SAVITRIBAI PHULE PUNE UNIVERSITY

A PROJECT REPORT ON

NATURAL LANGUAGE DESCRIPTION OF VIDEOS

SUBMITTED TOWARDS THE
PARTIAL FULFILLMENT OF THE REQUIREMENTS OF

BACHELOR OF ENGINEERING

(Computer Engineering)

BY

Omkar Acharya	B120054204
Parag Ahivale	B120054211
Nehal Belgamwar	B120054237
Gurnur Wadhwani	B120054485

Under The Guidance of

Prof. S.S.Sonawane

DEPARTMENT OF COMPUTER ENGINEERING
Pune Institute of Computer Technology
Dhankawadi, Pune-411043

PUNE INSTITUTE OF COMPUTER TECHNOLOGY DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE

This is to certify that the Project Entitled

NATURAL LANGUAGE DESCRIPTION OF VIDEOS

Submitted by

Omkar Acharya	B120054204
Parag Ahivale	B120054211
Nehal Belgamwar	B120054237
Gurnur Wadhwani	B120054485

is a bonafide work carried out by these students under the supervision of Prof. S.S.Sonawane and it is submitted towards the partial fulfillment of the requirement of Bachelor of Engineering (Computer Engineering).

Prof. S.S.Sonawane
Internal Guide
Dept. of Computer Engg.

Prof. G. P. Potdar
H.O.D
Dept. of Computer Engg.

Dr. P. T. Kulkarni
Principal
Pune Institute of Computer Technology

Signature of Internal Examiner

Signature of External Examiner

PROJECT APPROVAL SHEET

A Project Titled

Natural Language Description of Videos

Is successfully completed by

Omkar Acharya	B120054204
Parag Ahivale	B120054211
Nehal Belgamwar	B120054237
Gurnur Wadhwani	B120054485

at

DEPARTMENT OF COMPUTER ENGINEERING

PUNE INSTITUTE OF COMPUTER TECHNOLOGY

SAVITRIBAI PHULE PUNE UNIVERSITY,PUNE

ACADEMIC YEAR 2015-2016

Prof. S.S.Sonawane	Prof. G. P. Potdar
Internal Guide	H.O.D
Dept. of Computer Engg.	Dept. of Computer Engg.

Abstract

Giving the user the choice to watch the video or read its description even without playing it, or both, we present a model that generates natural language description of videos in English language. For this purpose, we are using deep learning algorithms with both convolutional and recurrent structure. Our system extracts features from the video frames using a pre-trained Convolutional Neural Network viz. VGGNET (16 layer) that is trained on ImageNet dataset. Caffes python module is used for the same. We feed these features to a LSTM (Long Short Term Memory) to generate the description for the input video. We trained the LSTM model on MSCOCO dataset and used Chainer framework for the related processing. Our system can find its application in video search engines that could provide better search results using video description instead of the existing systems that mainly rely on video titles. Also such a system can help the blind to comprehend video content.

Motivation: The motivation behind selecting this topic as a project was the absence of a system which generates semantically correct description of videos without playing them. The absence of semantically correct description introduces ambiguity, hence is undesirable. This has motivated us to take up the project so that we could contribute to better the previous versions of similar systems.

Acknowledgments

It gives us great pleasure in presenting the preliminary project report on ***'Natural Language Description of Videos'****.*

We would like to take this opportunity to thank our internal guide ***Prof. S.S.Sonawane*** *for giving us all the help and guidance we needed. We are really grateful to her for her kind support. Her valuable suggestions were very helpful.*

We are also grateful to other staff members of the Computer Department for their indispensable support, suggestions.

Our external guide ***Dr. Parag Kulkarni*** *was always there to extend his helping hand and share his valuable technical knowledge about the subject with us.*

We are thankful to PICT library and staff, for providing us the research papers and reference books that helped us to study the project topic in depth.

Omkar Acharya
Parag Ahivale
Nehal Belgamwar
Gurnur Wadhwani
(B.E. Computer Engg.)

Contents

List of Figures

List of Tables

CHAPTER 1

SYNOPSIS

1.1 Project Title

Natural Language Description of Videos

1.2 Project Option

The project is being sponsored and mentored by Iknowlation Research Labs, Pune.

1.3 Internal Guide

Prof. S.S.Sonawane

1.4 Sponsorship and External Guide

Sponsorship: Iknowlation Research Labs
External Guide: Dr. Parag Kulkarni

1.5 Technical Keywords (As per ACM Keywords)

1. I. Computing Methodologies
    (a) ARTIFICIAL INTELLIGENCE
        i. Natural Language Processing
            A. Concept Learning
            B. Connectionism and Neural Nets
            C. Knowledge Acquisition
            D. Language Acquisition
            E. Parameter Learning
        ii. Object Detection and Recognition
            A. Models : Neural Nets
            B. Applications : Computer vision

1.6 Problem Statement

To generate natural language description of videos in English language using Deep Learning.

1.7 Abstract

Giving the user the choice to watch the video or read its description even without playing it, or both, we present a model that generates natural language description of videos in English language. For this purpose, we are using deep learning algorithms with both convolutional and recurrent structure. Our system extracts features from the video frames using a pre-trained Convolutional Neural Network viz. VGGNET (16 layer) that is trained on ImageNet dataset. Caffes python module is used for the same. We feed these features to a LSTM (Long Short Term Memory) to generate the description for the input video. We trained the LSTM model on MSCOCO dataset and used Chainer framework for the related processing. Our system can find its application in video search engines that could provide better search results using video description instead of the existing systems that mainly rely on video titles. Also such a system can help the blind to comprehend video content.

1.8 Goals and Objectives

- The goal of this project is to generate natural language description of videos without playing them.
- Objectives:
  1. To divide the input video into frames.
  2. To extract features from frames.
  3. To generate a meaningful sentence using the previously extracted features.

1.9 Relevant mathematics associated with the Project

System Description:

```
Let S be the system solution for the given problem statement such that
S={s,e,X,Y,DD,NDD,Fme,Sc,Fc}
where
s-> start state
s={GPU , dataset}

e-> end state
e={Description in English language.}

X-> input set
X={X1,X2}
where
X1-> input dataset
X1={d1,d2,d3,...,dn| di is training image with captions, n=1.6m}
X2-> user input i.e., video file
X2={.mp4, .mkv, etc file format}

Y-> output set
Y={Y1,Y2}
where
Y1-> semantically correct description of input video.
Y2-> feature vector

Fme-> set of main function
Fme={fframe,fcnn,flstm}

fframe-> function to get distinct frames from the input video file.

input-> X2
output -> X3
X3-> image frames of videos
where
X3={m1,m2,m3,...,mn| mi    di ^ X1 }


fcnn-> function to train the CNN using the dataset
```

```
input-> X1
output -> Y2

flstm-> function to generate a meaningful sentence from the feature v
input-> Y2
output -> Y1

DD-> deterministic data
DD = {X1,X2}
NDD-> non deterministic data
NDD= {Y1,Y2,X3}

Sc-> success case
Sc= Semantically correct and relevant textual description of the input

Fc-> failure case
Fc= Vague video description and input video outside the specified scop
```

1.10 Names of Conferences / Journals where papers can be published

- IEEE ICMLA (International Conference on Machine Learning and Applications)
- IEEE ICMLC (International Conference on Machine Learning and Cybernetics)
- IEEE ICDM (International Conference on Data Mining)
- ICML (International Conference on Machine Leaning)

1.11 Review of Conference/Journal Papers supporting Project idea

- Oriol Vinyals, Alexander Toshev, Samy Bengio, Dumitru Erhan, "Show and Tell: A Neural Image Caption Generator", arXiv: 1411.4555v2 [cs.CV] 20 Apr 2015.

  This paper includes recognizing objects in images with a fixed set of categories. They work on the traditional methodology i.e., subject verb object mechanism.

- Yashaswi Verma, Ankush Gupta, Prashanth Mannem, C. V. Jawahar, “Generating Image Descriptions Using Semantic Similarities in the Output Space”, in 2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops.

  They propose an advancement to above mentioned paper by finding inter-phrase semantic similarity in sentences, which can be done using generalized latent semantic analysis. But it is less accurate.

- Girish Kulkarni, Visruth Premraj, Vicente Ordonez, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C. Berg, Tamara L. Berg, “BabyTalk: Understanding and Generating Simple Image Descriptions”, in IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 35, No. 12, December 2013, pp. 2891-2903.

  This paper makes the use of Conditional Random Field (CRF) to predict the label for an input image. The mechanism used is decoding using language models for surface realization. Such systems have their scope limited to specific images.

- Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, Trevor Darrell, “Long-term Recurrent Convolutional Networks for Visual Recognition and Description”, arXiv:1411.4389v3 [cs.CV] 17 Feb 2015.

  To leverage the strengths of neural networks, Long Term Recurrent Neural Networks (LRCN) model is used in this paper to predict image features. It applies to time varying inputs and outputs. But advancement leads to increase in model size and hence high running time.

1.12 Plan of Project Execution

1.) Begun by performing a survey on Deep Learning and its applications.
2.) Decision of object to be worked on, from Images, Text, Videos etc.

3.) Survey of applications of Deep Learning in object detection and recognition and also natural language processing.
4.) Begin implementation of the first increment.
5.) Training using dataset to create a model.
6.) Testing with the test data, measurement of accuracy.
7.) Classify a single image based on the learned features.
8.) Classification of frames generated from the video file.
9.) Generation of a meaningful sentence in English language of the objects in the frames of the video.

CHAPTER 2

TECHNICAL KEYWORDS

2.1 Area of Project

Deep Learning

2.2 Technical Keywords

1. I. Computing Methodologies
    - (a) ARTIFICIAL INTELLIGENCE
        - i. Natural Language Processing
            - A. Concept Learning
            - B. Connectionism and Neural Nets
            - C. Knowledge Acquisition
            - D. Language Acquisition
            - E. Parameter Learning
        - ii. Object Detection and Recognition
            - A. Models : Neural Nets
            - B. Applications : Computer vision

CHAPTER 3

INTRODUCTION

3.1 Project Idea

The idea of the project is to create a system which will generate the semantically correct natural language description of input video within a defined scope.

3.2 Motivation of the Project

The motivation behind selecting this topic as a project was the absence of a system which generates semantically correct description of videos without playing them. The absence of semantically correct description introduces ambiguity, hence is undesirable. This has motivated us to take up the project so that we could contribute to better the previous versions of similar systems.

3.3 Literature Survey

- Translating Videos to Natural Language Using Deep Recurrent Neural Networks
  Review: Convolutional Neural Networks (CNN): To convert videos to a xed length representation (input x1), we use Convolutional Neural Network(CNN) and Recurrent Neural Network(RNN): learning to map sequences of inputs to a xed length vector using one RNN, and then map the vector to an output sequence using another RNN. We identify the most likely description for a given video by training a model to maximize the log likelihood of the sentence S.

- BabyTalk: Understanding and Generating Simple Image Descriptions (IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 35, NO. 12, DECEMBER 2013)
  Review: Conditional Random Fields: It is used to predict labeling for an input image. Decoding using Language Models: Used for surface realization.

- Generating Image Descriptions Using Semantic Similarities in the Output Space (2013 IEEE Conference on Computer Vision and Pattern Recognition Workshops)
  Review: Phrase Prediction Model: Given images and corresponding descriptions, a set of phrases is extracted using all the descriptions. Each image is represented using set of features and distance between them is computed using weighted sum of each distances to each feature.

Using this,k most similar images are found and then the probability of a phrase for a given image is calculated.

- Automatic Caption Generation for News Images (IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 35, No.4, April 2013)
  Review: Content Selection: What is the article and image about? Surface Realization: Decides how to verbalize the chosen content.

CHAPTER 4

PROBLEM DEFINITION AND SCOPE

4.1 Problem Statement

To generate natural language description of videos in English language using Deep Learning.

4.1.1 Goals and objectives

- The goal of this project is to generate natural language description of videos without playing them.
- Objectives:
    1. To divide the input video into frames.
    2. To extract features from frames.
    3. To generate a meaningful sentence using the previously extracted features.

4.1.2 Statement of scope

- The system has been trained only for the scope of an educational institute or college.
- The system would run efficiently for 15 minutes long videos.

4.2 Major Constraints

- Short length videos.
- Videos in the scope.

4.3 Methodologies of Problem solving and efficiency issues

- Segmentation
  Different frames/images are extracted from the input video considering the dissimilarity from the previous frame, as similar frames may cause redundancy.

- Convolution
  This method transforms an image volume into an output volume, thus helping to classify the images based on the class scores. A ConvNet architecture which is a list of layers, helps to do this.
- Generation
  A semantically correct sentence is generated based on the objects classified in the images from the grammar and the captions of the images in the training data.

4.4 Outcome

- A system that can be used by video hosting websites to generate automatic description of hosted videos.

4.5 Applications

- Web hosting websites
  To help in generating video description automatically for hosted videos.
- Video search engines
  To give better search results for descriptive search queries.
- Segregation of videos based on content
  To help in segregating videos from massive dataset based on their content using the description generated by our system.

4.6 Hardware Resources Required

Sr. No.	Parameter	Minimum Requirement	Justification
1	CPU Speed	2 GHz	To achieve least Run-time of algorithm
2	RAM	6 GB	To process computationally intensive tasks
4	GPUs	NVIDA Tesla k40/ GTX	To parallelize training

Table 4.1: Hardware Requirements

4.7 Software Resources Required

Platform :

1. Operating System: Linux (Ubuntu version 14 and above)
2. IDE: Eclipse
3. Programming Language:Python with caffe, chainer framework.

CHAPTER 5
PROJECT PLAN

5.1 Project Estimates

5.1.1 Reconciled Estimates

5.1.1.1 Cost Estimate: Nil , Provided with GPU facility

5.1.1.2 Time Estimate: 150 hours for training on GPUs (3 hours per epoch), 45 days for implementation and testing after training

5.1.2 Project Resources

1. People: Internal and external guide,project development and testing team.

2. Hardware: GPU.

3. Software: UNIX based operating system like Ubuntu 14.04, Caffe Framework , Chainer Framework, FFmepg utility.

5.2 Risk Management w.r.t. NP Hard analysis

This section discusses Project risks and the approach to managing them.

5.2.1 Risk Identification

1. Have top software and customer managers formally committed to support the project?

   Yes

2. Are end-users enthusiastically committed to the project and the system/product to be built?

   Yes

3. Are requirements fully understood by the software engineering team and its customers?

   Yes

4. Have customers been involved fully in the definition of requirements?

   Yes

5. Do end-users have realistic expectations?

   No

6. Does the software engineering team have the right mix of skills?

   Yes

7. Are project requirements stable?

   To a certain extent,Yes

8. Is the number of people on the project team adequate to do the job?

   Yes

9. Do all customer/user constituencies agree on the importance of the project and on the requirements for the system/product to be built?

   Yes

5.2.2 Risk Analysis

The risks for the Project can be analyzed within the constraints of time and quality

5.2.3 Overview of Risk Mitigation, Monitoring, Management

Following are the details for each risk.

ID	Risk Description	Probability	Impact		
			Schedule	Quality	Overall
1	Failure to meet the given deadlines for the project	Low	Low	High	High
2	Risk assosiated with the new technology used	Low	Low	High	High
3	Risk assosiated with competitors working for the same problem usig different approaches	Medium	Low	Low	Medium
4	Risk assosiated with use of long videos	Low	Low	High	High
5	Risk assosiated with use of videos which are out-side the specified scope of our system	Medium	Low	High	High

Table 5.1: Risk Table

Probability	Value	Description
High	Probability of occurrence is	$> 75\%$
Medium	Probability of occurrence is	$26 - 75\%$
Low	Probability of occurrence is	$< 25\%$

Table 5.2: Risk Probability definitions:

Impact	Value	Description
Very high	$> 10\%$	Schedule impact or Unacceptable quality
High	$5 - 10\%$	Schedule impact or Some parts of the project have low quality
Medium	$< 5\%$	Schedule impact or Barely noticeable degradation in quality Low Impact on schedule or Quality can be incorporated

Table 5.3: Risk Impact definitions [?]

Risk ID	1
Risk Description	Use of long videos
Category	Development Environment
Source	Software requirement Specification document
Probability	Low
Impact	High
Response	Mitigate
Strategy	Use of long videos
Risk Status	Occurred and solved

Risk ID	2
Risk Description	Change in requirements
Category	Requirements
Source	Software Design Specification documentation review.
Probability	Low
Impact	Medium
Response	Mitigate
Strategy	Modular programming and testing for easy incorporation of changes
Risk Status	Identified

Risk ID	3
Risk Description	Use of videos which are outside the specified scope of our system
Category	Development Environment
Source	Software requirement Specification document
Probability	Low
Impact	High
Response	Mitigate
Strategy	Use of relevant videos
Risk Status	Occurred and solved

5.3 Project Schedule

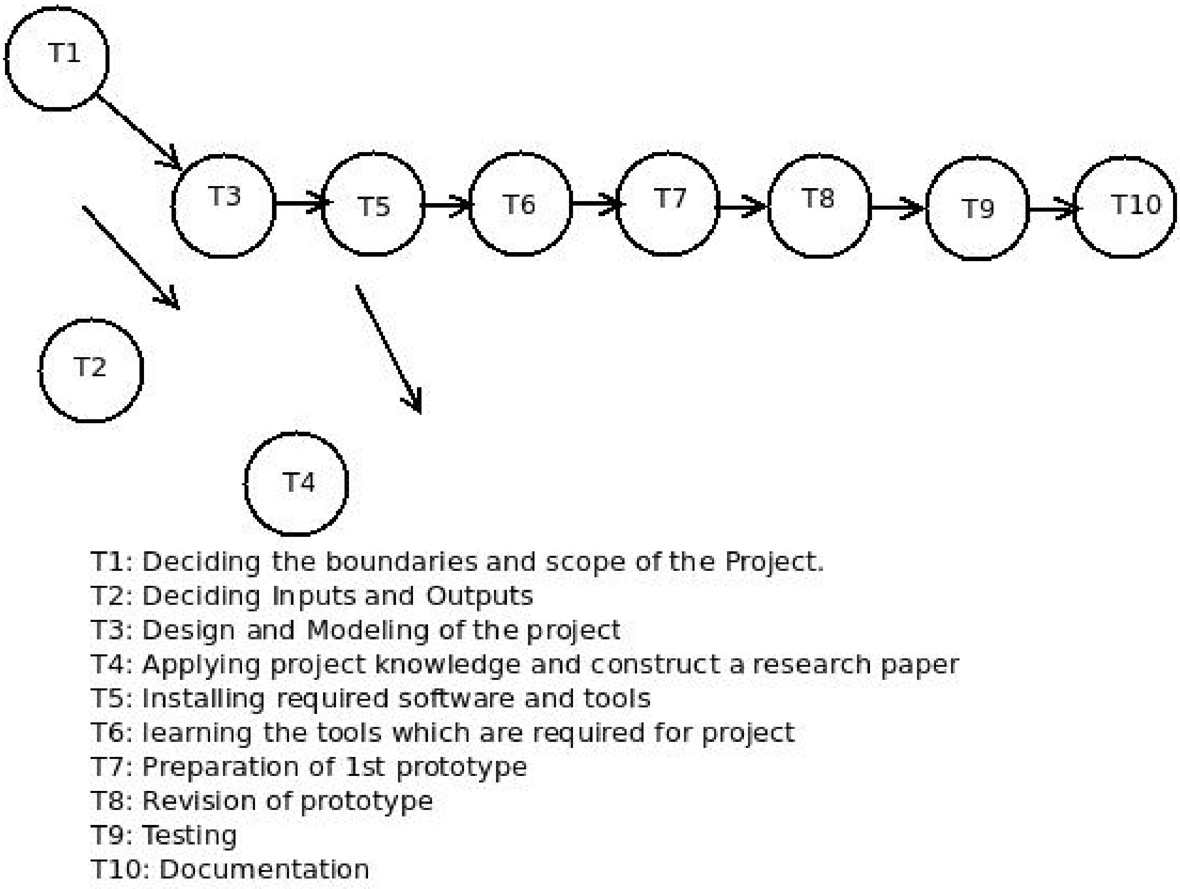
5.3.1 Project task set

Major Tasks in the Project stages are:

- Task 1 Deciding the boundaries and scope of the Project.
- Task 2: Deciding Inputs and Outputs.
- Task 3: Design and Modeling of the project
- Task 4: Applying project knowledge and construct a research paper.
- Task 5: Installing required software and tools
- Task 6: Learning the tools which are required for project.
- Task 7: Preparation of prototype.
- Task 8: Revision of prototype.
- Task 9: Testing.
- Task 10: Documentation.

5.3.2 Task network

Project tasks and their dependencies are noted in this diagrammatic form.



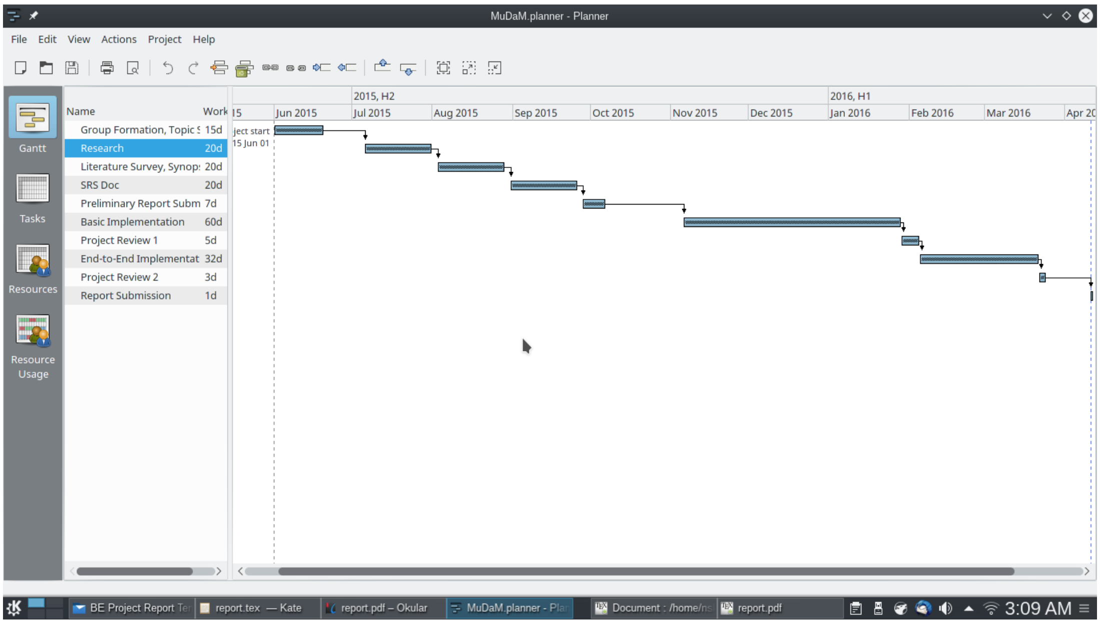
T1: Deciding the boundaries and scope of the Project.
T2: Deciding Inputs and Outputs
T3: Design and Modeling of the project
T4: Applying project knowledge and construct a research paper
T5: Installing required software and tools
T6: learning the tools which are required for project
T7: Preparation of 1st prototype
T8: Revision of prototype
T9: Testing
T10: Documentation

Figure 5.1: Task dependencies

5.3.3 Timeline Chart

A project timeline chart is presented below:

Sr.No	Deliverable	Submission Date	Review Date
1.	Group Formation, Topic search	1st week of June	2nd week of June.
2.	Research on finalised topic and requirement gathering	2nd week of July	1st week of August.
3.	Detail problem definition and synopsis review	1st week of August	1st week of September.
4.	SRS, High Level Design document	1st week of September.	Last week of September.
5.	Submission of 1st term report	1st week of October.	1st week of October.
6.	Low Level Software Design	1st week of November.	last week of November.
7.	Coding	1st week of December.	1st week of February.
8.	Testing	1st week of February.	1st week of March.
9.	Submission of final report	1st week of June.	1st week of June.



Name	Work
Group Formation, Topic S	15d
Research	20d
Literature Survey, Synop:	20d
SRS Doc	20d
Preliminary Report Subm	7d
Basic Implementation	60d
Project Review 1	5d
End-to-End Implementat	32d
Project Review 2	3d
Report Submission	1d

Figure 5.2: Timeline Chart

5.4 Team Organization

5.4.1 Team structure

The project is being worked upon by a team of 6 people (1 project internal guide, 1 external guide and 4 project developers). Each project developer is aware of the entire working of the project. This is possible due to the fact that the project group is small. Thus distribution of work is according to the need of the hour. It is decided to keep the team structure highly flexible throughout the project. Each individual shall contribute equally through all the phases of the project namely Prob- lem Definition, Requirements Gathering and Analysis, Design, Coding, Testing and Documentation.

Role	Participant	Responsibilities
Internal Guide	Prof.S.S.Sonawane	Guidance for the project, Monitor the project plan, Feedback and corrective action plans
External Guide	Dr. Parag Kulkarni	Guidance for the project, Feedback and corrective action plans, Focus the team on project objectives.
Project Members	1. Omkar Acharya 2. Parag Ahivale 3. Nehal Belgamwar 4. Gurnur Wadhwani	Decide problem definition, requirement and risk analysis; Communicate project goals, status of the project; Assure quality of product that will meet the project goals; Design and coding of project.

5.4.2 Management reporting and communication

Communication took place through mails and personal meetings and the tasks were monitored personally. The management was reported every once in 7 days.

CHAPTER 6

SOFTWARE REQUIREMENT SPECIFICATION

6.1 Introduction

6.1.1 Purpose and Scope of Document

The purpose of this document is to present a detailed description of the system. It will explain the purpose and features of the system, the constraints and the functions. The scope of the document is to collect and organize all assorted ideas that have come up to define the system and to provide a detail overview of our product, its parameters and goals. It also describes the hardware and software requirements. It would also help any organizer and developers to assist in software delivery life-cycle (SDLC) processes.

6.1.2 Overview of responsibilities of Developer

Developer can carry out the following responsibilities:
1. Analysing the requirements of the project.
2. Design for recommendations for the workload specified.
3. Provide an easy and effective user interface.
4. Coding and testing the tool with predefined results and check for efficiency.

6.2 Usage Scenario

This section provides various usage scenarios for the system to be developed.

6.2.1 User profiles

1.User: The user will provide video input to the system
2.Frame Generator: This will generate frames from the input video
3.CNN: CNN will extract features from the generated frames and perform mean pooling
4.RNN: RNN will generate natural language description of the input video.

6.2.2 Use-cases

All use-cases for the software are presented. Description of all main Use cases using use case template is to be provided.

Sr No.	Use Case	Description	Actors
1	Input Videos	The user provides the input video to the system	user
2	Generate Frames	The frames are generated from the input video	Frame Generator and CNN
3	Input images to CNN	The generated frames are given to CNN as input	Frame Generator
4	Feature Extraction	Features are extracted from the given frames	CNN and RNN
5	Generate video description	The semantically correct description of videos is generated	RNN

Table 6.1: Use Cases

6.2.3 Use Case View

Use Case Diagram. Example is given below

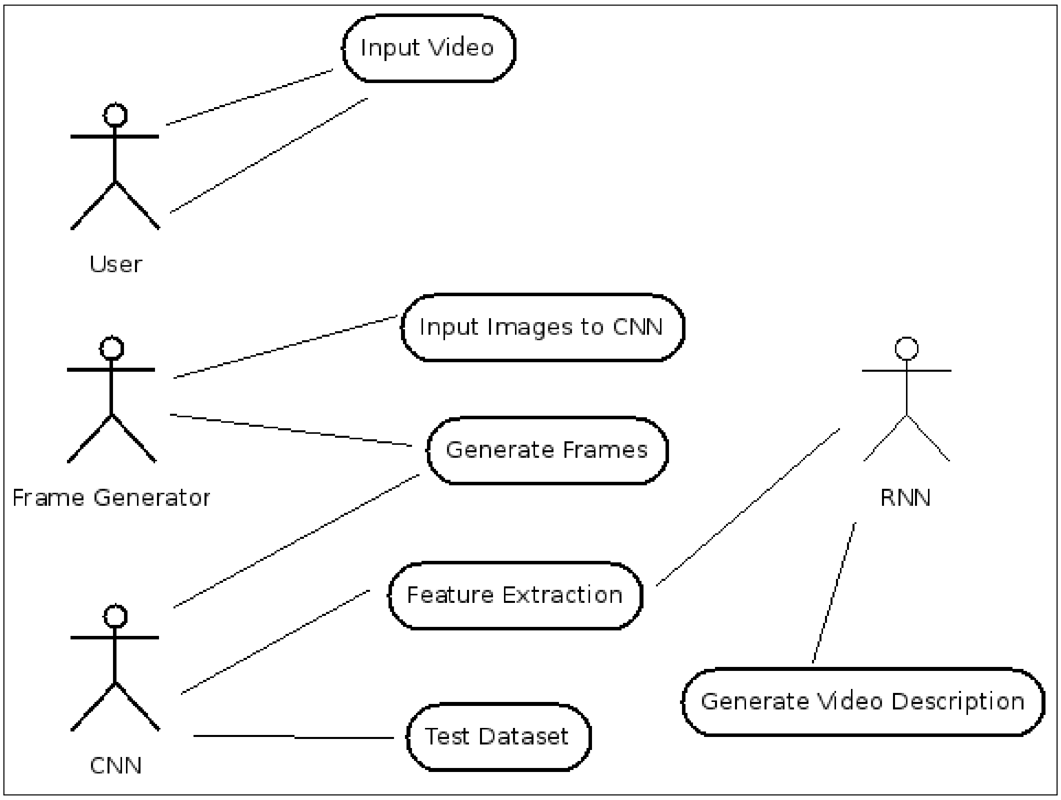


Figure 6.1: Use case diagram

6.3 Data Model and Description

6.3.1 Data Description

Data objects that will be managed/manipulated by the software are described in this section. Data to be used:
1. Image and video corpus
2. Images (Extracted Frames from videos) of size: 224*224 pixels.
3. Data Structure to store the image:
Numpy Array of size 224*224*3
4. Feature vector of size 4096

6.3.2 Data objects and Relationships

Data objects and their major attributes and relationships among data objects are described using an ERD- like form.

6.4 Functional Model and Description

A description of each major software function, along with data flow (structured analysis) or class hierarchy (Analysis Class diagram with class description for object oriented system) is presented.

6.4.1 Data Flow Diagram

6.4.1.1 Level 0 Data Flow Diagram

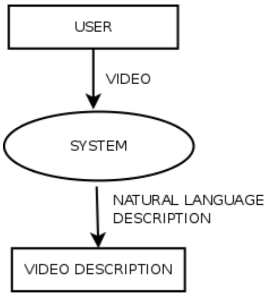


Figure 6.2: DFD0

6.4.1.2 Level 1 Data Flow Diagram

6.4.2 Description of functions

1) Obtain dataset: Get dataset for training.
2)Convert video to image frames.
3)Obtain the image features by passing the image through Convolutional Neural Network.
4)Obtain the image description by passing the image through Recurrent Neural Network.
5)Train the system.
6)Test the system.

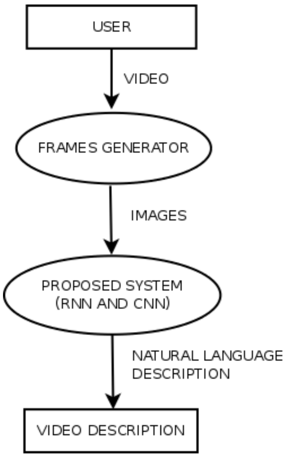


Figure 6.3: DFD1

6.4.3 Activity Diagram:

- The Activity diagram on the following page represents the steps taken.

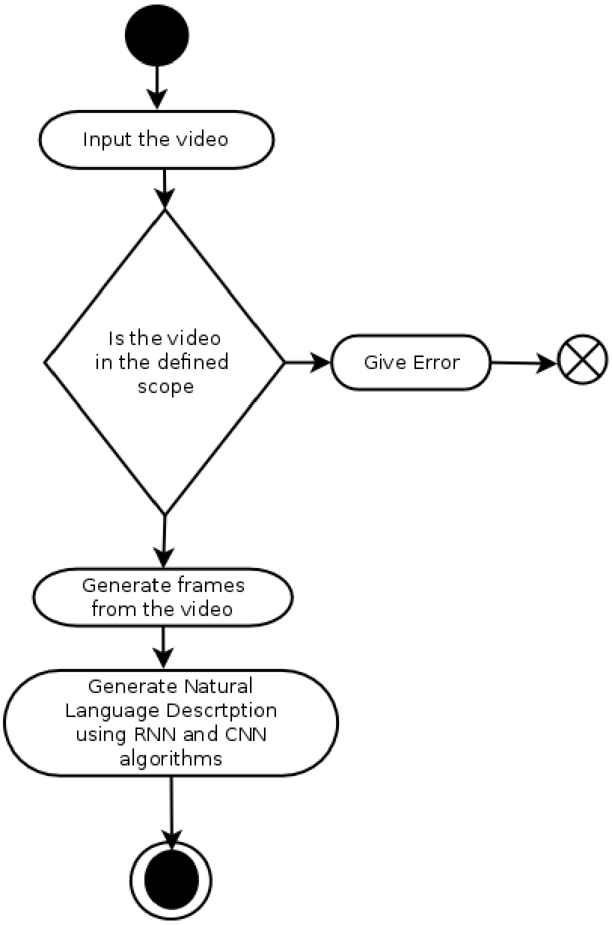

Input the video
Is the video in the defined scope
Give Error
Generate frames from the video
Generate Natural Language Descrtption using RNN and CNN algorithms

Figure 6.4: Activity diagram

6.4.4 Non Functional Requirements:

- The system produces the video description in finite duration.
- The system handles out of scope videos by gracefully throwing errors.
- The system works correctly for video of any format.

6.4.5 State Diagram:

State Transition Diagram
Fig.6.5 The states are represented in ovals and state of system gets changed when certain events occur. The transitions from one state to the other are represented by arrows. The Figure shows important states and events that occur while creating new project.

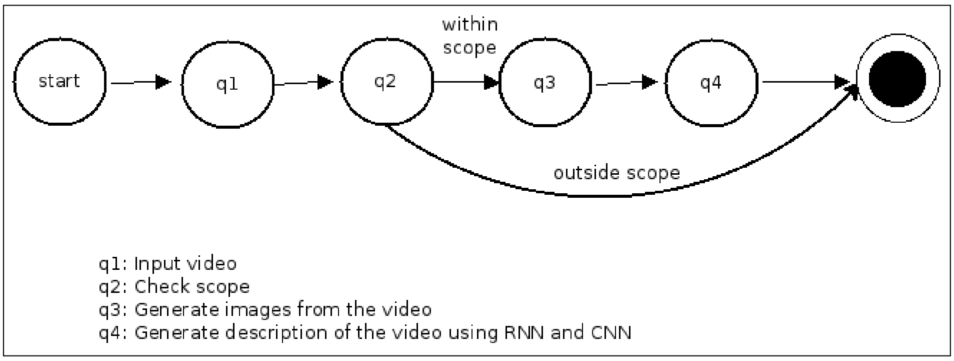

start
q1
q2
within scope
q3
q4
outside scope
q1: Input video
q2: Check scope
q3: Generate images from the video
q4: Generate description of the video using RNN and CNN

Figure 6.5: State transition diagram

6.4.6 Design Constraints

1. Use of short videos
2. Use of videos which are within the specified scope of our system.

6.4.7 Software Interface Description

To be able to get a semantically correct description of a video uploaded through a web application through browser.

CHAPTER 7

DETAILED DESIGN DOCUMENT USING APPENDIX A AND B

7.1 Introduction

This document specifies the design that is used to solve the problem.

7.2 Architectural Design

A description of the program architecture is presented. Subsystem design or Block diagram,Package Diagram,Deployment diagram with description is to be presented.

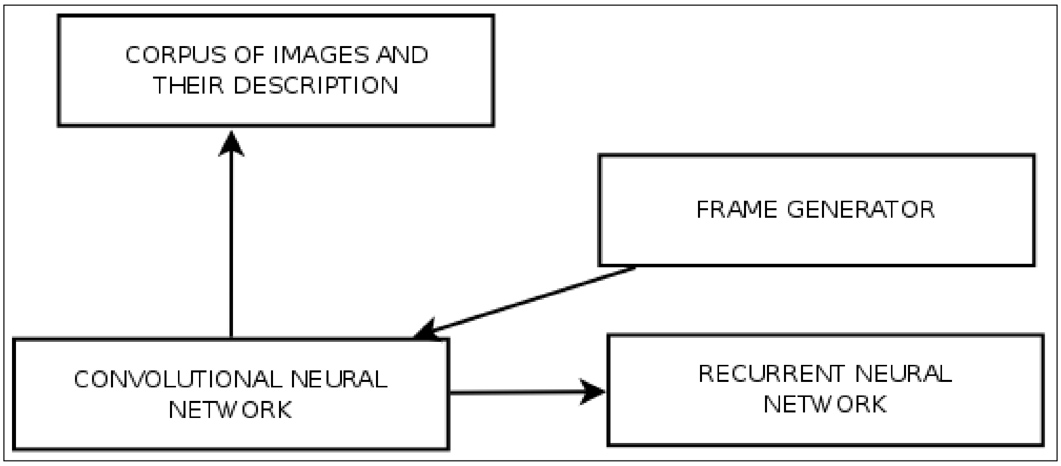

CORPUS OF IMAGES AND THEIR DESCRIPTION
FRAME GENERATOR
CONVOLUTIONAL NEURAL NETWORK
RECURRENT NEURAL NETWORK

Figure 7.1: Architecture diagram

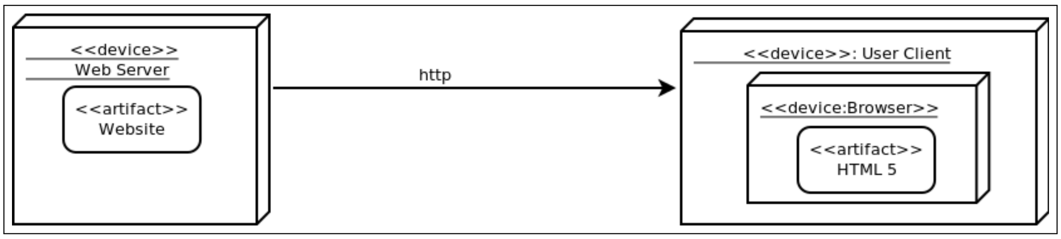


Figure 7.2: Deployment Diagram

7.3 Data design (using Appendices A and B)

A description of all data structures including internal, global, and temporary data structures, database design (tables), file formats.

7.3.1 Internal software data structure

Numpy array of size 224*224*3 to store images. Array of size 4096 to store the features.s

7.3.2 Global data structure

Numpy array which stores image.

7.3.3 Temporary data structure

Pickle file containing extracted features of training dataset. Chainer file to store the model weights.

7.3.4 Database description

Training Data-set containing images along with their description.
Test Data-set containing images along with their description.

7.4 Component Design

Class diagrams, Interaction Diagrams, Algorithms. Description of each component description required.

7.4.1 Class Diagram

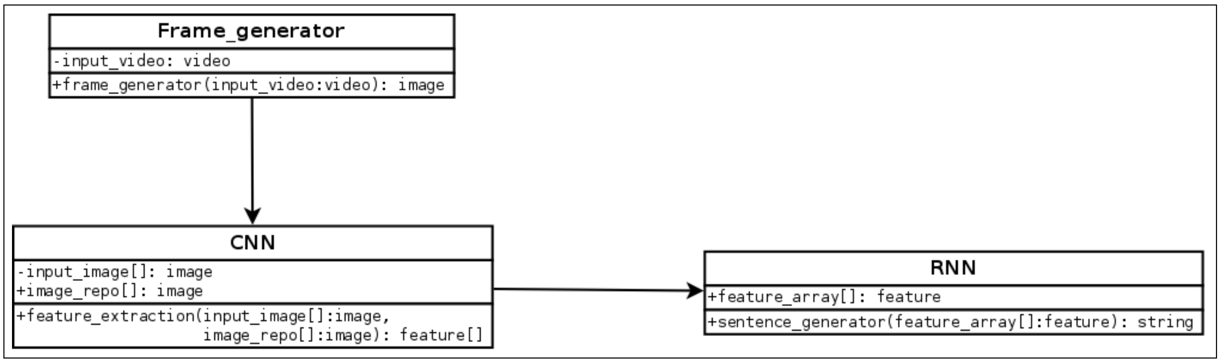

Frame_generator
-input_video: video
+frame_generator(input_video:video): image
CNN
-input_image[]: image
+image_repo[]: image
+feature_extraction(input_image[]:image, image_repo[]:image): feature[]
RNN
+feature_array[]: feature
+sentence_generator(feature_array[]:feature): string

Figure 7.3: Class Diagram

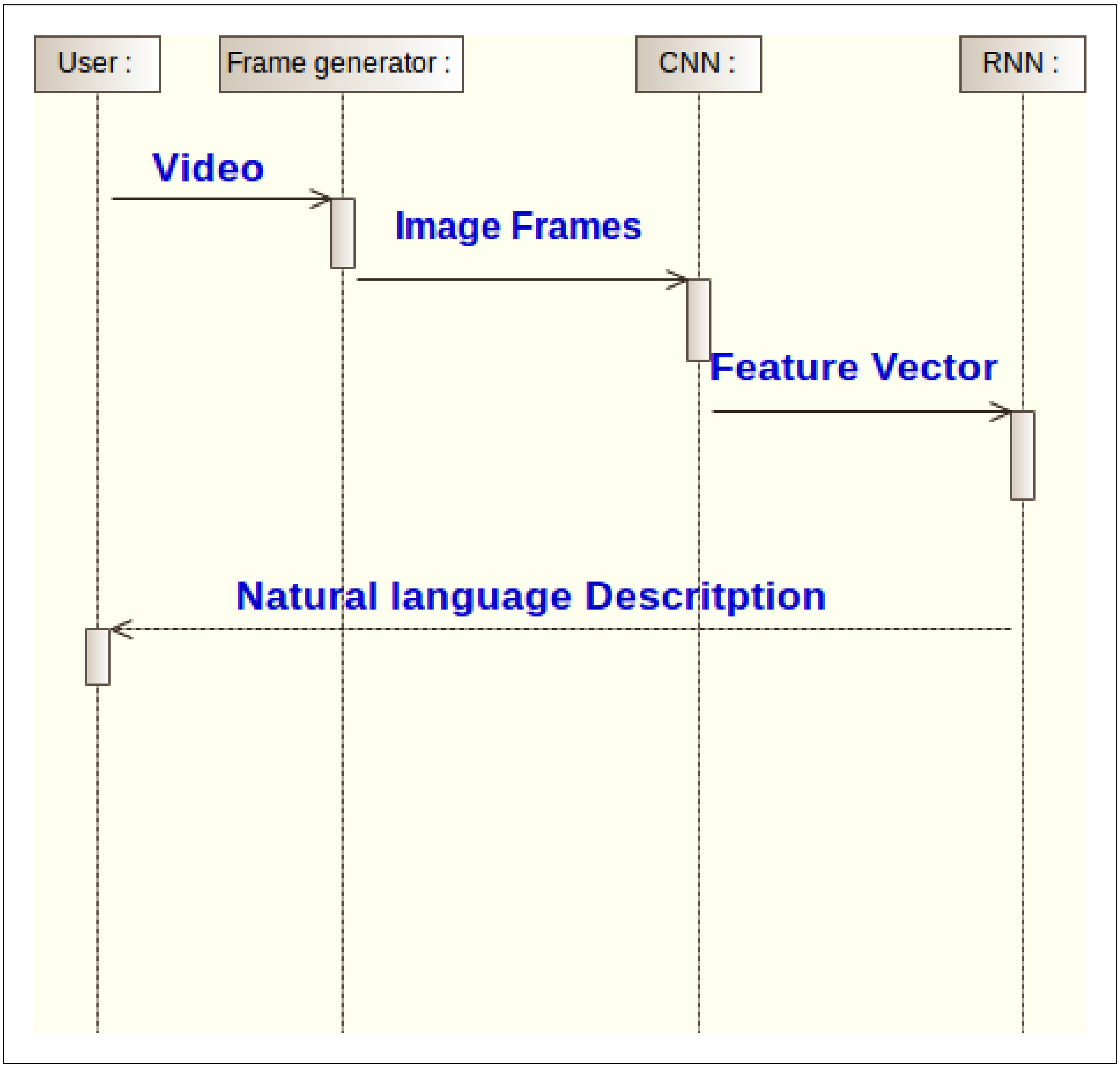


Figure 7.4: Sequence Diagram

CHAPTER 8

PROJECT IMPLEMENTATION

8.1 Introduction

We began the project implementation by dividing it into modules. The first module deals with the extraction of images from the videos and preparing the image for the actual usage by the Deep learning algorithm CNN. The second module is the implementation of the CNN algorithm for extracting the relevant features out of that image. The third module is the implementation of Long Short Term Memory, an RNN algorithm, for predicting the classes of higher probabilities for that particular image. The last module deals with the summarizing of all the words together and frame the meaningful sentence out of them.

8.1.1 Tools and Technologies Used

1. Intel i5/i7 processor
2. RAM min. 4 GB
3. Ubuntu 14.04 or higher
4. NVIDIA GPU card
5. Caffe - Image feature extraction framework
6. Chainer - An LSTM Framework
7. ffmpeg - For frame extraction from a video
8. Numpy
9. Scipy

8.1.2 Methodologies/Algorithm Details

Our system uses two deep learning algorithms viz. Convolutional Neural Networks and Long Short Term Memory (LSTM)

8.1.3 CNN Algorithm:

1. INPUT [32x32x3] will hold the raw pixel values of the image, in this case an image of width 32, height 32, and with three color channels R,G,B.

2. CONV layer will compute the output of neurons that are connected to local regions in the input, each computing a dot product between their weights and a small region they are connected to in the input volume. This may result in volume such as [32x32x12] if we decided to use 12 filters.

3. RELU layer will apply an elementwise activation function, such as the max(0,x)max(0,x) thresholding at zero. This leaves the size of the volume unchanged ([32x32x12]).

4. POOL layer will perform a downsampling operation along the spatial dimensions (width, height), resulting in volume such as [16x16x12].

5. FC (i.e. fully-connected) layer will compute the class scores, resulting in volume of size [1x1x1000], where each of the 1000 numbers correspond to a class score, such as among the 1000 categories of ImageNet.

6. For our convenience, we remove the FC7 layer in our architecture, and give the generated output to the next algorithm i.e. RNN

8.1.4 RNN Algorithm:

8.2 Verification and Validation for Acceptance

8.2.1 Verification:

Verification is related to the process of ensuring the product is built right. The project was verified after every module was built to check its functionality and such that it provides the maximum performance and efficiency and is bug free. Verification was doing using unit testing method to check the module for any errors/bugs.

1. The system is valid only for the videos with high resolution. The system doesn't work for black-white videos.

2. The proposed system provides us with the validation accuracy of 93.05 % against a training set data of 1.2M images.

3. The above system has corresponding test accuracy of 78.63 %.

8.2.2 Validation:

Validation is related to the process of ensuring that the right product is built. During the entire project, the customer (in this case, iKnowlation Research Labs) was consulted at different levels of the product development process so that the system that is built is in accordance with the specified requirements.

CHAPTER 9

SOFTWARE TESTING

9.1 Type of Testing Used

9.1.1 Unit Testing:

Two major processing modules of the system are tested independently as a part of unit testing.

- **CNN module:** Caffe, a deep learning framework was used for feature extraction in textual form of an image. A VGG-16 model was trained for ImageNet dataset. Standard images were tested and 93% accuracy was obtained for standard objects.
- **RNN module:** Based on the extracted features from CNN, a meaningful sentences were created to get a brief description about the video.

9.1.2 Integration Testing

Two individually tested components of the system are integrated together. Integration testing is done to check the performance of the modules after integration. The complete end to end system was tested here.

9.1.3 High Order Testing

High order testing was done after completely integrating all the modules of the system using various test cases.

9.2 Test Cases and Test Results

9.2.1 Unit Testing:

Sr. No.	Test Case	Expected Output	Actual Output
1	Input File type :- Movie	Description is generated	Description is generated (Process takes too much time.)
2	Input File type:- Video clip	Description is generated	Description is generated
3	Input File format :- .mkv,.flv,mp4 etc.	Description is generated irrespective of the format	Description is generated irrespective of the format.
4	Out of scope video	Random description	Description is generated using random words fro m the dictionary.

Table 9.1: Unit Testing

9.2.2 High Order Testing

Sr. No.	Test Case	Expected Output	Actual Output
1	complete end-to-end testing	The system is able to generate the correct description	Success

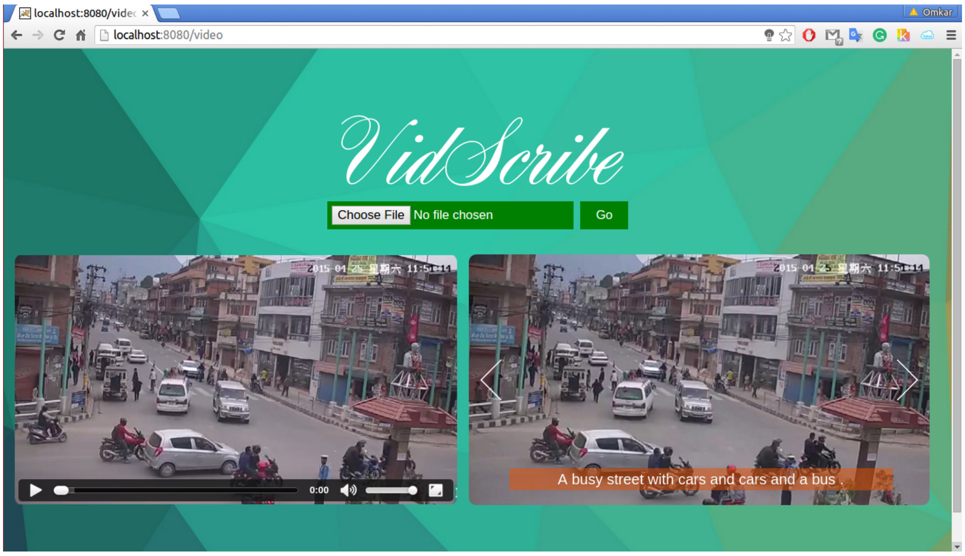
Table 9.2: High Order Testing

CHAPTER 10

RESULTS

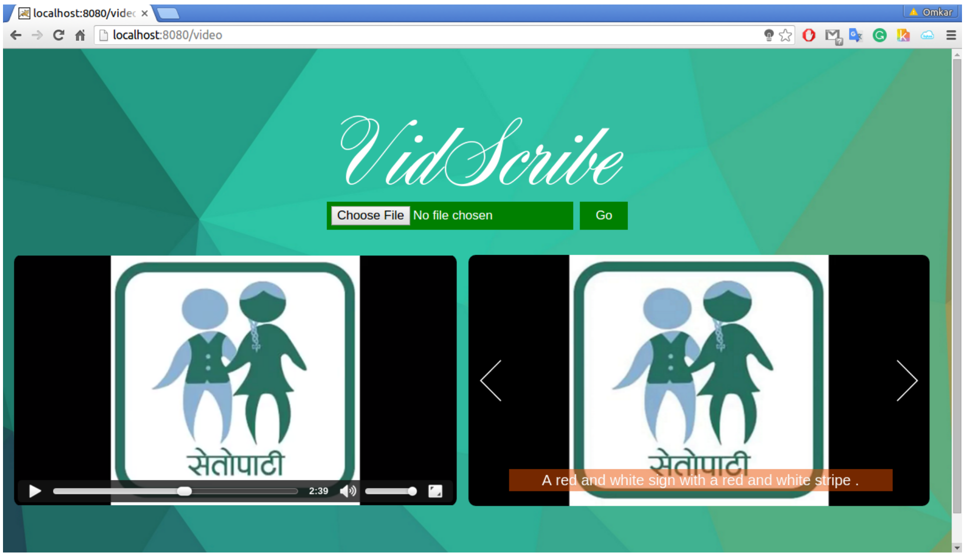
10.1 Screen shots

1. **Input video with better accuracy**



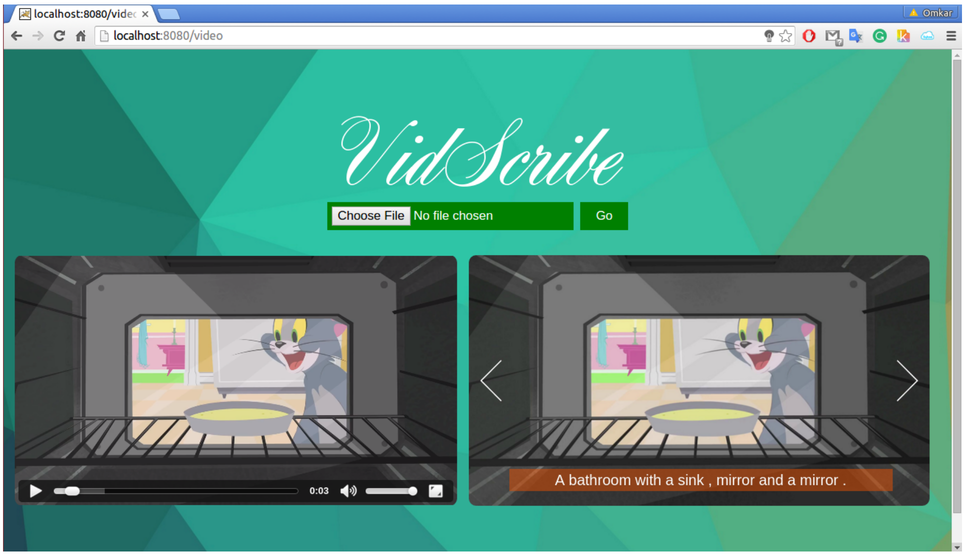
A busy street with cars and cars and a bus .

2. **Out of scope video as input**



A red and white sign with a red and white stripe .

3. **Animated video as input**



A bathroom with a sink , mirror and a mirror .

CHAPTER 11

DEPLOYMENT AND MAINTENANCE

11.1 Installation and un-installation

1. Installation of Apache Tomcat.
   (a) Download The Apache-Tomcat latest version tar file.
   (b) Untar it.
   (c) Go to bin folder of Apache Tomcat directory.
   (d) Start tomcat server by giving './startup.sh'.
2. Installation of Theano on Ubuntu: pip install theano
3. Installation of Numpy on Ubuntu: pip install numpy
4. Installation of Scipy on Ubuntu : pip install scipy

11.2 User help

1. Give the URL.
2. Enter the location.
3. Enter the time frame from the drop-down menu.
4. Click on submit to get the prediction on the image for that area.

CHAPTER 12

CONCLUSION AND FUTURE SCOPE

1. **Summary**

   Through this report, various parameters that are required to complete this project have been described. It encompasses the mathematical model, the literature survey, the software development life cycle description as well as the SRS document.

   The synopsis gives a brief description of the proposed project.Through the literature survey, it was possible to find the most efficient algorithm for the proposed application.The mathematical model helped in deciding the flow of project development and also identified the system requirements such as input, output and functions. The proposed project scope has fulfilled all the requirements.

   The project plan described the processes which have been followed and the different requirements that need to be fulfilled for successful completion of the project.Risk analysis helped in finding out the various risks related to the requirements gathered and the process followed till then.The various diagrams describe the various processes that were followed for proper completion of project.

2. **Conclusion**

   The CNN architecture proposed for the system has 11 layers through which we have obtained the validation acuracy of 94.15% on a minimal dataset of 4000 images.The training was conducted on Titan GeForce GPU's provided by NVIDIA which helped in increasing the training rate immensely.The existing technique for predicting the travel time involves the use of GPS of users to indicate how much time the user takes to move from one place to another.Our aim to eliminate the need for human intervention in traffic density prediction has been achieved successfully.

3. **Future Scope**

   (a) Training the system on larger datasets.

   (b) Allocating more GPU training time.

   (c) Real-time traffic prediction system.

   (d) Integrating the system with the various traffic management systems.

ANNEXURE A
REFERENCES

[1] Zhen Dong, Yuwei Wu , Mingtao Pei , Yunde Jia , Vehicle Type Cla

[2] Wasin Thubsaeng, Aram Kawewong, Karn Patanukhom, Vehicle Logo D
Histogram of Oriented Gradients , 11th International Joint Conferenc
(JCSSE)

[3] Junho Yim, Jeongwoo Ju, Heechul Jung, and Junmo Kim, Image Clas

[4] Tianjun Xiao, Jiaxing Zhang, Kuiyuan Yang, Yuxin Peng and Zheng Zh
2014

[5] Luiz G. Hafemann, Luiz S. Oliveira, Paulo Cavalin
Forest Species Recognition using Deep Convolutional Neural Network
IEEE 2014 22nd International Conference on Pattern Recognition (ICPR)

[6] Soumya T Soman, Ashakranthi Nandigam, V. Srinivasa Chakravarthy,
2013 IEEE

[7] A. B. Chan and N. Vasconcelos, "Probabilistic Kernels for the Cla

[8] Source: http://www.wsdot.wa.gov/

ANNEXURE B

LABORATORY ASSIGNMENTS ON PROJECT ANALYSIS OF ALGORITHMIC DESIGN

- To develop the problem under consideration and justify feasibilty using concepts of knowledge canvas and IDEA Matrix.

  Problem Statement Feasibility:
  The product is a system which has learnt to identify which images have high traffic density and which have low traffic density. It is designed using deep learning technique for high accuracy. It will not be a real time system, but this can be implemented in the module of a real time system so as to assist in predicting traffic density as per time of the day and hence provide driving assistance to save valueable time of the people today. It may be used for providing recommendations regarding the traffic.
  The risks identified are associated with the new technology used and competitors working for the same problem using different approaches.We plan to mitigate them by using latest and compatible versions of eclipse as well as Theano and incorporating modular programming and testing for easy induction of changes.

  Training Problem:
  Works have shown that training a Neural Network is known to be an NP-Complete Problem if it is meant to give correct output for two-thirds of the training samples. While NP-completeness does not render a problem totally inapproachable in practice, and does not address the specific instances one might wish to solve, it often implies that only small instances of the problem can be solved exactly, and that large instances at best can be solved approximately even with large amounts of computer time.

idea_matrix.png

ANNEXURE C

LABORATORY ASSIGNMENTS ON PROJECT QUALITY AND RELIABILITY TESTING OF PROJECT DESIGN

It should include assignments such as

- Use of divide and conquer strategies to exploit distributed/parallel/-concurrent processing of the above to identify object, morphisms, overloading in functions (if any), and functional relations and any other dependencies (as per requirements). It can include Venn diagram, state diagram, function relations, i/o relations; use this to derive objects, morphism, overloading

- Use of above to draw functional dependency graphs and relevant Software modeling methods, techniques including UML diagrams or other necessities using appropriate tools.

- Testing of project problem statement using generated test data (using mathematical models, GUI, Function testing principles, if any) selection and appropriate use of testing tools, testing of UML diagram's reliability. Write also test cases [Black box testing] for each identified functions. You can use Mathematica or equivalent open source tool for generating test data.

- Additional assignments by the guide. If project type as Entreprenaur, Refer [?],[?],[?], [?]

ANNEXURE D

REVIEWERS COMMENTS OF PAPER SUBMITTED

1. Paper Title: CNN for Vehicle Density Classifcation
2. Name of the Conference/Journal where paper submitted : National Conference on Recent Trends in Computer Engineering, PICT, Pune
3. Paper accepted/rejected : Published in Proceedings
4. Review comments by reviewer : Best Paper Award
5. Corrective actions if any : -

ANNEXURE E
PLAGIARISM REPORT

Plagiarism report

ANNEXURE F

TERM-II PROJECT LABORATORY ASSIGNMENTS

1. Review of design and necessary corrective actions taking into consideration the feedback report of Term I assessment, and other competitions/conferences participated like IIT, Central Universities, University Conferences or equivalent centers of excellence etc.

2. Project workstation selection, installations along with setup and installation report preparations.

3. Programming of the project functions, interfaces and GUI (if any) as per 1 st Term term-work submission using corrective actions recommended in Term-I assessment of Term-work.

4. Test tool selection and testing of various test cases for the project performed and generate various testing result charts, graphs etc. including reliability testing.
   Additional assignments for the Entrepreneurship Project:

5. Installations and Reliability Testing Reports at the client end.

ANNEXURE G

INFORMATION OF PROJECT GROUP MEMBERS

1. Name : Supratika Banerjee
2. Date of Birth : 7th August, 1994
3. Gender : Female
4. Permanent Address : Flat no. E-8, Konark Park, 206, Dhole Patil Road, Pune - 411001
5. E-Mail : bansup7@gmail.com
6. Mobile/Contact No. : 9860648564
7. Placement Details : Campus Placement- Accenture
8. Paper Published : Proceedings of NCRTCE.(National Conference)

1. Name : Himani Deshpande
2. Date of Birth : 11-07-1994
3. Gender : Female
4. Permanent Address :'Ishawasyam',82/1B/2, Aranyeshwar, Pune-9
5. E-Mail :hiatde11@gmail.com
6. Mobile/Contact No. :7798573039
7. Placement Details : Opted out of placements.
8. Paper Published : Proceedings of NCRTCE.(National Conference)

1. Name : Shubham Koshti
2. Date of Birth :20-09-1994
3. Gender : Female
4. Permanent Address :Dream Citi ,Orange bldg flat 303,opp fame cinema,ring road,nashik-pune road ,nashik
5. E-Mail : kostishubham@gmail.com
6. Mobile/Contact No. :9890462011
7. Placement Details :Yardi
8. Paper Published : Proceedings of NCRTCE.(National Conference)

1. Name : Sweta Kumari
2. Date of Birth :30-03-1994
3. Gender : Female
4. Permanent Address :Flat no 7, Sukamal Plaza, near gurudwara, airforce road, ojhar, Nashik
5. E-Mail : brilli.sweta@gmail.com
6. Mobile/Contact No. : 9561049275
7. Placement Details :Cybage
8. Paper Published : Proceedings of NCRTCE.(National Conference)