Kammavari Sangham (R) 1952, K.S.Group of Institutions

K. S INSTITUTE OF TECHNOLOGY

No.14, Raghuvanahalli, Kanakapura Road, Bengaluru - 560109

Affiliated to VTU, Belagavi & Approved by AICTE, New Delhi, Accredited by NBA , NAAC & IEI

Department of Computer Science and Engineering

Project Phase – II (18CSP83)

Review – 1

FACE MASK DETECTION

Group No.: G3 **Batch No.: 2021_CSE_11**

1KS18CS121	YASHWANTH K
1KS18CS101	SUDHAKAR YASWANTH
1KS18CS070	PRATEEK HAVALE
1KS18CS118	VINOD H MALALI

Guided By:
Mrs. Pavithra J
Assistant Professor
Dept of CSE, KSIT

Contents

- Introduction
- Comparison with similar work
- Problem Statement and Objectives
- Methodology Proposed/ Design
- Technologies / Tools Used
- Implementation of Modules with codes
- Snapshots
- References

Introduction

- Coronavirus disease 2019 (COVID-19) has globally infected over 5.1 billion people causing over 6.2 million deaths.
- Individuals with COVID-19 have had a wide scope of symptoms reported – serious illness.
- Fever, dry cough, tiredness, loss of taste and smell are the major symptoms of coronavirus.
- Respiratory problems like difficulty in breathing.
- Elder people having lung disease can possess serious complications from COVID-19 illness as they appear to be at higher risk.

Cont...

- Many precautionary measures have been taken to fight against coronavirus. Among them cleaning hands, maintaining a safe distance and wearing a mask.
- In order to protect ourselves from the COVID-19 Pandemic, every one of us should wear a face mask.
- It becomes necessary to check if the people wear face mask in most public gatherings such as School, College, Malls etc.
- This model can be used to develop a full-fledged software to scan every person before they can enter the public gathering.

Comparison with similar work

SL NO	TITLE OF THE PAPER	AUTHOR	JOURNAL AND PUBLICATION YEAR OF PAPER	METHODOLOGY	LIMITATIONS
1	**Covid-19 Face Mask Detection Using TensorFlow, Keras and OpenCV**	1.Arjya Das 2.Mohammad Wasif Ansari 3.Rohini Bask	IEEE 2020	The proposed method consists of a cascade classifier and a pre-trained CNN which contains two 2D convolution layers	It can detect if a person is wearing the mask or not.
2	**Detection of Face Mask using Convolutional Neural Network**	1.Riya Chiragkumar Shah 2.Rutva Jignesh Shah	IEEE 2021	Used the MobileNetV2 of convolutional neural network for detection of mask.	Dataset size 2k images. 99% precision and 99% recall this may lead to overfitting problem.

Comparison with similar work

SI NO	TITLE OF THE PAPER	AUTHOR	JOURNAL AND PUBLICATION YEAR OF PAPER	METHODOLOGY	LIMITATIONS
3	**Face Mask Detection on Real-World Webcam Images**	1.Eashan Adhikarla 2.Brian D. Davison	IEEE 2021	Used Yolo V5 Model to detect mask	Yolo V5 needs high computational devices to train and predict the images.
4	**Real Time Face Mask Detection and Recognition using Python**	1.Roshan M Thomas 2.Motty Sabu 3.Tintu Samson	IJERT 2021	Used CNN to detect face mask.	Uses classification not object detection.

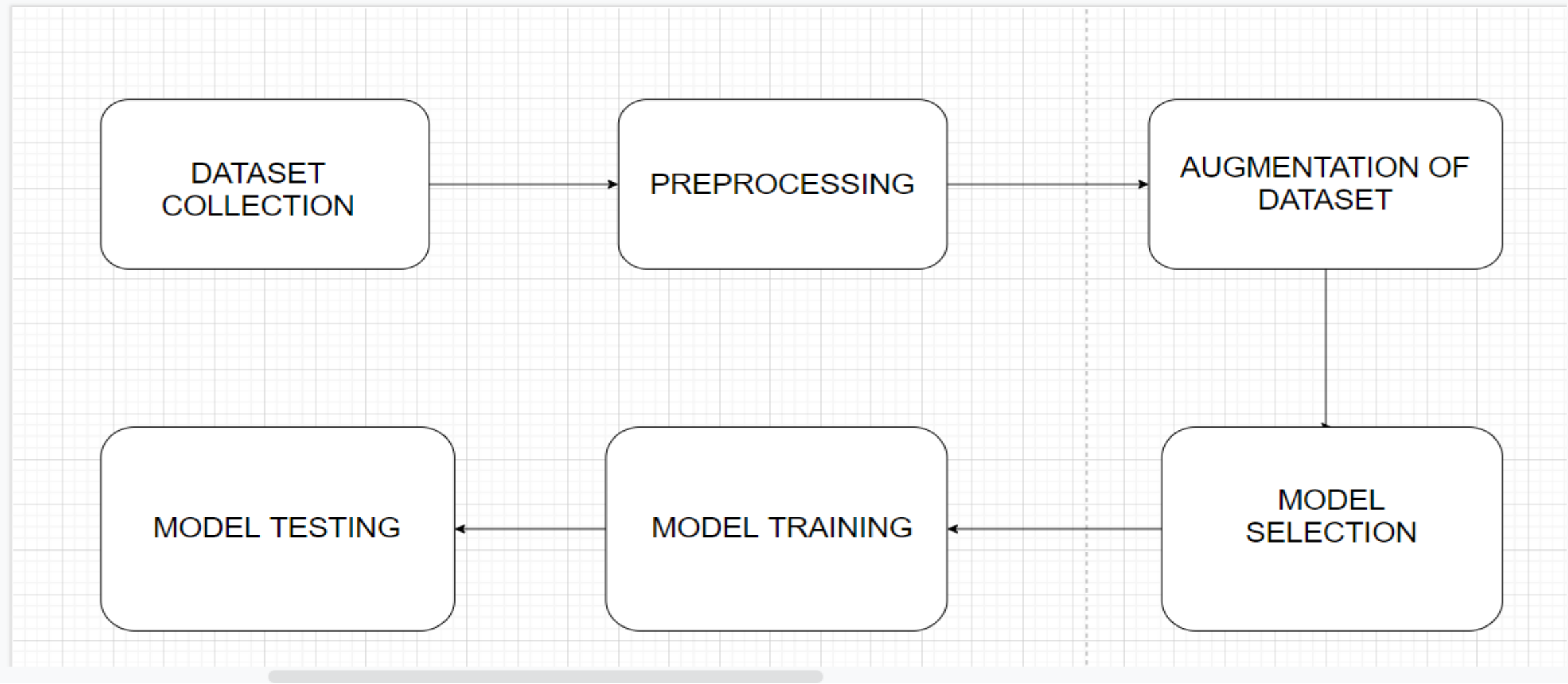
Problem Statement

"In order to protect ourselves from the COVID-19 Pandemic, almost every one of us tends to wear a face mask. It becomes increasingly necessary to check if the people in a crowd wear face masks in public gatherings such as Malls, Theaters, and Parks. The development of a solution to detect if the person is wearing a face mask and allow their entry would be of great help to society."

Objectives

- To effectively provide a working model for accurate face mask detection.
- Classification of real time face images as With and Without Mask.
- Increasing the number of images in Dataset in order to improve the Accuracy of the model.
- Optimization of the algorithm to capture and detect With and Without Mask Faces in real time.

Methodology Proposed / Design



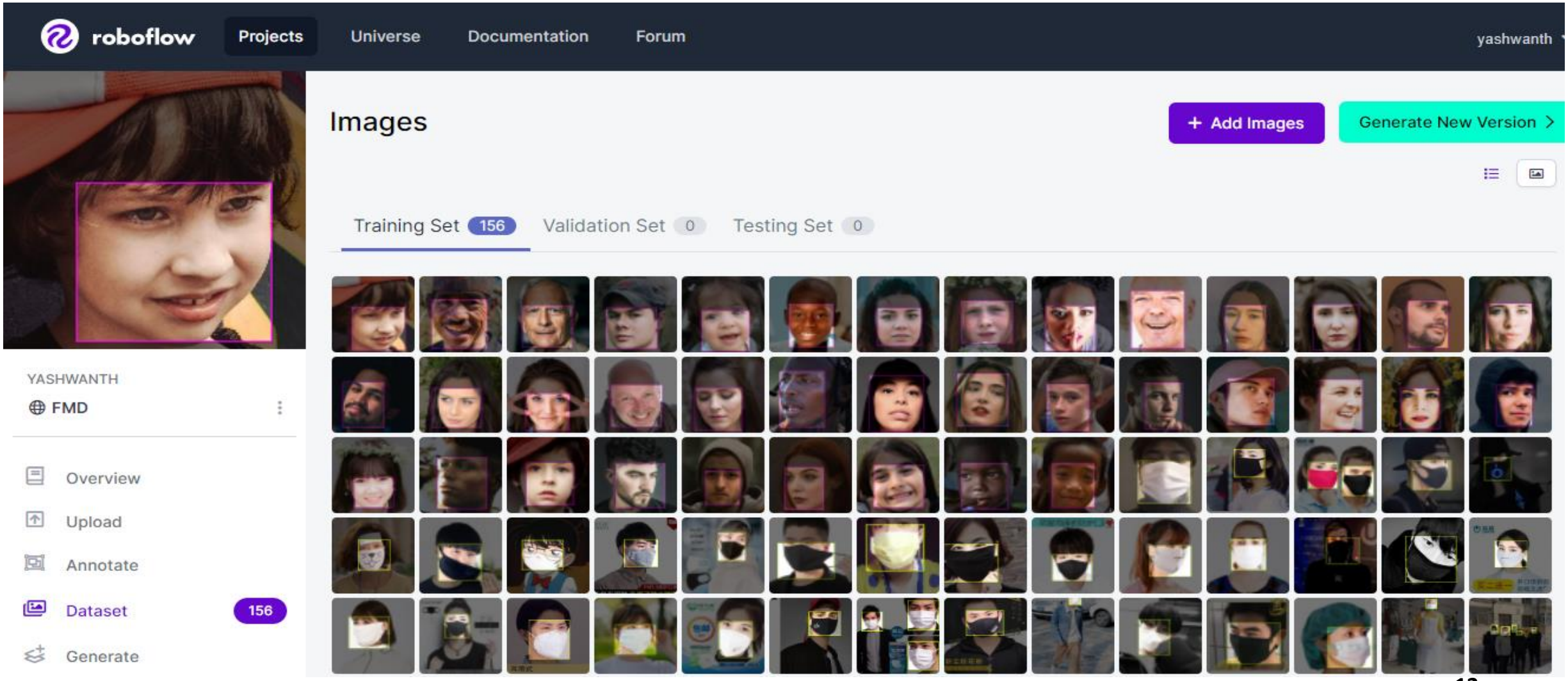
Methodology Proposed / Design

- Dataset: Collected from GitHub repository and kaggle.
- Preprocessing : Labelling, Resize done using Roboflow.
- Augmentation : Increases the size of data.
- Model Selection : YoloV4-tiny
- Model Training : Used Darknet
- Model Testing

Technologies / Tools Used

- Google Colab
- Python
- YOLO V4 Tiny
- Darknet
- Webcam
- Windows OS

Implementation of Modules with codes



Implementation of Modules with codes

```
[65] %cd ..
     from google.colab import drive
     drive.mount('/content/gdrive')

     !ln -s /content/gdrive/My\ Drive/ /mydrive

     !ls /mydrive/yolov4-tiny
```

```
/content
Drive already mounted at /content/gdrive; to attempt to forcibly remount, call drive.mount("/content/gdrive", force_remount=True).
ln: failed to create symbolic link '/mydrive/My Drive': File exists
obj.data  obj.names  obj.zip  process.py  training  yolov4-tiny-custom.cfg
```

```
[66] %cd /content/darknet/
     !sed -i 's/OPENCV=0/OPENCV=1/' Makefile
     !sed -i 's/GPU=0/GPU=1/' Makefile
     !sed -i 's/CUDNN=0/CUDNN=1/' Makefile
     !sed -i 's/CUDNN_HALF=0/CUDNN_HALF=1/' Makefile
     !sed -i 's/LIBSO=0/LIBSO=1/' Makefile
```

```
/content/darknet
```

```
[67] # building darknet
     !make
```

```
In file included from src/yolo_v2_class.cpp:2:0:
include/yolo_v2_class.hpp: In member function ‘void track_kalman_t::clear_old_states()’:
include/yolo_v2_class.hpp:879:50: warning: comparison between signed and unsigned integer expressions [-Wsign-compare]
                  if ((result_vec_pred[state_id].x > img_size.width) ||
include/yolo_v2_class.hpp:880:50: warning: comparison between signed and unsigned integer expressions [-Wsign-compare]
                      (result_vec_pred[state_id].y > img_size.height))
include/yolo_v2_class.hpp: In member function ‘track_kalman_t::tst_t track_kalman_t::get_state_id(bbox_t, std::vector<bool>&)’:
```

✓ 4s completed at 1:39 PM

Implementation of Modules with codes

```
[62] # training dataset

!./darknet detector train data/obj.data cfg/yolov4-tiny-custom.cfg yolov4-tiny.conv.29 -dont_show -map
```

```
Streaming output truncated to the last 5000 lines.
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 37 Avg (IOU: 0.759776), count: 4, class_loss = 0.342405, iou_loss = 3.686358, total_loss = 4.028763
 total_bbox = 609912, rewritten_bbox = 0.385957 %
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 30 Avg (IOU: 0.909744), count: 4, class_loss = 0.001773, iou_loss = 0.227873, total_loss = 0.229647
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 37 Avg (IOU: 0.000000), count: 1, class_loss = 0.000000, iou_loss = 0.000000, total_loss = 0.000000
 total_bbox = 609916, rewritten_bbox = 0.385955 %
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 30 Avg (IOU: 0.902353), count: 3, class_loss = 0.000295, iou_loss = 0.149265, total_loss = 0.149560
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 37 Avg (IOU: 0.834224), count: 2, class_loss = 0.005616, iou_loss = 1.323494, total_loss = 1.329110
 total_bbox = 609921, rewritten_bbox = 0.385952 %
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 30 Avg (IOU: 0.876056), count: 4, class_loss = 0.005750, iou_loss = 0.176988, total_loss = 0.182738
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 37 Avg (IOU: 0.000000), count: 1, class_loss = 0.000000, iou_loss = 0.000000, total_loss = 0.000000
 total_bbox = 609925, rewritten_bbox = 0.385949 %
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 30 Avg (IOU: 0.832914), count: 6, class_loss = 0.410965, iou_loss = 0.270438, total_loss = 0.681403
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 37 Avg (IOU: 0.000000), count: 1, class_loss = 0.000171, iou_loss = 0.000000, total_loss = 0.000171
 total_bbox = 609931, rewritten_bbox = 0.385945 %
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 30 Avg (IOU: 0.879437), count: 4, class_loss = 0.147108, iou_loss = 0.105788, total_loss = 0.252896
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 37 Avg (IOU: 0.000000), count: 1, class_loss = 0.000002, iou_loss = 0.000000, total_loss = 0.000002
 total_bbox = 609935, rewritten_bbox = 0.385943 %
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 30 Avg (IOU: 0.890362), count: 5, class_loss = 0.058161, iou_loss = 0.499339, total_loss = 0.557500
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 37 Avg (IOU: 0.917770), count: 1, class_loss = 0.089802, iou_loss = 0.495673, total_loss = 0.585475
 total_bbox = 609941, rewritten_bbox = 0.385939 %
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 30 Avg (IOU: 0.811395), count: 4, class_loss = 0.187889, iou_loss = 0.180261, total_loss = 0.368151
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 37 Avg (IOU: 0.000000), count: 1, class_loss = 0.000109, iou_loss = 0.000000, total_loss = 0.000109
 total_bbox = 609945, rewritten_bbox = 0.385936 %
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 30 Avg (IOU: 0.896808), count: 3, class_loss = 0.031683, iou_loss = 0.122770, total_loss = 0.154453
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 37 Avg (IOU: 0.608238), count: 2, class_loss = 0.276968, iou_loss = 0.631789, total_loss = 0.908757
 total_bbox = 609950, rewritten_bbox = 0.385933 %
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 30 Avg (IOU: 0.875216), count: 4, class_loss = 0.236979, iou_loss = 0.146939, total_loss = 0.383918
v3 (iou loss, Normalizer: (iou: 0.07, obj: 1.00, cls: 1.00) Region 37 Avg (IOU: 0.000000), count: 1, class_loss = 0.000287, iou_loss = 0.000000, total_loss = 0.000287
 total_bbox = 609954, rewritten_bbox = 0.385931 %
```

✓ 4s completed at 1:39 PM

Implementation of Modules with codes

```
[yolo] params: iou loss: ciou (4), iou_norm: 0.07, obj_norm: 1.00, cls_norm: 1.00, delta_norm: 1.00, scale_x_y: 1.05
[74] nms_kind: greedynms (1), beta = 0.600000
  31 route  27                                    ->   13 x  13 x 256
  32 conv    128       1 x 1/ 1     13 x  13 x 256 ->   13 x  13 x 128 0.011 BF
  33 upsample                 2x    13 x  13 x 128 ->   26 x  26 x 128
  34 route  33 23                                 ->   26 x  26 x 384
  35 conv    256       3 x 3/ 1     26 x  26 x 384 ->   26 x  26 x 256 1.196 BF
  36 conv     21       1 x 1/ 1     26 x  26 x 256 ->   26 x  26 x  21 0.007 BF
  37 yolo
[yolo] params: iou loss: ciou (4), iou_norm: 0.07, obj_norm: 1.00, cls_norm: 1.00, delta_norm: 1.00, scale_x_y: 1.05
nms_kind: greedynms (1), beta = 0.600000
Total BFLOPS 6.789
avg_outputs = 299797
 Allocate additional workspace_size = 12.46 MB
yolov4-tiny-custom
 0 : compute_capability = 370, cudnn_half = 0, GPU: Tesla K80
net.optimized_memory = 0
```

```
def imShow(path):
  import cv2
  import matplotlib.pyplot as plt
  %matplotlib inline

  image = cv2.imread(path)
  height, width = image.shape[:2]
  resized_image = cv2.resize(image,(3*width, 3*height), interpolation = cv2.INTER_CUBIC)

  fig = plt.gcf()
  fig.set_size_inches(18, 10)
  plt.axis("off")
  plt.imshow(cv2.cvtColor(resized_image, cv2.COLOR_BGR2RGB))
  #plt.show('')
```

Implementation of Modules with codes

```
[77] #set custom cfg to test mode
     %cd cfg
     !sed -i 's/batch=64/batch=1/' yolov4-tiny-custom.cfg
     !sed -i 's/subdivisions=16/subdivisions=1/' yolov4-tiny-custom.cfg
     %cd ..

     /content/darknet/cfg
     /content/darknet
```

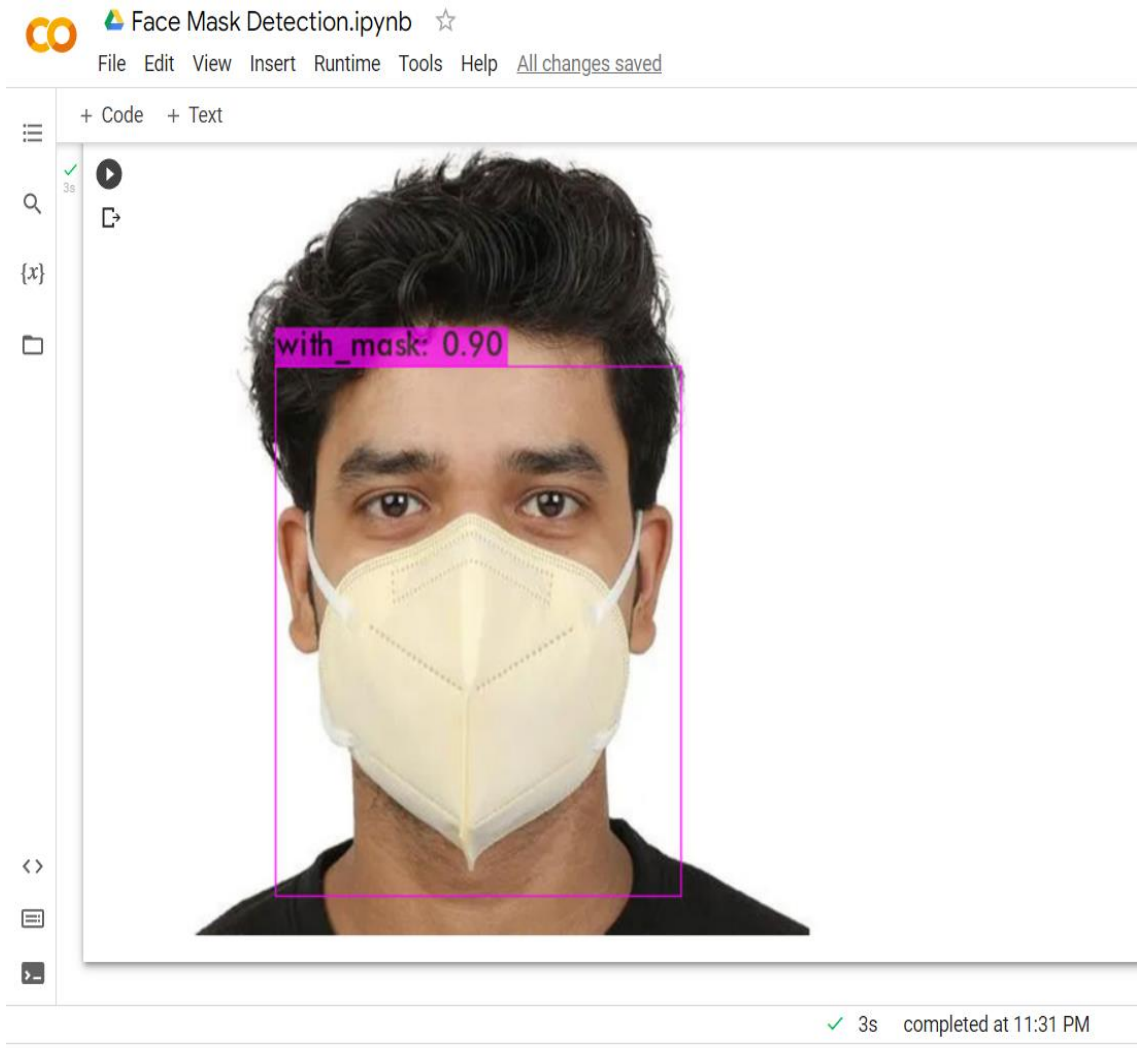
```
[78] #Testing

     !./darknet detector test data/obj.data cfg/yolov4-tiny-custom.cfg /mydrive/yolov4-tiny/training/yolov4-tiny-custom_best.weights /mydrive/test_images/image.jpg -thresh 0.3
     imShow('predictions.jpg')
```

```
 CUDA-version: 11010 (11020), cuDNN: 7.6.5, CUDNN_HALF=1, GPU count: 1
 CUDNN_HALF=1
 OpenCV version: 3.2.0
 0 : compute_capability = 370, cudnn_half = 0, GPU: Tesla K80
net.optimized_memory = 0
mini_batch = 1, batch = 1, time_steps = 1, train = 0
   layer   filters  size/strd(dil)      input                output
   0 Create CUDA-stream - 0
 Create cudnn-handle 0
conv     32       3 x 3/ 2    416 x 416 x   3 ->  208 x 208 x  32 0.075 BF
   1 conv     64       3 x 3/ 2    208 x 208 x  32 ->  104 x 104 x  64 0.399 BF
   2 conv     64       3 x 3/ 1    104 x 104 x  64 ->  104 x 104 x  64 0.797 BF
   3 route  2                                 1/2 ->  104 x 104 x  32
   4 conv     32       3 x 3/ 1    104 x 104 x  32 ->  104 x 104 x  32 0.199 BF
   5 conv     32       3 x 3/ 1    104 x 104 x  32 ->  104 x 104 x  32 0.199 BF
   6 route  5 4                                   ->  104 x 104 x  64
   7 conv     64       1 x 1/ 1    104 x 104 x  64 ->  104 x 104 x  64 0.089 BF
   8 route  2 7                                   ->  104 x 104 x 128
   9 max                2x 2/ 2    104 x 104 x 128 ->   52 x  52 x 128 0.001 BF
  10 conv    128       3 x 3/ 1     52 x  52 x 128 ->   52 x  52 x 128 0.797 BF
```

Snapshots


without_mask: 0.88
with_mask: 0.89
without_mask: 0.81
with_mask: 0.95

Snapshots


Face Mask Detection.ipynb
File Edit View Insert Runtime Tools Help All changes saved
+ Code + Text
with_mask: 0.90
3s completed at 11:31 PM


Face Mask Detection.ipynb
File Edit View Insert Runtime Tools Help All changes saved
+ Code + Text
Unable to init server: Could not connect: Connection refused
(predictions:785211): Gtk-WARNING **: 18:02:35.160: cannot open display:
without_mask: 0.87
with_mask: 0.95
shutterstock.com · 1704410278
+ Code + Text
3s completed at 11:32 PM

References

- https://www.kaggle.com/datasets
- https://roboflow.com/
- [1] Arjya Das, Mohammad Wasif Ansari, Rohini Basak “Covid-19 Face Mask Detection Using TensorFlow, Keras and OpenCV”, IEEE 17th India Council International Conference (INDICON), 2020
- [2] Riya Chiragkumar Shah, Rutva, “Detection of Face Mask using Convolutional Neural Network”, arXiv, 10th June 2021
- [3] Eashan Adhikarla Brian, D. Davison “Face Mask Detection on Real-World Webcam Images” ACM ISBN 978-1-4503-8478- 0/21/09, 2021, pp 139-144. https://dl.acm.org/doi/10.1145/34622033475903
- [4] Roshan M Thomas, Motty Sabu, Tintu Samson, Shihana Mol B, Tinu Thomas “Real Time Face Mask Detection and Recognition using Python”, IJERT, 2021, pp 57-62.


THANK
YOU