MIT Academy of Engineering
(An Autonomous Institute affiliated to Savitribai Phule Pune University)

School of Computer Engineering & Technology

A Mini Project Report

on

"Object Recognition and classification"

by

Snehal Thombare (194008)

Samruddhi Yadav (194023)

Shreyas Sanghai (194034)

Virag Shende (194062)

of

B.Tech

Academic Year: 2019-2020

Contents

Sr. No.	Topic		Page No.
Chapter-1	**Introduction**		iii
	1.1	Introduction to Project	
	1.2	Aim and Objectives	
	1.3	Problem Statement	
Chapter-2	**Survey**		iv
	2.1	Background Study of the project topic	
Chapter-3	**Proposed Model**		vi
	3.1	Algorithm used and Code	
Chapter-4	**Dataset**		ix
	4.1	Details of dataset	
	4.2	Feature vector used	
	4.3	Graph representation of Selection of features vectors	
Chapter-5	**Results & Analysis (Graphical visualization is mandatory)**		xii
Chapter-6	**Conclusion & Future work(if applicable)**		xiii
	References		xiii

1.Introduction

1.1 Introduction to Project

The recognition and classification of the diversity of materials that exist in the environment around us are a key visual competence that computer vision systems focus on in recent years. Understanding the identification of materials in distinct images involves a deep process that has made usage of the recent progress in neural networks which has brought the potential to train architectures to extract features for this challenging task. People are able to recognize the environment they are in as well as the various objects in their everyday life no matter the influence on the item's features or if their view is obstructed, as this is one of the very first skills we learn from the moment we are born.

Computers, on the other hand, require effort and powerful computation and complex algorithms to attempt to recognize correctly patterns and regions where a possible object might be. Object detection and recognition are two main ways that have been implemented over multiple decades that are at the center of Computer Vision systems at the moment. These approaches are presented with challenges such as scale, occlusion, viewpoint, illumination or background clutter, all issues that have been attempted as research topics that provided functionality that led to the introduction of Neural Networks and Convolutional Neural Networks (CNN).

1.2 Aim and Objective

The main aim and objectives are as below

- To get object like plane, car, etc. recognised.
- To classify the image into a proper category.
- Built a model which can perform both the function in a mannered way.

1.3 Problem Statement

The problem statement is to build a model which can recognize the given object and classifying it with proper category.

2. Survey

1. Overview of deep-learning based methods for salient object detection in videos

Video salient object detection is a challenging and important problem in computer vision domain. In recent years, deep-learning based methods have contributed to significant improvements in this domain. This paper provides an overview of recent developments in this domain and compares the corresponding methods up to date, including

1) Classification of the state-of-the-art methods and their frameworks

2) Summary of the benchmark datasets and commonly used evaluation metrics

3) Experimental comparison of the performances of the state-of-the-art methods

4) Suggestions of some promising future works for unsolved challenges

2. Key protected classification for collaborative learning

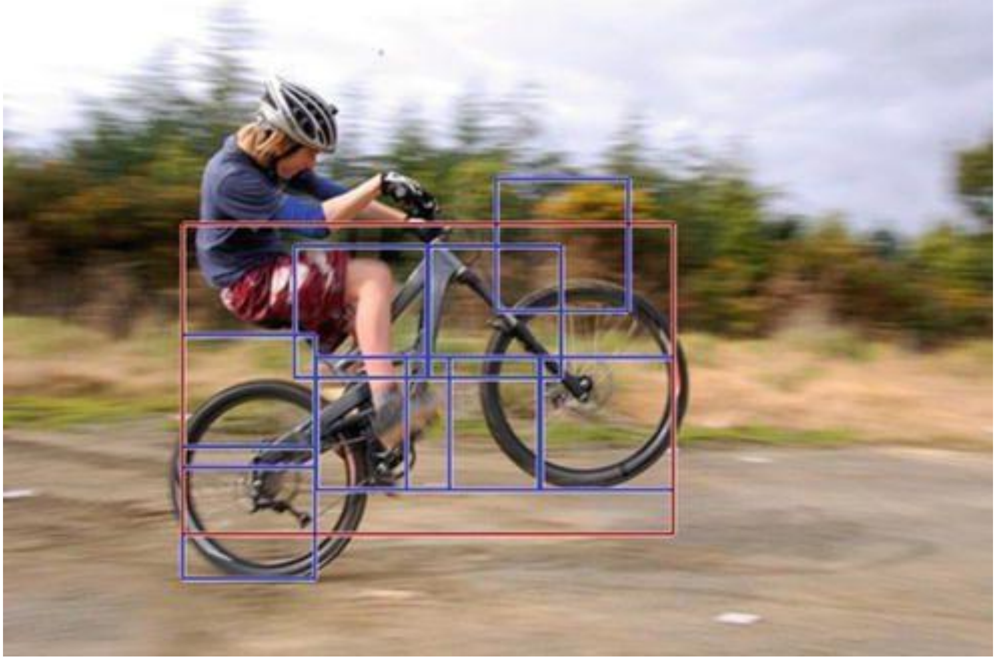
Large-scale datasets play a fundamental role in training deep learning models. However, dataset collection is difficult in domains that involve sensitive information. Collaborative learning techniques provide a privacy-preserving solution, by enabling training over a number of private datasets that are not shared by their owners. However, recently, it has been shown that the existing collaborative learning frameworks are vulnerable to an active adversary that runs a generative adversarial network (GAN) attack. In this work, we propose a novel classification model that is resilient against such attacks by design. More specifically, we introduce a key-based classification model and a principled training scheme that protects class scores by using class-specific private keys, which effectively hide the information necessary for a GAN attack. We additionally show how to utilize high dimensional keys to improve the robustness against attacks without increasing the model complexity. Our detailed experiments demonstrate the effectiveness of the proposed technique.

3. End-to-end training of CNN ensembles for person re-identification

We propose an end-to-end ensemble method for person re-identification (ReID) to address the problem of over fitting in discriminative models. These models are known to converge easily, but they are biased to the training data in general and may produce a high model variance, which is known as over fitting. The ReID task is more prone to this problem due to the large discrepancy between training and test distributions. To address this problem, our proposed ensemble learning framework produces several diverse and accurate base learners in a single DenseNet. Since most of the costly dense blocks are shared, our method is computationally efficient, which makes it favourable compared to the conventional ensemble models. Experiments on several benchmark datasets demonstrate that our method achieves state-of-the-art results. Noticeable performance improvements, especially on relatively small datasets, indicate that the proposed method deals with the over fitting problem effectively.

2.1 Background Study of the project

The aim of object detection is to detect all instances of objects from a known class, such as people, cars or faces in an image. Generally, only a small number of instances of the object are present in the image, but there is a very large number of possible locations and scales at which they can occur and that needs to somehow be explored. Each detection of the image is reported with some form of pose information. This is as simple as the location of the object, a location, and scale, or the extent of the object defined in terms of a bounding box. In some other situations, the pose information is more detailed and contains the parameters of a linear or non-linear transformation. For example for face-detection in a face detector may compute the locations of the eyes, nose, and mouth, in addition to the bounding box of the face. An example of a bicycle detection in an image that specifies the locations of certain parts is shown in below fig. The pose can also be defined by a three-dimensional transformation specifying the location of the object relative to the camera. Object detection systems always construct a model for an object class from a set of training examples. In the case of a fixed rigid object in an image, only one example may be needed, but more generally multiple training examples are necessary to capture certain aspects of class variability.



3. Proposed Model

3.1 Algorithm used and code

```
# Load Packages

library(EBImage)

library(keras)


# To install EBimage package, you can run following 2 lines;

# install.packages("BiocManager")

# BiocManager::install("EBImage")


# Read images

setwd('---')

pics <- c('p1.jpg', 'p2.jpg', 'p3.jpg', 'p4.jpg', 'p5.jpg', 'p6.jpg',

          'c1.jpg', 'c2.jpg', 'c3.jpg', 'c4.jpg', 'c5.jpg', 'c6.jpg')

mypic <- list()

for (i in 1:12) {mypic[[i]] <- readImage(pics[i])}


# Explore

print(mypic[[1]])

display(mypic[[8]])

summary(mypic[[1]])

hist(mypic[[2]])

str(mypic)


# Resize

for (i in 1:12) {mypic[[i]] <- resize(mypic[[i]], ---, ---)}
```

```
# Reshape
for (i in 1:12) {mypic[[i]] <- array_reshape(mypic[[i]], c(---, ---,---))}

# Row Bind
trainx <- NULL
for (i in 7:11) {trainx <- rbind(trainx, mypic[[i]])}
str(trainx)
testx <- ---(mypic[[6]], mypic[[12]])
trainy <- c(0,0,0,0,0,1,1,1,1,1 )
testy <- c(---, ---)

# One Hot Encoding
trainLabels <- ---(trainy)
testLabels <- ---(testy)

# Model
model <- keras_model_sequential()
model %>%
      layer_dense(units = 256, activation = ---, input_shape = c(2352)) %>%
      layer_dense(units = 128, activation = 'relu') %>%
      layer_dense(units = 2, activation = ---)
summary(model)

# Compile
model %>%
```

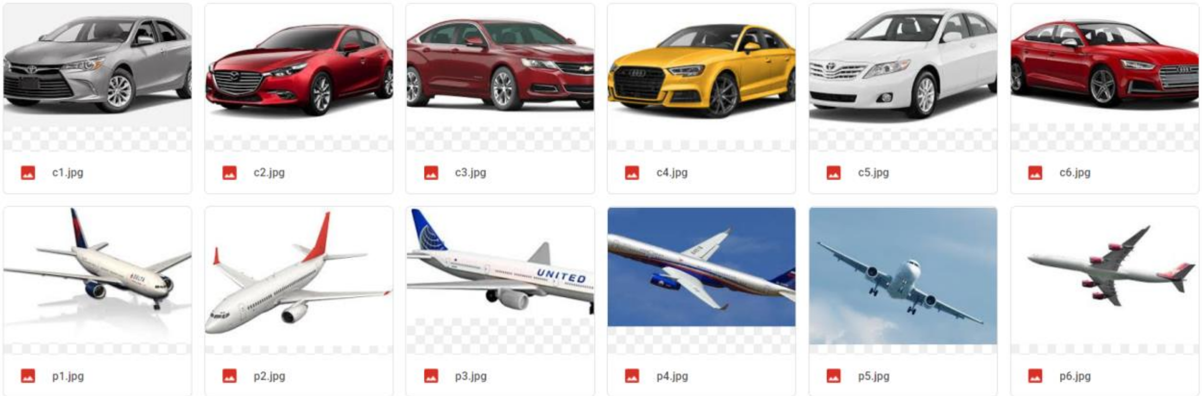

```
      compile(loss = ---,
            optimizer = optimizer_rmsprop(),
```

```
                metrics = c('accuracy'))

# Fit Model
history <- model %>%
         fit(trainx,
             ---,
             epochs = 30,
             batch_size = 32,
             validation_split = 0.2)

# Evaluation & Prediction - train data
model %>% evaluate(---, ---)
pred <- model %>% predict_classes(trainx)
table(Predicted = pred, Actual = trainy)
prob <- model %>% predict_proba(trainx)
cbind(prob, Prected = pred, Actual= trainy)
```

4. Dataset

4.1 Detail of Dataset

The dataset consist of photo of different objects (cars and planes). All the images are in .jpeg format with no particular size. Images in the dataset can be of any size. For instance, we have considered only five images of car as well as plane. In order to get more accuracy, we can use large dataset to train the model.

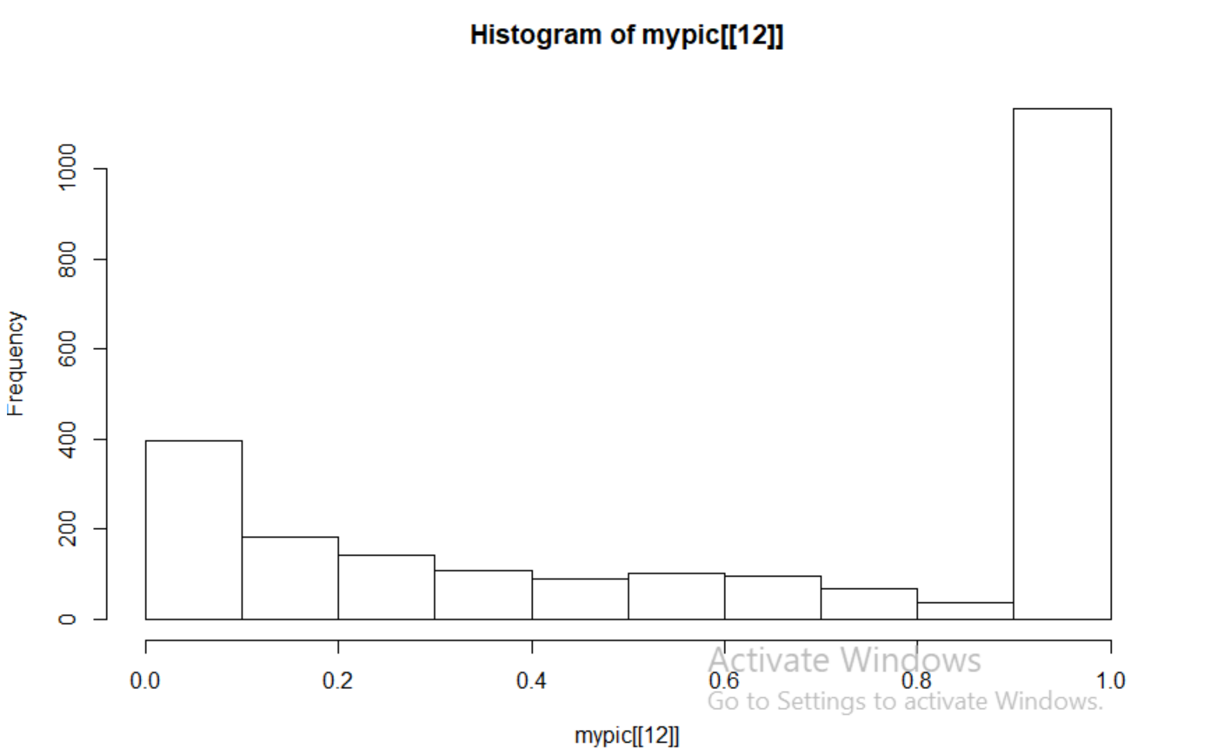


4.2 Feature Vector Used

There following are the feature vectors on which the images are getting recognised and classified.

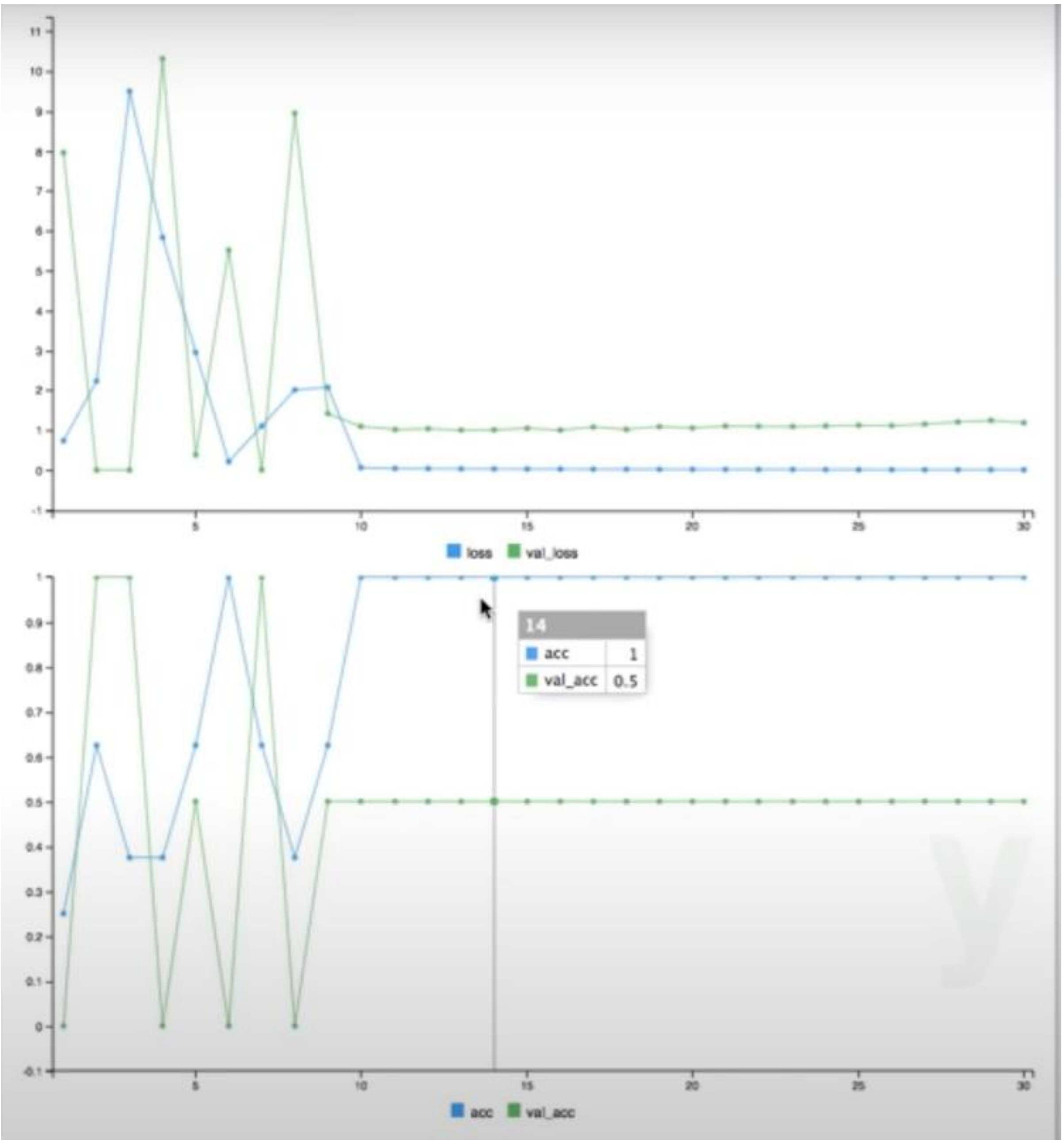
- Wings of plane.
- Wheel of car.
- Surface area of the object.

4.3 Graph Representation



In the above graph, you can see the frequency vs the dataset provided graph.

```
$loss
[1] 0.009192532

$acc
[1] 1
```



In the above fig, the first graph represents an a loss function of the model and the next graph shows the accuracy of the model.

5. Result and Analysis

The model predicts the objects with 90 percentage of accuracy. Only one of the given images has the wrong assumption.

[2,]	0.997967660	0.002032341	0	0
[3,]	0.997110963	0.002889029	0	0
[4,]	0.994729102	0.005270926	0	0
[5,]	0.993413627	0.006586361	0	0
[6,]	0.014752991	0.985247016	1	1
[7,]	0.001549294	0.998450637	1	1
[8,]	0.005016454	0.994983554	1	1
[9,]	0.011367152	0.988632858	1	1
[10,]	0.905808449	0.094191574	0	1

6. Conclusion

With the help of pattern recognition techniques, we can build a model that can extract the features and classify them according to different categories.

References

[1] T. O. Binford, "Visual perception by computer," in IEEE Conference on Systems and Control, 1971.

[2] C. Liu, L. Sharan, E. H. Adelson and R. Rosenholtz, "Exploring features in a Bayesian framework for material recognition," in The Twenty-Third {IEEE} Conference on Computer Visoin and Pattern Recognition {CVPR}, San Francisco, CA, USA, 13-18 June 2010.

[3] A. Krizhevsky, I. Sutskeevr and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Advances in Neural Information Processing Systems, pp. 1106-1114, 2012.

[4] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in

Computer Vision - {ECCV} 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014.

[5] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng and T. Darrell, "Decaf: A Deep Convolutional Activation Feature for Generic Visual Recognition," CoRR, vol. abs/1310.1531, 2013.

[6] A. S. Razavian, H. Azizpour, J. Sulliva and S. Carlsson, "CNN Features off-the-shelf: an Astounding Baseline for Recognition," CoRR, vol. abs/1403.6382, 2014.

[7] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in Conference on Computer Vision and Pattern Recognition {CVPR}, Columbus, OH, USA, pp. 580-587, 2014.

[8] K. J. Dana, B. V. Ginneken, S. K. Nayar and J. J. Koenderink, "Reflectance and Texture of RealWorld Surfaces," ACM Trans. Graph, vol. 18, pp. 1-34, 1999.

[9] M. Varma and A. Zisserman, "A Statistical Approach to Materialn Classification Using Image Patch Exemplars," IEEE Trans. Pattern Anal. Mach. Intel, vol. 31, pp. 2032-2047, 2009.