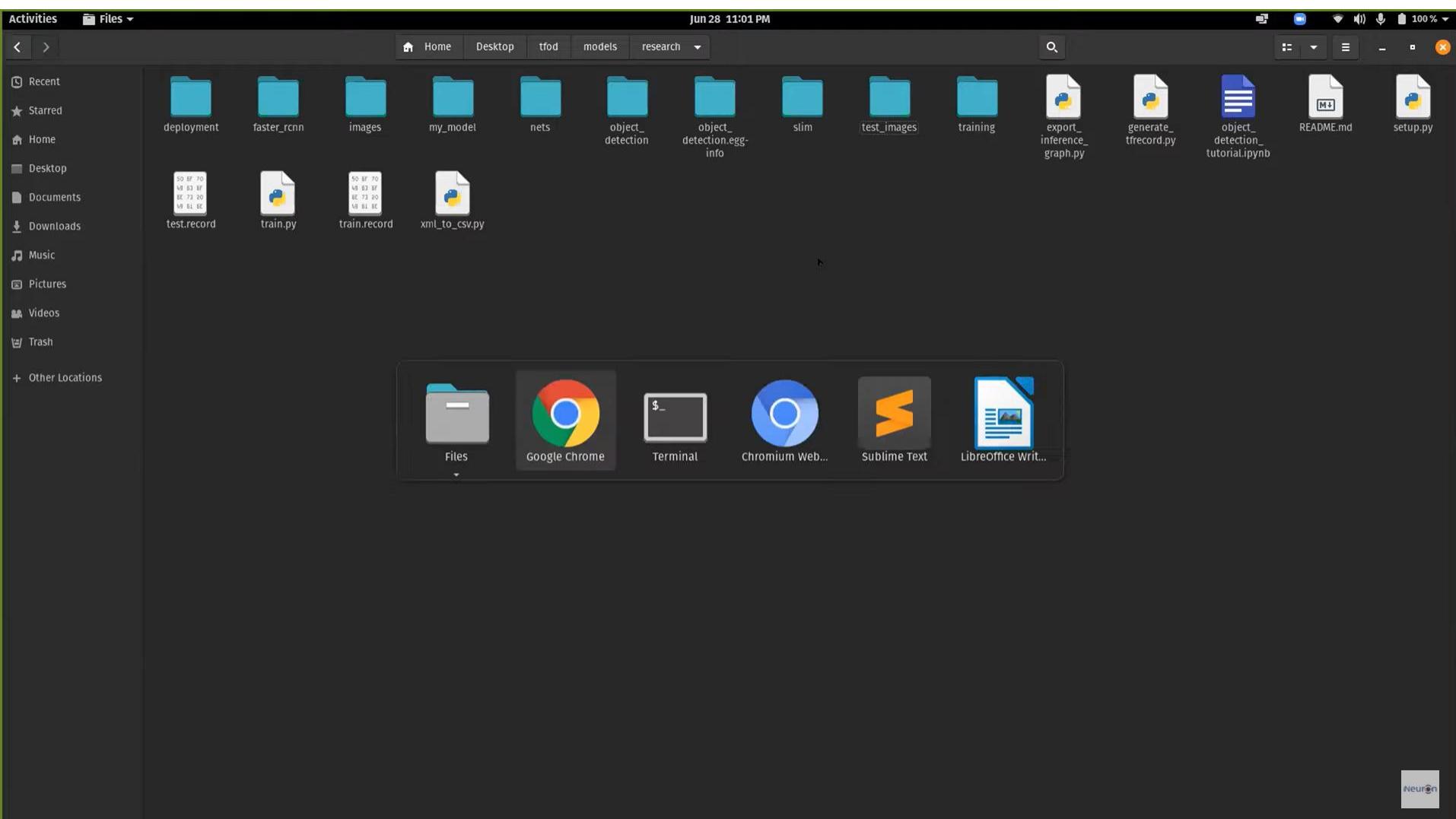
The bottom of the screen shows the Windows taskbar with various pinned application icons, and the system tray displays the date and time as 20/03/2022 12:42, along with battery and network status.

## Change these line in jupyter notebook

change

```
from object_detection.utils import label_map_util  
from object_detection.utils import visualization_utils as vis_util  
  
# What model to download.  
MODEL_NAME = 'my_model'  
#MODEL_FILE = MODEL_NAME + '.tar.gz'  
#DOWNLOAD_BASE = 'http://download.tensorflow.org/models/object_detection/'  
  
# Path to frozen detection graph. This is the actual model that is used for the object  
# detection.  
PATH_TO_FROZEN_GRAPH = MODEL_NAME + '/frozen_inference_graph.pb'  
  
# List of the strings that is used to add correct label for each box.  
PATH_TO_LABELS = os.path.join('training', 'labelmap.pbtxt')
```

# For Project many folder can delete



# Prediction

<https://c17hawke.github.io/tfod-setup/p02/>

The screenshot shows a Jupyter Notebook interface running on localhost:8888. The notebook title is "object\_detection\_tutorial". A toolbar at the top includes icons for file operations, cell types, and help. Below the toolbar, a menu bar offers File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. On the right side of the menu bar are buttons for Trusted status and Python 3 (ipykernel). The main content area displays a photograph of a playing card (the Jack of Diamonds) resting on a wooden surface. A green bounding box highlights the card, and the text "jack: 58%" is displayed above it. The card has a white border and features a portrait of a man with a crown and a sword. The notebook cell below contains the code used to display the image:

```
In [17]: %matplotlib inline  
plt.figure(figsize=(50,50))  
plt.imshow(image_np)
```

Out[17]: <matplotlib.image.AxesImage at 0x227ac703b08>

At the bottom of the screen, the Windows taskbar shows various open applications including a search bar, file explorer, and browser icons. The system tray indicates the date as 20/03/2022, the time as 16:11, and the temperature as 28°C.

code to get the pop camera

```
#code to get the pop camera
import cv2
cap = cv2.VideoCapture(0)
with detection_graph.as_default():
    with tf.Session(graph=detection_graph) as sess:
        while True:
            ret, image_np = cap.read()
            # Expand dimensions since the model expects images to have shape: [1, None, None, 3]
            image_np_expanded = np.expand_dims(image_np, axis=0)
            image_tensor = detection_graph.get_tensor_by_name('image_tensor:0')
            # Each box represents a part of the image where a particular object was detected.
            boxes = detection_graph.get_tensor_by_name('detection_boxes:0')
```

code to get the pop camera

```
# Each score represent how level of confidence for each of the objects.  
# Score is shown on the result image, together with the class label.  
  
scores = detection_graph.get_tensor_by_name('detection_scores:0')  
classes = detection_graph.get_tensor_by_name('detection_classes:0')  
num_detections = detection_graph.get_tensor_by_name('num_detections:0')  
  
# Actual detection.  
  
(boxes, scores, classes, num_detections) = sess.run([boxes, scores, classes,  
num_detections],feed_dict={image_tensor: image_np_expanded})  
  
# Visualization of the results of a detection.  
  
vis_util.visualize_boxes_and_labels_on_image_array(  
    image_np,  
    np.squeeze(boxes),  
    np.squeeze(classes).astype(np.int32),
```

code to get the pop camera

```
np.squeeze(scores),  
category_index,  
use_normalized_coordinates=True,  
line_thickness=8)  
  
cv2.imshow('object detection', cv2.resize(image_np, (800,600)))  
if cv2.waitKey(25) & 0xFF == ord('q'):  
    cv2.destroyAllWindows()  
    break  
  
cap.release()
```

# Assignment: Face Mask Detection-Project

Download Dataset

<https://www.kaggle.com/andrewmvd/face-mask-detection>

OR

Take 50 images of face mask people and take 50 images for no face mask people

Apply Any TFOD model to detect face Mask OR No Mask.

