Society for Computer Technology & Research's

PUNE INSTITUTE OF COMPUTER TECHNOLOGY

(Approved by A.I.C.T.E. & Government of Maharashtra, affiliated to the Savitribai Phule Pune University & Accredited by NBA [All UG Programs] & NAAC)

Survey No. 27, Near Trimurti Chowk, Dhankawadi, Pune - 411043.

Tel. : +91 20 24371101/8063 **Fax No : +91 20 24364741**

Website : www.pict.edu **E-mail : principal@pict.edu**

EASEBUZZ		TRANSACTION ID : **EFS08NYP3U**	DATE : **2022-07-20**
PAID TO (INSTITUTE NAME) **PUNE INSTITUTE OF COMPUTER TECHNOLOGY**		STUDENT :	**MANDAL IMON ASHOK**
		Reg. No. : Roll No :	**C2K19106274** **41338**
Class :	**1-COMP**	Batch :	**BE-COMP-2022-23**

PARTICULARS

Installment	**Amount (Rs)**	**Penalty (Rs)**	**Total (Rs)**
Tution fees	44484.0	0.0	44484.0
Development fees	11033.0	0.0	11033.0
University fees	1218.0	0.0	1218.0
Amount (In Words) Fifty-six thousand, seven hundred and thirty-five Only		**Total (Rs)**	**56735.00**

Note :- *Subject to change from FRA.*
Computer generated receipt does not require seal or signature
All disputes are subject to Pune jurisdiction

This receipt is valid subject to bank realization.

Issued By: PICT (PUNE INSTITUTE OF COMPUTER TECHNOLOGY)

SAVITRIBAI PHULE PUNE UNIVERSITY

(formerly University of Pune)

GANESHKHIND PUNE 411 007

STATEMENT OF MARKS / GRADES FOR T.E.(2019 COURSE) EXAM., APR/MAY 2022
BRANCH CODE:42-T.E.(2019 PAT.)(COMPUTER)

SEAT NO. T190054377 CENTRE PICT[5] PERM. REG. NO.: 72017026F

NAME MANDAL IMON ASHOK MOTHER: PRATIMA MANDAL

COLLEGE / SCHOOL

[CEGP010440] - PUNE INSTITUTE OF COMPUTER TECHNOLOGY, PUNE.

COURSE CODE	COURSE NAME	CO. TYPE	TOT. CRD	EARN. CRD	GRD	CRD. PTS
SEM.:1						
310241	DATABASE MANAGEMENT SYSTEMS	TH	03	03	O	30
310242	THEORY OF COMPUTATION	TH	03	03	O	30
310243	SYS. PROG & OPERATING SYS.	TH	03	03	O	30
310244	COMPUTER NETWORKS AND SEC.	TH	03	03	O	30
310245C	DISTRIBUTED SYSTEMS	TH	03	03	O	30
310249	SEMINAR AND TECH. COMN.	TUT	01	01	A+	09
310246	DATABASE MGMT. SYS. LAB.	PR	02	02	O	20
310248	LABORATORY PRACTICE I	PR	02	02	O	20
310247	COMP. NET. AND SEC. LAB.	PR	01	01	O	10
310250A	CYBER SECURITY	AC	00	00	AC	00
SEM.:2						
310251	DATA SCI & BIG DATA ANA.	* TH	03	03	A+	27
310252	WEB TECHNOLOGY	* TH	03	03	A+	27
310253	ARTIFICIAL INTELLIGENCE	* TH	03	03	O	30
310254C	CLOUD COMPUTING	* TH	03	03	O	30
310255	INTERNSHIP	* TW	04	04	O	40
310251	DATA SCI & BIG DATA ANA.	* PR	02	02	O	20
310258	LABORATORY PRACTICE-II	* PR	02	02	O	20
310252	WEB TECHNOLOGY	* PR	01	01	O	10
310259A	DIG. & SOCIAL MEDIA MKT.	* AC	00	00	AC	00

THIRD YEAR SGPA : 9.83, TOTAL CREDITS EARNED : 42

NOTE : PLEASE SEE THE BACKSIDE OF THIS STATEMENT FOR MORE DETAILS.

MEDIUM OF INSTRUCTION : ENGLISH

Director
Board of Examinations & Evaluation

R22092600471 DATE : 12 SEP 2022

12512212025003560649

तहसीलदार कार्यालय हवेली

क्रमांक : ४१९३५८९८७२३
जिल्हा : **पुणे**

१ वर्षासाठी उत्पन्नाचे प्रमाणपत्र

प्रमाणित करण्यात येते की, **श्री. अशोक नरेंद्रनाथ मंडल** राहणार **मोसरी** गाव **पिंपरी चिंचवड (महानगरपालिका)**, तहसील **हवेली**, जिल्हा **पुणे** येथील अर्जदार आहेत. त्यांचे आयकर विवरणपत्र या आधारावर अर्जदार व त्यांच्या कुटुंबातील सर्व सदस्यांचे सर्व मार्गाने व साधनाने मिळालेले १ वर्षाचे उत्पन्न खालील प्रमाणे आहे.

वर्ष	वार्षिक उत्पन्न (₹)	अक्षरी (रुपये)
२०२१ - २०२२	२,८१,०३०	दोन लाख एक्याऐंशी हजार तीस

सदरचा दाखला **श्री. अशोक नरेंद्रनाथ मंडल** यांचा **मुलगा कुमार इमोन अशोक मंडल** यांना **शैक्षणिक प्रयोजन** या कामासाठीच देण्यात येत आहे, तसेच त्यांनी कार्यालयास सादर केलेल्या कागदपत्रांच्या आधारे देण्यात येत आहे.

हे प्रमाणपत्र ३१ मार्च २०२३ पर्यंतच वैध राहील.

सादर केलेल्या दस्तऐवज / पुराव्याचे तपशील

१.आयकर भरण्यासाठी वापरला जाणारा फॉर्म
२.शिधापत्रिकेची प्रमाणित प्रत
३.वीज देयक / मालमत्ता कराची पावती / तलाठ्याने जारी केलेले रहिवास प्रमाणपत्र
४.आधार कार्ड
५.केंद्र चालकाचे घोषणा पत्र
६.स्वघोषणं पत्र

Signature valid
Digitally Signed by Pravin Kisan Dhamale
Date:2022-12-09 19:13:09 PM
तहसीलदार
हवेली

स्थळ : हवेली
दिनांक : ०८/१२/२०२२

Printed By -OMTID :MH032200702 VLE Name :Sandhya Sunil Mhaske, Date:08/12/2022 7:34PM

Aadhaar seeding process

Aadhaar seeding is necessitated for receiving Direct Benefit Transfers (DBT) provided by various Government schemes. The following is the process flow of Aadhaar seeding

1. Customer to visit the bank branch where he / she is holding an account and submit the duly filled consent form – Annexure I
2. The bank officials after verifying the details and documents provided (as may be required) and authenticity of the customer based on the signature will accept Aadhaar seeding consent form and provide an acknowledgement to the customer.
3. The branch will then link the Aadhaar number to the customer's account and also in NPCI mapper.
4. Once this activity is completed and Aadhaar number will reflect in NPCI mapper.

Role of the customer:

1. Submit the consent form with complete details either in physical or electronic form as per the facility provided by his / her bank.
2. In case of moving Aadhaar number from one bank to another bank, the customer should provide the name of the bank from which the Aadhaar is being moved.
3. In case of physical form, the consent form should be duly signed as per the bank records.
4. After seeding is completed the customer may approach their Gas service provider (Oil Marketing Company) for the pending subsidy amount.
5. For non-receipt of subsidies customer to approach respective OMC's through their toll free number : 1800 2333 555

Role of the Bank / Branch:

1. Verifying the completeness of the consent form, checking the documentation and authenticating the customer's signature.
2. After the officials are satisfied with the documentation they should carry out the following activities
    a. Linking the Aadhaar number to the bank account (in CBS)
    b. Updating NPCI mapper

    Note: By linking the Aadhaar number to the account the branch is not updating the mapper. The mapper update process has to be followed by their central team or IT division as the case may be.

3. After the mapper files are uploaded the response files received from NPCI have to be verified.
4. In case of failure in updating any Aadhaar number/s then necessary corrective action has to be taken and CBS also should be updated accordingly.
5. Customer query / complaint handling
    1. Branches should understand that if Aadhaar number is not updated in NPCI mapper the action is purely lies with the bank only. The customer should not be told that NPCI has not updated the Aadhaar number.
    2. Aadhaar number being active in bank's CBS does not mean that mapper file is updated, the branch should not show CBS screen or provide screen shot to the customer confirming seeding.
    3. If the customer complaints, the branch should approach their internal team handling Aadhaar mapping and ascertain the reason for non-updating the Aadhaar in NPCI mapper.
    4. After ascertaining the root cause bank should take corrective action and redress the grievance of the customer.

Responsibility of NPCI:

1. Mapper is a platform provided by NPCI for the banks to update or remove Aadhaar numbers as per their customer's request.
2. The activity of updating or removing an Aadhaar number from mapper can be performed only by the banks.
3. NPCI on its own does not update the mapper records.
4. In case customer approaches NPCI for grievance redressal, NPCI will have to reach out to the teams concerned in banks for necessary action.

5. NPCI will ensure that mapper platform is available, files submitted by banks are processed and response is provided.

Customer grievance:

1. If the Aadhaar number is not reflecting in NPCI mapper after submitting all the relevant documents to the bank the action rests with the bank only.
2. The customer should approach the bank's customer service cell for grievance redressal and follow escalation matrix if the issue is not resolved.
3. If customer wants to write to NPCI then the copy of the consent form duly acknowledged by the bank should be provided for taking up with the bank concerned.
4. For any escalations customer may write to npci.dbtl@npci.org.in with Aadhaar consent acknowledgment copy provided by the bank.

Additional information:

Customer can link only **one account** with Aadhaar at any point of time.

If customer gives consent to multiple banks then subsidy will be credited to the last seeded Bank with which the status is active in NPCI mapper.

If Aadhaar status is inactive, customer to visit respective bank branch in person and submit the duly filled customer consent form.

OMC's to be approached for reinitiating the failed transactions to last seeded bank account.

APPLICATION FOR LINKING/ SEEDING AADHAR NUMBER
AND RECEIVING DBT BENEFITS INTO BANK ACCOUNT-(NPCI MAPPING)*

The Branch Manager, Date:
…….Branch
…….Bank
Dear Sir,
Account No. _____ in A/c Name _____
Linking / Seeding of Aadhaar in NPCI-Mapping for Receiving Direct Benefits

I am maintaining a Bank account No. _____ with your Branch.

2. I submit my Aadhaar number and voluntarily give my consent
to: o Use my Aadhaar Details to authenticate me from UIDAI.
- o Use my Mobile Number mentioned below for sending SMS Alerts to me.
- o Link the Aadhaar Number to all my existing/new/future accounts and customer profile (CIF) with your Bank.

(Signature/Thumb Impression of customer)

OPTION FOR RECEIVING DBT BENEFITS (TICK ONE)

- o I wish to seed my account No. _____ with NPCI mapper to enable me to receive Direct Benefit Transfer (DBT) including LPG Subsidy from Govt. of India (GOI) in my above account. I understand that if more than one Benefit transfer is due to me, I will receive all the benefit transfers in the same account**.(for customer who have not so far seeded account with NPCI Mapper)**
- o I already have an account with _____ (name of Bank) having IIN Number**_____, and seeded with NPCI Mapper for receiving DBT from GOI. **I request you to change my NPCI mapping(DBT Benefit Account)** to my account with your Bank.
- o I already have an account with another bank _____(name of Bank) having IIN Number**_____, and seeded with NPCI Mapper for receiving DBT from GOI. **I do not want to change my NPCI mapping(DBT Benefit Account)** from the existing Bank.
- o I do not wish to seed my accounts from your Bank with NPCI Mapper (**I will not be getting DBT**).

3. I have been explained about the nature of information that may be shared upon authentication. I have been given to understand that my *information submitted to the bank herewith shall not be used for any purpose other than mentioned above, or as per requirements of law.*

4. I hereby declare that all the above information voluntarily furnished by me is true, correct and complete.

Yours faithfully

(Signature/Thumb Impression of customer)

Name :

Mobile No.:

Email:

Encl: Copy of Aadhaar

[if consent sent through BC/BDO/VO]

o I hereby authorise the Banking Correspondent ……...
o I hereby authorise the Sarpanch,/ V.O./B.D.O./ ……..…
to submit the above consent letter to the bank.

(Signature/Thumb Impression of Customer)

*NPCI Mapping : Mapping is a process of associating a Bank with Aadhaar number which is facilitated by NPCI for Direct Benefit Transfer to the respective Bank who have linked the Aadhaar Number to a specific Bank account for receiving Direct Benefits to which customer has given the consent.
** IIN number will be provided by Bank receiving the consent Application

COMMUNICATION THROUGH INDIAN SIGN LANGUAGE : AN APPROACH FOR DATA PREPROCESSING

1st Dr. Prajakta Khadkikar
Dept. of Computer Engg.
Pune Institute of Computer Technology
Pune, India
pakhadkikar@pict.edu

2nd Himanshu Shah
Dept. of Computer Engg.
Pune Institute of Computer Technology
Pune, India
hpshah294@gmail.com

3rd Shubham Chaudhary
Dept. of Computer Engg.
Pune Institute of Computer Technology
Pune, India
shubhamchaudhary632001@gmail.com

4th Shlok Kalekar
Dept. of Computer Engg.
Pune Institute of Computer Technology
Pune, India
shlok.kalekar@gmail.com

5th Jenish Chavda
Dept. of Computer Engg.
Pune Institute of Computer Technology
Pune, India
jenishchavda2001@gmail.com

***Abstract*—People who are specially impaired with regards to speaking and hearing find it difficult to communicate with people who are not impaired. The same is true vice-versa. For this very reason, Sign Language is necessary to be known and understood by both the parties. However, there are many people who are completely unaware of Sign Language and its importance. Hence, this project is aimed at bridging this communication gap between the two parties, and eventually, have an awareness on this importance of Sign Language. The paper presents an introductory analysis for the development of the Indian Sign Language (ISL) Recognition System. The primary focus of the paper is to discuss various steps involved in the formation, preprocessing, and augmentation of the dataset. The paper also briefly discusses the various steps performed after preprocessing to train and implement the model, with the focus on the systems based on Artificial Neural Networks.**

***Index Terms*—Indian Sign Language (ISL), Deep Learning, Object Detection, Computer Vision, Image Classification**

INTRODUCTION

According to the National Association of the Deaf (India), there are about 18 million Indians have some kind of hearing disability, without accounting those with the speaking disability. With such a large number of the differently-abled people who speak in the form of sign language, a concern arises based on the fact that the majority of the population is not able to communicate in such languages. For this reason, rather than teaching the Indian Sign Language to the entire population, which adds a number of logistic concerns, the better solution would be to automate the translation process with the help of a Sign Language Interpreter.

We have chosen Indian Sign Language as our target language for interpretation, as it has the largest population of users on the Indian subcontinent. However, there are only about 250 certified ISL interpreters (according to the-World.org), which is the cause of a multitude of communication issues in a largely populated country such as India. With such a lack of support for this language with a great number of users, we have chosen to specifically focus on Indian Sign Language.

Shagun Katoch et al. discussed that the Indian Sign Language (ISL) has certain complexity that many other sign languages do not have, which is regional dialects or versions of the same sign language. [1] Other than the fact that the language is double-handed language, the variations in the language from state-to-state makes it difficult to create a translator for the sign language. This led us to focus on the official ISL data provided by the Government of India, as one of our primary reference for developing the Indian Sign Language recognition system.

With languages such as American Sign Language (ASL) having been heavily researched upon, there are numerous existing approaches on automatic sign language translation through the use of Machine Learning. Pratik Likhar et al. discussed two of the main methods used at present for sign language recognition, which are based on using either sensors (such as gloves, etc.) or vision, both of which use Convolutional Neural Networks (CNNs) for training the datasets. [2]

The Sensor-Based Approach is a Depth-Sensing Approach, where a depth sensing camera or even gloves are used to create a model of the hand making specific hand signs. After training, this approach can be seen as highly accurate with 98.81 % classification accuracy as mentioned by Pratik Likhar et al. [2]. However, one of the issues with this approach is that, it requires a depth sensing camera or gloves at all times, which makes it less economical, affordable and accessible, especially in the remote parts of India. This remains a primary concern when it comes to the communication using sign language

throughout India.

The Vision-Based Approach, on the other hand, uses a standard camera for classification. The model is trained through semantic segmentation, which differentiates various signs by shape and colour. However, this can be quite concerning in case of the Indian population as it has a variety of skin colours. However, Ashish Sharma et al. discusses a solution for this issue, which is to train the model using the dataset with variety of skin colours. This method is slightly less accurate with 97% classification accuracy, but has the advantage of creating a more accessible means of translation throughout India as it requires only a standard RGB camera available on most smart phones and computers [3].

With the selected approach, we are creating an interpreter for ISL which focuses on the words, as that would allow for a much better expression of sign language in the form of a sentence. The aim would be to translate the sign language, and form sentences using the detected words, which can then be displayed or voiced, allowing for an easy understanding without the requirement of any prior knowledge of ISL. Thus, it allows the ease-of-communication between the general population and the people who primarily use ISL for communication.

Throughout the paper, we discuss the in-depth processes and steps involved in preparing and pre-processing the ISL dataset to train the model for sign detection and recognition, in a fast and effective way.

After performing the pre-processing steps, if the final dataset contains only uniform data without any variations, then new data needs to be generated for training. For this project, we have decided to apply data augmentation to increase diversity in the data so that it prevents overfitting of the model. The various data augmentation techniques are discussed in the paper further.

Related Work

Ease-of-access is a necessity these days with technology becoming simple to use day-by-day. This also holds true in case of the sign language interpreter, which are very less accessible, and are slowly being replaced with the automatic hand gestures detection and classification systems for the sign language. The detection and classification of hand gestures requires accuracy, so as to easily and accurately translate the sign language used by the differently-abled, to those who do not understand it.

However, to train a system for automatically detecting and classifying hand gestures, we need to apply the concepts of deep learning, and neural networks, due to the high complexity involved in the problem. The final aim is to train a highly accurate deep learning model, which also provides fast results, as the sign language translation happens at high speed.

To achive the required speed and accuracy, we need to train the model on a reliable dataset of adequate size, which covers most of the possible variations that might occur during the sign language translation.

The dataset needs to be consistent with the official Indian Sign Language, and then pre-processed, so that it can be converted into the form that is useful for training the model, while still retaining the useful information. The various steps involved in these processes are discussed in the paper.

Pavel Campr et al., in their, paper, presented the thorough process that they went through for the data collection and acquisition for Czech Sign Language, starting with the dataset specification and recording the video for the dataset. They also discuss in brief about the data pre-processing and feature extraction steps involved in the project [4].

Aleksejs Zorins et al. presented a brief review on the various data pre-processing techniques that can be used on the Latvian Sign Language dataset. These techniques include grayscale conversion, histogram equalization, fast fourier transformations, etc. [5].

After preparing the dataset, it is used for training the model using various approaches as discussed.

Rajarshi Bhadra et al. has proposed a deep convolutional neural network of multiple layers for training the system to detect sign language using images of hand gestures. In this proposition, 32 convolution filters are used which have 3x3 kernels. Also, the multiple layers of the CNN use LeakyReLU as the activation function. Then, after convolution, a 2x2 max pooling operation is done on the multiple layers of the CNN structure. The output layer uses SoftMax as the activation function. The proposition was evaluated on a database containing both static (54000 images and 36 classes) and dynamic (49613 images and 23 classes) hand gesture images [6].

Soma Shrenika and Myeni Madhu Bala proposed a system which utilises a camera for capturing a diverse variety of hand gestures, and afterwards the image is processed by multiple algorithms. At first, the image pre-processing is done, followed by the determining the edges using edge detection algorithm. Then, the sign is identified through a template-matching algorithm, which then displays the text, which can be converted to proper sentences [7].

K. Revnath and N.S.M Raja conducted a study on the comparison between various classifiers that can be used for hand gestures classification including SVM, KNN, Logistic Regression and Naive-Bayes classifiers. Out of all the mentioned classifiers, SVM provides the highest accuracy of about 90%, while Naive Bayes classifier provides only 55% accuracy. However, the proposed model only works for images and not for videos [8].

Ashish Sharma et al. performed an analytical comparison between three popular deep learning-based approaches for identifying ISL gestures. The prediction of the words is done when the input is given as a sequence. The accuracy for one-hand gestures is 98.52% and two-hand is 97%. However, it could not properly categorize the alphabet ‘J’, and it only deals with images [3].

T.F. Dima, and M.E. Ahmad conducted a research based on American sign language and proposed a model well suited for videos as it performs very fast end-to-end object detection with an average 0.98 F1-score. The proposed model is quite

lightweight to use and is suitable for mobile devices. However, it is quite complex to implement and require high computation power to train the model. [9]

Shagun Katoch et al. conducted a survey in which they presented a technique that uses a model named as the Bag of Visual Words (BOVW). This model helps in recognizing ISL alphabets (A-Z) and digits (0–9) from the live video. Then, it provides predicted labels as the output, which are used to form text. This text can be converted to speech as well. SURF (Speeded Up Robust Features) features are extracted from data to map the signs with corresponding labels. The model then uses Convolutional Neural Networks (CNN) and Support Vector Machine (SVM) for performing classification [1].

PROPOSED WORK

Sign Language requires a properly articulated data which is robust and efficient for developing a system with high accuracy. The basic architecture of the sign language recognition system used to detect and classify sign gestures is provided in Fig. 1.

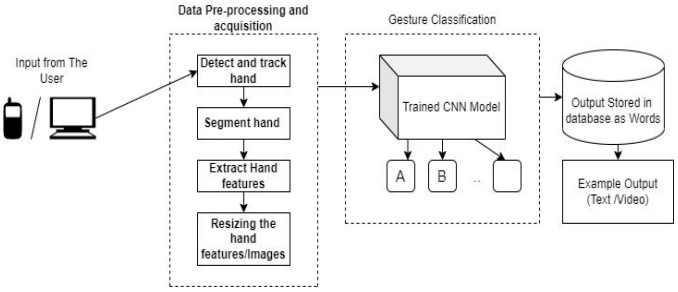


Fig. 1: Architecture Diagram

A. Dataset

The deep learning model performs detection and classification of signs in ISL based on the input video (dynamic), and not on static images. Thus, the dataset used in the project to train the model includes videos of various signs (gestures) representing words in ISL. These videos have been obtained from two sources:

- INCLUDE: A Large Scale Dataset for ISL, provided by Advaith Sridhar et al. [10]
- The Official Dictionary for ISL, provided by the Indian Sign Language Research and Training Centre (ISLRTC), Ministry of Social Justice & Empowerment, Government of India

The INCLUDE dataset consists of 263 classes (signs) with 4287 videos. These classes or terms (words) are spread across 15 categories out of which 13 represent nouns, and others are adjectives and pronouns. Most of the videos for each class are 2-4 seconds long, and there are about 16 videos per class.

The official dictionary for ISL comprises of 10000 everyday terms and phrases spread across various categories, and their representation in ISL using gestures in the form of videos. They are also categorized alphabetically. These videos are either 2-4 seconds long (in case of single words), or 10-12 seconds long (in case of phrases or idioms), but there is only one video per class.

For the sake of simplicity, we consider only the most common words including adjectives, pronouns, and nouns which limits the total terms to around 700-1000 terms.

The useful terms are extracted from the ISL dictionary to either increase the instance for classes (signs) provided in INCLUDE dataset, or to introduce new terms to cover a broader range of expressions through the trained model.

The videos from both the datasets have a frame rate of 25fps, and a resolution of 1920x1080. Thus, each video (of 2-4 seconds length) can be considered as a collection of 50-100 frames or images taken in a sequence.

Before training the model, we pre-process the data and then, apply data augmentation techniques (if required).

B. Data Pre-processing

The dataset comprises of videos, which are not suitable for being used as the training data for deep learning models. Thus, we need to pre-process the video data, and convert it into a format suitable for training the model effectively.

Before applying data augmentation to produce the sufficient and diverse data for training the model, we perform data pre-processing through the techniques as discussed below:

1) Data Acquisition
   The videos are located and downloaded from reliable sources (here, official ISL dictionary, and INCLUDE dataset), and stored separately.
2) Data Cleaning
   The videos are checked for their integrity and consistency, whether they are running correctly. The corrupted videos are discarded from the dataset.
3) Data Integration
   The data from both the sources (dataset) are integrated into a single coherent dataset on which the model is to be trained. Before proceeding further, data cleaning is performed again to ensure that there is no conflict between data from different sources.
4) Data Reduction
   The dataset is condensed by extracting only the useful classes/terms, which helps in training the deep learning model effectively, while maintaining the integrity of the original dataset.
5) Data Annotation/Labeling
   Videos are properly annotated or renamed so that they indicate the exact class or term that they represent. In other words, we properly assign labels to the videos in the dataset, so that the model can be trained effectively.
6) Gray-scaling
   For processing, images are represented as a matrix of same dimensions as the image resolution, wherein each cell stores the numeric value of intensity of that pixel (0 to 255). In case of a colour image (RGB), there are three such matrices corresponding to red, green, and blue

components, mixing which we get the specific colour shown by the pixel. Thus, in case of colour images, we need to train the model on each of the three matrices, which further increases the model complexity.

However, by applying gray-scaling, the image can be represented as a single matrix corresponding to only the intensity information of light. Thus, it reduces the size of training dataset, improves training time and reduces model complexity.

7) Resizing

Many of the deep learning model architectures contain neural networks with fully connected layers. These models require the input videos/images to be of a specific size to provide accurate results. Thus, we need to resize the input videos before applying data augmentation and training the model.

There are two ways to resize the videos/images:

i) *Re-scaling* : By re-scaling, we multiply the height and width of video by a scaling factor. If the scaling factor is not same for both dimensions, it leads to a change in the aspect ratio, and spatial extents of the pixels.

ii) *Cropping* : By cropping, we extract a sub-section of the image while preserving the spatial extend of each pixel.

C. Data Augmentation

Data Augmentation is a technique used to increase the size of the training dataset by generating modified data from the existing data. It introduces diversity in the dataset, which improves the accuracy of the model.

The official ISL dictionary comprise of only a single video per class, while the INCLUDE dataset comprise of videos with uniform properties. Thus, we use data augmentation techniques to introduce diversity in the dataset and prevent over-fitting of model.

To apply data augmentation on video data, we first split the it into frames, and apply the augmentation techniques on them. After generating the augmented images, we combine them to create the output video, which is included in the training dataset, as shown in Fig 2.

The various augmentation techniques that can be performed on the videos are as follows. For reference, we have applied these techniques on a frame extracted from the video on ISL gesture for word 'lecture'.

1) Flipping

In this augmentation, video frames are flipped along either a vertical axis (horizontal flip) or a horizontal axis (vertical flip). Due to horizontal flip, a left-handed signer appears as right-handed, and vice-versa.

The vertical flip is not applicable in our use case as the signer will never appear upside-down on the camera. Therefore, only the horizontal flip is used.

2) Cropping

In this augmentation, video frames are cropped at a random location. Also, to ensure that the augmented video is consistent, all the frames are applied the same cropping.

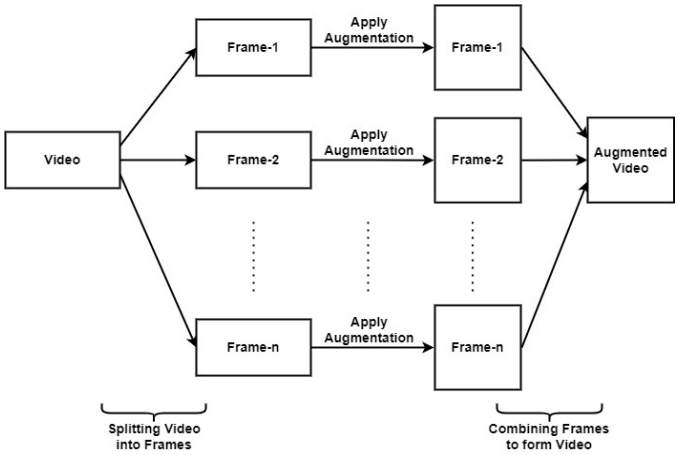


Fig. 2: Data Augmentation

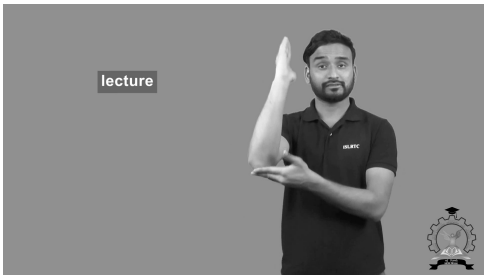
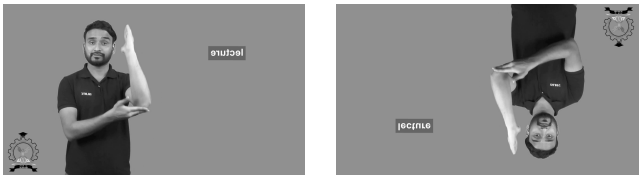


Fig. 3: Original Image/Frame

3) Down-sampling

In this augmentation, video frames are randomly selected and dropped from the video. Due to this, the resultant video appears to be in fast motion. This ensures that the deep learning model is trained for gestures made at faster speeds.



(a) Horizontal Flip (b) Vertical Flip

Fig. 4: Flipping

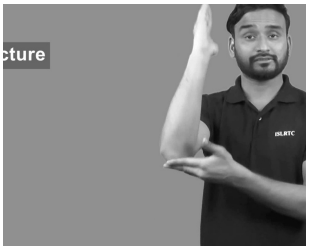


Fig. 5: Cropping (About the centre)

4) Up-sampling

In this augmentation, video frames are duplicated to increase their quantity in the video. Due to this, the augmented video appears to be in slow motion. This ensures that the deep learning model is trained for detecting and classifying gestures made at slower speeds.

5) Rotation

In this augmentation, all video frames are rotated clockwise or anti-clockwise by 90 degrees as shown in Fig.8. This augmentation technique is not applicable for our use case, as the signer will never appear in any other rotation. However, a small angle of rotation (within ± 20 degrees) might be applicable for our use case.

6) Colour Augmentation

In this augmentation, brightness, contrast, and/or saturation of the video frames are modified, due to which, the model is trained for real life scenarios of varying light conditions, etc.

7) Blurring (Gaussian Blur)

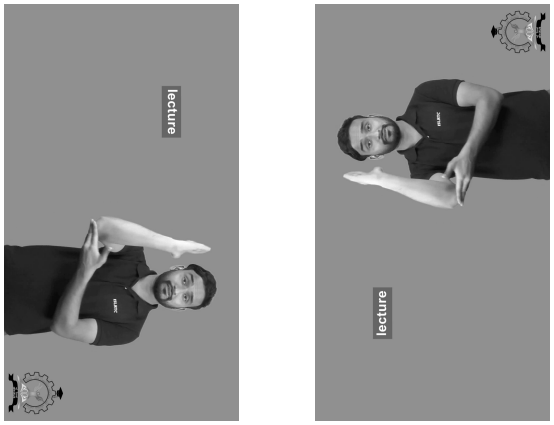
In this augmentation, video frames are blurred by using a Gaussian distribution and convolution kernel, due to which, the model is able to provide accurate results even if the input video is blurred due to external factors.

The degree of blurring depends upon the value of sigma. More the value of sigma, more blurry is the image. Thus, we need to define a threshold value of sigma, beyond which the image becomes unrecognizable.

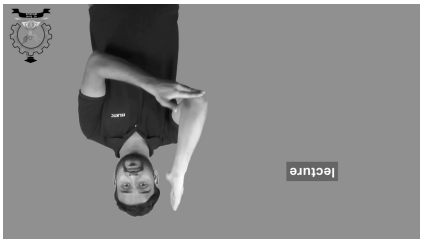
8) Adding Noise (Salt-Pepper Noise)

In this augmentation, white and black dots (noise) are added randomly to the video frames, due to which, the model is trained to deal with videos containing noise.

Thus, numerous variations are generated for a single video by applying the augmentation techniques mentioned above, and then included in the training dataset. All the techniques can also be applied sequentially with each other to generate more variety of data.

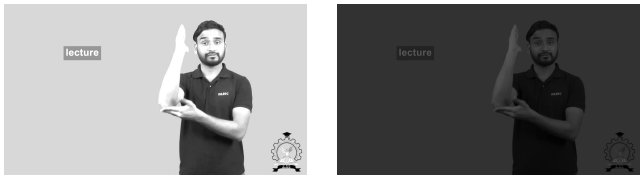


(a) 90^o (b) 270^o



(c) 180^o

Fig. 6: Rotation (Clockwise)



(a) Bright (b) Dark

Fig. 7: Colour Augmentation

D. Hand Gesture Recognition

This phase involves the recognition of hand gestures provided by the user. In this method, we are using the object detection techniques first to detect the hand and its gestures, and then we use either of the two approaches to create the proper input for the CNN model for classification: first one being Sensor-Based Approach which uses the concept of depth-sensing where we can use special depth sensing camera or even gloves to create a model of the hand making specific hand signs, and the second one being the Vision-Based Approach, which uses a normal standard camera for classification. The model is trained through semantic-segmentation, which has the potential to distinguish the different signs by shape and colour.

(a) $\sigma = 5$ (b) $\sigma = 10$

Fig. 8: Gaussian Blurring

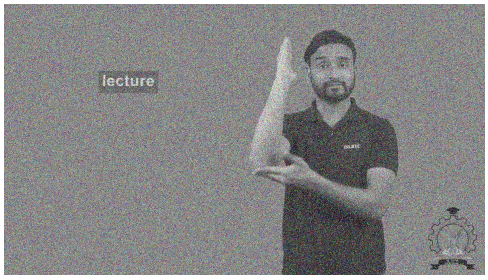


Fig. 9: Salt-Pepper Noise

E. CNN Based Model and Classification

Convolutional Neural Networks are the artificial neural networks which are primarily composed of multiple layers. The main objective and work of the CNN is to extract features from the image data and classify them with certain accuracy. CNN has a basic architecture as shown in Fig 2. Basically, in CNN, initially, we provide an input image to the convolutional layer. In that layer, the features are extracted from the input images using various feature detector filters. We get various features map from input images. Then, the pooling layers are added after the convolution layer, and are used to reduce the number of parameters, size of image and extract essential features. Many such layers are required for the classification to take place. The images are classified as per predefined classes where output unit represents a class of hand gestures. Classes will be the different alphabets, digits, or words.

F. Output

The classified output will be converted into a sequence of words, using which a proper sentence is formed. Then, a Text-to-Speech synthesizer (TTS) will be used for converting the sentence to speech. The TTS technology enables computers to speak to a person. The text is fed into the TTS system, which then uses a computer algorithm known as the TTS engine to analyse, pre-process, and synthesize the voice using some mathematical models. This voice will be played by the device which will help the physically challenged people to use modern services.

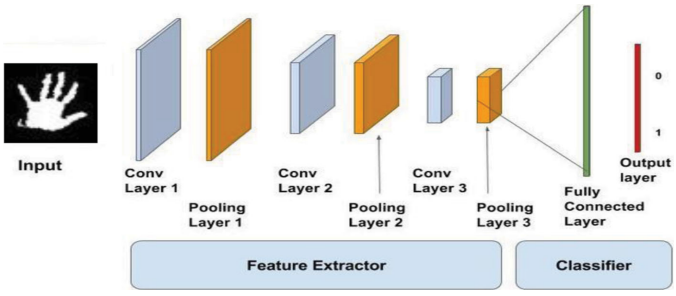


Fig. 10: CNN Basic Architecture [11]

CONCLUSION

Through the proposed process, we are going to develop a system which can detect the hand gestures of the Indian Sign Language (ISL). Before the training of model using the data, we will perform various data pre-processing and data augmentation steps. Then, after training the CNN-based model to detect and classify hand gestures using the video data from the dataset, we will use it to detect the hand gestures made by the user by taking video of sign gestures as input. Then, we will use the text output and convert it into speech using Text-to-Speech Synthesizer (TTS). The related work provides us the with insights of all the previous works done, but they have certain gaps which are going to be addressed in our proposed project. The future challenge is to create a robust system that can be process the sentences in Sign Language.

REFERENCES

[1] Shagun Katoch, Varsha Singh, and Uma Shanker Tiwary. Indian Sign Language recognition system using SURF with SVM and CNN. *Array*, 14:100141, 2022.

[2] Pratik Likhar, Neel Kamal Bhagat, and Rathna G N. Deep Learning Methods for Indian Sign Language Recognition. In *2020 IEEE 10th International Conference on Consumer Electronics (ICCE-Berlin)*, pages 1–6, 2020.

[3] Ashish Sharma, Nikita Sharma, Yatharth Saxena, Anuraj Singh, and Debanjan Sadhya. Benchmarking deep neural network approaches for Indian Sign Language recognition. *Neural Computing and Applications*, 33(12):6685–6696, Jun 2021.

[4] Pavel Campr, Marek Hruz, and Jana Trojanova. Collection and Preprocessing of Czech Sign Language Corpus for Sign Language Recognition. 01 2008.

[5] Aleksejs Zorins and Peter Grabusts. Review of Data Preprocessing Methods for Sign Language Recognition Systems based on Artificial Neural Networks. *Information Technology and Management Science*, 19, 12 2016.

[6] Rajarshi Bhadra and Subhajit Kar. Sign Language Detection from Hand Gesture Images using Deep Multi-layered Convolution Neural Network. In *2021 IEEE Second International Conference on Control, Measurement and Instrumentation (CMI)*, pages 196–200, 2021.

[7] Soma Shrenika and Myneni Madhu Bala. Sign Language Recognition Using Template Matching Technique. In *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*, pages 1–5, 2020.

[8] K. Revanth and N. Sri Madhava Raja. Comprehensive svm based indian sign language recognition. In *2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN)*, pages 1–4, 2019.

[9] Tasnim Ferdous Dima and Md. Eleas Ahmed. Using YOLOv5 Algorithm to Detect and Recognize American Sign Language. In *2021 International Conference on Information Technology (ICIT)*, pages 603–607, 2021.

[10] Advaith Sridhar, Rohith Gandhi Ganesan, Pratyush Kumar, and Mitesh Khapra. INCLUDE: A Large Scale Dataset for Indian Sign Language Recognition. In *Proceedings of the 28th ACM International Conference on Multimedia*, MM '20, page 1366–1375, New York, NY, USA, 2020. Association for Computing Machinery.

[11] Jayesh Gangrade and Jyoti Bharti. Vision-based Hand Gesture Recognition for Indian Sign Language Using Convolution Neural Network. *IETE Journal of Research*, pages 1–10, 11 2020.

A Comparative Study of Driver Drowsiness Detection Techniques

Dr. Prajakta Khadkikar
Computer Engineering
Pune Institute of Computer Technology
Pune, India
pakhadkikar@pict.edu

Miss. Rutuja Banginwar
Computer Engineering
Pune Institute of Computer Technology
Pune, India
rutujabanginwar01@gmail.com

Mr. Imon Mandal
Computer Engineering
Pune Institute of Computer Technology
Pune, India
imonashokmandal@gmail.com

Miss. Urmila Matade
Computer Engineering
Pune Institute of Computer Technology
Pune, India
matadeurmila1712@gmail.com

Mr. Mayur Mote
Computer Engineering
Pune Institute of Computer Technology
Pune, India
mayurmote8055@gmail.com

***Abstract*—Over the past ten years, the number of motor vehicles has steadily grown in emerging nations. The reports of traffic accidents note that risky driving behaviors, such as drunken driving or fatigued driving, are responsible for most accidents. According to studies, sleepy driving is a contributing factor in 20% of all accidents. Drowsiness is a condition when the level of consciousness is lowered as a result of fatigue or lack of sleep, and it can make a driver fall asleep silently. Drowsy driving causes the driver to lose control of the vehicle, causing it to drift off the road, collide with an obstacle, or overturn.In order to extract and synthesise the methods and features that have been utilized in drowsiness detection studies, we conducted a Systematic Literature Review (SLR) for this work. We found 100 relevant studies using our search parameters, of which we have chosen 30 for additional research. We thoroughly examined these chosen works, evaluated the methods and features applied, and identified the research gap as well. According to our analysis, the most used features are facial expressions like yawning, closing of the eyes, and head motions. These insights are based on an analysis of 30 papers and show that the Eye Aspect Ratio, Haar Cascade, Support Vector Machine (SVM), and dlib library are frequently utilised techniques.**

***Index Terms*—driver drowsiness, visual features, machine learning , fatigue detection, drowsiness detection.**

I. Introduction

One of the primary causes of catastrophic injuries, fatalities, and monetary losses on the road is drowsiness. Serious traffic accidents are triggered by a lack of alertness brought on by the unconscious transition from awake to sleep.

When developing methods to track driver inattention, researchers concentrated on two areas of study: drowsiness and distraction. Visual feature-based and non-visual feature-based systems are typically used in systems developed for drowsiness analysis and detection. Computer vision methods are used in techniques that employ visual elements to identify sleepiness. The extraction of facial characteristics like the face, eyes, and lips is the main goal of the utilisation of visual features. Analyzing the state of the mouth and eyes can offer observable hints for the detecting procedure. The four basic types of techniques that make use of visual features are: facial expression analysis, blink analysis, mouth and yawning analysis, and eye state analysis. The two main processes for visual feature recognition and processing are image processing and machine learning approaches. The two types of non-visual feature-based techniques—vehicle parameter analysis and driver physiological analysis—are often obtrusive. For instance, multiple sensors must be linked to the driver when they are driving in EEG technology. The three parts or modules that make up the basic sleepiness detection system are the retrieval system, processing system, and alert system. Here, the retrieval system records a video of the driver's front face and sends it to the processing block for online processing to identify sleepiness. If drowsiness is detected, the alert system sends a warning or alert to the driver.

This document's remaining sections are arranged as follows: The background is described in Section 2. The procedure is covered in Section 3. The various drowsiness detection techniques are covered in Section 4. Section 5 gives the SLR findings, and Section 6 wraps up the document.

II. Related Work

While searching for the research paper, we excluded the research works that were very old and only considered the latest research. Shruti Mohanty and et.al published a paper on drowsiness detection using opencv, python, dlib and calculated the Euclidean aspect ratio of eyes and mouth to for drowsiness detection [1]. Using the six landmarks surrounding the eye ROI, Jagendra Singh's article on Learning-based Driver Drowsiness Detection Model was proposed in 2020. An warning is given to the driver informing them that they may have tired eyes if the EAR drops below the threshold for at least 20 frames and the threshold is set at 0.3. An alert is produced to the driver, urging them to pay attention to the

road, if the landmarks around the mouth ROI and the eyes ROI cannot be determined for at least 15 frames [2].

Amin Azizi Suhaiman and et.al proposed a paper in 2020 titled "Development of an Intelligent Drowsiness Detection System for Drivers Using Image Processing Technique," in which they used a computerised camera to automatically track and process the driver's eye using Python, dlib, and OpenCV. Image processing depends on light intensity, which impacts the results of detection [3]. In 2011, R. N. Khushaba and et.al proposed a work on the categorization of driver drowsiness using fuzzy wavelet-packet-based feature extraction methods. As physiological approaches frequently call for electrodes to be placed on the body of the driver, which is inherently invasive and can be burdensome for the driver[29], they have developed a fuzzy mutual information based wavelet packet transform model to estimate the level of drowsiness from a set of electroencephalogram, electrooculogram, and electrocardiogram signals.

By using the driver's eye movements and driving performance data gathered in a simulator environment known as the in-vehicle information system, the authors of the paper titled "Real-time detection of driver cognitive distraction using support vector machines" have used a real-time approach for detecting distraction. After that, the data were utilised to train and test support vector machine (SVM) and logistic regression models for identifying driver distraction. Thus, the type of vehicle, the driver's background, and the state of the road all have a significant role. If the driver falls asleep while driving on a straight road, such systems may fail because the vehicle fails to deliver crucial information [4]. The usage of a smart watch and a headband with sensors to detect drowsiness was proposed in a study by Li, Gang, and et.al in 2015, titled "Smartwatch Based Wearable EEG System for Driver Drowsiness Detection." Instead of discrete class labels designed to identify tiredness as early as feasible, the amount of driver drowsiness could be translated to any probability value between 0 and 1. The driver's decision to wear a different watch or forego a headband places restrictions on this [5].

In 2009, Li, Lingling, and et.al proposed a paper on Yawning detection for monitoring driver fatigue based on two cameras. In it, they monitor the driver's yawning patterns using two separate cameras to collect data from the upper part of the body in order to track the driver's mouth. However, this has the drawback of being more dependent on hardware [6].

A computer vision-based method for real-time detection of sleep onset in fatigued drivers was proposed by A. B. Albu and et.al in a study published in 2008. We now know from reading this work that real-time performance was achieved by concentrating on one specific visual cue, or eye condition, and utilising a specially created template-matching algorithm for online eye condition recognition. A computer vision-based sleep monitoring technique has also been developed, and it will be implemented using DSPs or FPGAs that have been custom-designed and integrated into the vehicle [7].

The 2010 paper "An EEG-based approach for diagnosing drowsy driving circumstances" by M. Liand and et.al made a recommendation for a vehicle driving simulation system that was employed in the study. Drivers who were awake and sleepy provided EEG data. The interference of EMG and EOG in the EEG was eliminated using FastICA arithmetic. The EEG exceeding equation's tiredness index F can provide aspects of the management of weariness. It can be established to analyse and evaluate the degree of weariness since the driving fatigue F value was much higher than that for normal driving [8].

According to research presented in the 2007 publication "Detecting Driver Yawning in Successive Images" by Lu Yufeng and Wang Zengcai, we may identify driver yawning by measuring the vertical distance between the centre of the nostrils and the chin. Because the approach was not reliant on hue or intensity, which are vulnerable to variations in illumination, they claimed that it was robust to these variations [9]. In a paper published in 2011, S. Abtahi and coworkers hypothesised that it could be able to select a challenging algorithm that would bring the system under "Yawn Detection-Based Driver Drowsiness Monitoring" one step closer to being really implemented [10].

"Real-time categorization of driver's gaze zone using the deep learning technique" a work by Choi and et.al, was published in 2016. Deep learning and face expression recognition were employed. Over 95% of the time, the suggested system had a mean detection accuracy. This method's drawback is that a significant amount of data is needed to train the neural network to function with high levels of accuracy [11].

The hardware-based fatigue detection system presented by Fouzia and et.al is based on visual parameters. The Raspberry Pi and Python OpenCV library are used by this system to identify tiredness based on the number of eye blinks. The suggested model continuously tracks the driver's eye movement and alerts them when they become sleepy by activating the vibrator. If the driver's eyes are identified to have been closed for an extended period of time, a vibrator signal is generated to alert them [12]. Drowsiness is one of the leading causes of traffic accidents. Ashish and Rusha suggested a driver fatigue detection system based on yawning and mouth analysis to get around this problem. To identify drowsiness using adaptive thresholding based on these values, facial landmarks on the detected face are first recognised. Based on these values the mouth aspect ratio, nose length ratio and eye aspect ratio are then calculated [13]. Sathasivam and et.al proposed the fatigue detection system using the eye aspect ratio technique. The primary goal of the suggested method is to identify driver drowsiness and alert them by sounding an alarm. In order to identify drowsiness, this system uses a camera situated in front of the driver to record real-time video, which is then utilised to analyse video frames. [14].

Javed Ahmed and et.al published a paper in 2015 called "Eye Behaviour based Drowsiness Detection System" in the IEEE. In this paper computer vision-based approach has been used to create a driver drowsiness detection system. In order to specifically identify driver drowsiness, a system that uses the small camera and analyzes the driver's eyes while focusing directly on his or her face has been developed. When a driver's

weariness is apparent, a warning sign is displayed to alert other road users. The procedure of locating the driver's eyes and determining whether they are open or closed is demonstrated in this paper. The system is able to find the margins of a face and determine where a person's eyes may be by using binary image data that was acquired. The proposed method ensures that the driver is dozing off if their eyelids are closed for five consecutive frames, at which point a warning signal is sent. The framework also functions in conditions with reasonable lighting and can detect when the eyes are hidden from view [15].

In 2018, Ratnarup Dey and Joy Paulose published a revised algorithm for nonintrusive driving fatigue detection in the International Journal of Applied Engineering Research. The paper discusses a car-integrated system that monitors the behaviour of the driver based on a variety of stimuli. For the purpose of detecting head movement, steering gripping, and drunk driving, sensor characteristics are employed. As a part of this work, a system that can detect driver drowsiness by evaluating any one, two, or all three of the following parameters head movement, alcohol influence, and steering grasping is proposed. When one, two, or all of these circumstances exist, it is assumed that the driver is drowsy [16].

III. Methodology

A. Review protocol

The initial step is to define the research questions. Relevant studies are chosen from the databases once the research questions are ready. The databases used in this investigation were Google Scholar, Springer Link and Science Direct. After selecting the related research, it was filtered and evaluated using a set of acceptance and performance criteria. From the chosen studies, all relevant data are retrieved, and ultimately, the extracted data are synthesized to address the research objectives.

B. Research questions

The purpose of this literature review is to find out already published algorithms and techniques for drowsiness detection and evaluate their performance. To prepare this literature review following research questions have been defined.

RQ1:- Which technique has been used for drowsiness detection?

RQ2:- What algorithms have been used to identify drowsiness?

RQ3:- What are the different visual features that the system takes into account?

RQ4:- What is the research gap in the proposed algorithm?

C. Search strategy

An automated search handles the fundamental searching. The search's initial inputs were "machine learning", "drowsiness detection system" AND "driver". Three databases were searched in this operation. To find a wide range of papers, the search terms "driver" AND "drowsiness detection system" were used.

Following is a detailed explanation of the search terms for each database:

ResearchGate: The search term is ["machine learning" AND "driver drowsiness detection"] and ["driver drowsiness detection"]

Springer Link: The search term is ["drowsiness detection system" AND "yawning"] and ["drowsiness recognition" AND "facial features"]

Google Scholar: The search term is ["driver drowsiness detection" AND "eyes aspect ratio"] and ["image processing" AND "drowsiness detection" AND "machine learning"]

TABLE I
Search Stratergy

Database	Initially retrieved papers	Papers after exclusion criteria	Percentage of Papers (%)
ResearchGate	7	1	14
Springer	13	2	15
Google Scholar	80	27	34

IV. Drowsiness Detection Techniques

The Driver Drowsiness detection system continuously tracks the driver's physical behaviour, driving habits, and environmental conditions, depending on the technique being used. Methods for detecting sleepiness are often categorised into three groups.

1) Methods based on behavioural parameter data
2) Methods that are based on vehicle parameters
3) Methods using physiological parameters

A. *Behavioural Parameter-Based Techniques*

Computer vision algorithms and cameras were combined with behavioral-based strategies to extract behavioural information. The non-intrusive methods of sleepiness detection are behavioural parameters. The behavioural indicators used by these strategies to gauge driver weariness include facial expressions, yawning, head position ratio, eye closure ratio and eye blinking. Environmental elements, such as lighting, brightness, and road conditions, have an impact on the measurement's reliability and accuracy, which are issues with behavioural measures.

*1) **Eye Blink Detection Method**:* The hardware-based fatigue detection system presented by Fouzia and et.al is based on visual parameters. The Raspberry Pi and Python OpenCV library are used by this system to identify tiredness based on the number of eye blinks. The suggested model continuously tracks the driver's eye movement and alerts them when they become sleepy by activating the vibrator. If the driver's eyes are identified to have been closed for an extended period of time, a vibrator signal is generated to alert them. If the driver's eyes are identified to have been closed for an extended period of time, a vibrator signal is generated to notify them. The system performs well in terms of accurate sleepiness detection results.

*2) **Eye and Yawning Analysis**:* A real-time drowsiness detection system utilizing Dlib was proposed by Mohanty, Hegde, and Manikandan [1]. This paper employed a drowsiness detection system using Python and the Dlib model. The face detector in Dlib is used to map the positions of the facial landmarks in the video frame. it is also employed for tracking the eyes and mouth movement using aspect ratio technique to detect the driver drowsiness. For the eyes and mouth, respectively, predefined threshold values of 0.15 and 0.83 are used to compare the estimated aspect ratios with each. The left and right eyes' average aspect ratio must be above the threshold in order to be considered drowsy. The driver is alerted if indicators of fatigue are noticed for an extended period of time. To assess the effectiveness of the proposed system, real-time video and videos from a general public dataset are tested. The greatest recognition accuracy for video datasets for the proposed system was 96.71%.

*3) **Nose, Eye and Yawning Analysis Method**:* Ashish and Rusha suggested a driver fatigue detection system based on yawning and mouth analysis to get around this problem. To identify drowsiness using adaptive thresholding based on these values, facial landmarks on the detected face are first recognised. Based on these values the mouth aspect ratio, nose length ratio and eye aspect ratio are then calculated. This method uses a linear support vector machine and a histogram of oriented gradients (HOG) to identify the face in the frames (SVM). The landmark points on the face are estimated using image normalisation and a group of regression trees. Then, for classification, Support Vector Machine (SVM) with linear kernel were applied and Fisher's linear discriminant analysis (FLDA) were used. The INVEDRIFAC dataset has been used to test the newly built algorithm. SVM and FLDA have been found to perform better than Bayesian classifiers. Both FLDA and SVM have a specificity of 1, while their respective sensitivity values are 0.896 and 0.956.

*4) **Mouth and Yawning Analysis**:* Saradadevi and Bajaj [17] suggested a mouth and yawning analysis-based driver fatigue detection system. First, the driver's face from the input frames is recognised by the Viola-Jones classifier. Following that, SVM is used to train the mouth and yawning image datasets. Finally, yawning and sleepiness are detected by classifying the mouth regions using SVM. A dataset of more than 100 typical films and 20 yawning photos is chosen for the experiment. The results demonstrate that the suggested system outperforms the system founded on geometric features.

B. Vehicular Parameter Based Techniques

Vehicle attributes like frequent lane changes, variable vehicle speeds, steering wheel tilt, steering wheel grip force, etc. are used in vehicle parameter-based strategies to identify driver weariness. These measures require sensors on automotive components such as the steering wheel, gas pedal and brake pedal other. These sensors produce signals that are used for analysis driver drowsiness. The main objective of these techniques is analyze driving behavior and record a decline in driving performance due to fatigue. However, evaluating driver weariness based on vehicle movement has its limitations because measurement results are often influenced by outside variables like the geometry of the road and the weather.

*1) **Real Time Lane Detection System**:* In the modern era, traffic accidents are all too common, endangering both the lives of those on the road and their property. Road accidents can occur for a variety of causes, including: rash driving, inexperience, disobeying signs, jumping signals, etc. The Drivers' Drowsiness Detection system was suggested by Katyal et al. [18] as a solution to the problems. The system operates in two stages: first, it uses the hough transform to recognise lanes. Secondly, it looks for signs of tiredness in the drivers' eyes. The viola jones approach is used to detect faces first, followed by picture segmentation, otsu thresholding, and clever edge detection for eye detection. The data are used with the circle detection through transform method to find eyeballs and gauge level of exhaustion. Low lightning conditions won't affect its effectiveness. The results indicate that the suggested strategy is beneficial for drivers who take long journeys, drive at night, or drive after drinking.

*2) **Wheel Angle of Steering For Real-Time Driving Conditions For Drowsiness Detection System**:* To track the degree of driver weariness under real-world driving conditions, Li et al. [19] proposed using an online Drowsiness Detection System that used steering wheel angles (SWA). SWA data is collected by sensors that are located on the steering wheel. On the real-time steering wheel angle time series, the system first extracts the approximate entropy characteristic (ApEn) from the fixed sliding windows. The system then uses a variation of the piecewise adaptive linear technique to linearize the ApEn properties. The method then calculates the linear properties of the strain distance between a series of sample data. Finally, the system uses the deformation distance in accordance with the created decision classifier to determine the drivers' fatigue level. The empirical analysis uses data collected over 14.68 hours of actual driving time, assessed at two levels of fatigue: sleepiness and alertness. The results indicate that the proposed system is beneficial for the prevention of traffic accidents caused by driver drowsiness and is able to operate online with an accuracy of 78.01

C. Physiological Factors Based Techniques

Based on the drivers' physiological data, such as heartbeat, respiration, respiratory rate, pulse rate, body temperature, the physiological factors based algorithms identify drowsiness. These biological indicators are more dependable and accurate at detecting tiredness because they focus on the physical state of the driver. Systems for detecting drowsiness based on physiological characteristics notice these modifications and warn the driver when he is getting close to falling asleep. Electrodes must be positioned on the drivers' bodies because these measures are intrusive. This technique can occasionally irritate drivers, making it challenging to use.

*1) **EEG Based Driver Fatique Detection**:* M. Li, Z. Cheng, and J.F. Fang proposed the paper "An EEG-based approach for diagnosing drowsy driving circumstances" [12].

In this study, the driving experience was simulated using a vehicle driving simulation system. Both while the drivers were awake and when they were asleep, EEG data was recorded. Using FastICA arithmetic, the interference between EMG and EOG and EEG was eliminated. Following the signals' FFT conversion from the time domain to the frequency domain, the power spectral density for the EEG data was determined. The following formula is used to construct the EEG-derived fatigue index F:

$$F = \frac{E_\delta + E_\theta}{E_\alpha + E_\beta}$$

The fatigue index F identifies the signs of drowsy driving. Driving exhaustion has an F-value that is much higher than that of normal driving, making it a useful tool for assessing the level of fatigue.

*2) **Driver Drowsiness Detection System Using Wearable**:* Li, Gang, Lee, Boon-Leng, and Chung, Wan-Young [5] published a paper in 2015 proposing a wearable EEG system for drivers that is based on a smartwatch. To identify fatigue, the technology employs a headband and a smart watch with sensors. With the intention of replacing discrete labels with a value between 0 and 1, the Posterior Probabilistic Model based on Support Vector Machine(SVMPPM) used for Driver Drowsiness Detection is demonstrated in this work. To analyse the given model in real-time, a Commercial smartwatch and a bluetooth-enabled EEG headgear were employed. In a one-hour repetitive driving simulation experiment, this model was tested on 20 participants after being developed on 15 subjects and tested on 5 individuals. The proposed system attained an accuracy of up to 91.25%, which was the highest.

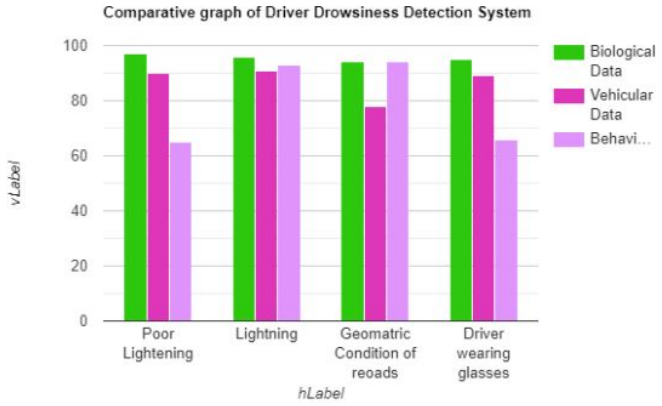


Fig. 1. Comparative graph Of Driver Drowsiness Detection System

V. Conclusion

This study demonstrated that the chosen publications employ various features based on the breadth of the research and the accessibility of the data. While its aspects differ, each study looks into how to identify tiredness using machine learning. The biological, mechanical, and behavioural factors

TABLE II
Driver Drowsiness Detection Techniques

I. Behavioural Parameter-Based Techniques			
Ref	**Features**	**Approach/Techniques**	**Efficiency**
[1]	Eye Blinking	Raspberry Pi Python OpenCV library	NIL
[3]	Eye State and Yawning	Dlib library and Python	96.71%
[2]	Face components(nose length, eye closing, yawning)	Aspect Ratio method and SVM	89.6%
[4]	Mouth and Yawning Analysis	SVM	NIL
II. Vehicular Parameter-Based Techniques			
Ref	**Features**	**Approach/Techniques**	**Efficiency**
[31]	Lane Detection	Extracting Approximate Entropy from Steering Wheel Angle	78.01%
[32]	Steering Wheel Angle	Hough Transform, voila jones algorithm	NIL
III. Physiological Parameter-Based Techniques			
Ref	**Features**	**Approach/Techniques**	**Efficiency**
[12]	EEG (electroencephalogram)	Fatigue Index (F)	NIL
[27]	Wearable Devices	SVM and Posterior Probabilistic model	91.25%

of studies also vary. The availability of the data collection and the research's objective both influence the choice of functions. Studies show that the best performance for sleepiness detection is not always provided by multiple feature models. Models with various features should be evaluated in order to determine which one performs the best. Numerous algorithms have been used in various research. The findings indicate that it is impossible to determine which machine learning model is the best with certainty, but they do demonstrate that some models are employed more frequently than others. The methods based on CNN, SVM and Aspect Ratio are the most popular models. To determine which machine learning model had the best detection, the majority of studies tested many models.

References

[1] S. Mohanty, S. V. Hegde, S. Prasad, and J. Manikandan, "Design of real-time drowsiness detection system using dlib," *2019 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)*, pp. 1–4, 2019.

[2] J. Singh, "Learning based driver drowsiness detection model," *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*, pp. 698–701, 2020.

[3] A. A. Suhaiman, Z. May, and N. A. A.Rahman, "Development of an intelligent drowsiness detection system for drivers using image processing technique," *2020 IEEE Student Conference on Research and Development (SCOReD)*, pp. 233–236, 2020.

[4] Y. Liang, M. L. Reyes, and J. D. Lee, "Real-time detection of driver cognitive distraction using support vector machines," *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 2, pp. 340–350, 2018.

[5] G. Li, B. L. Lee, and W. Y. Chung, "Smartwatch-based wearable eeg system for driver drowsiness detection," *IEEE Sensors Journal*, vol. 15, no. 12, pp. 7169–7180, 2015.

[6] L. Li, Y. Chen, and Z. Li, "Yawning detection for monitoring driver fatigue based on two cameras," *2009 12th International IEEE Conference on Intelligent Transportation Systems*, pp. 1–6, 2009.

[7] A. B. Albu, B. Widsten, T. Wang, J. Lan, and J. Mah, "A computer vision-based system for real-time detection of sleep onset in fatigued drivers," *2008 IEEE Intelligent Vehicles Symposium*, pp. 25–30, 2008.

[8] M. a. Li, C. Zhang, and J. F. Yang, "An eeg-based method for detecting drowsy driving state," *2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 2164–2167, 2010.

[9] L. Yufeng and W. Zengcai, "Detecting driver yawning in successive images," *2007 1st International Conference on Bioinformatics and Biomedical Engineering*, 2007.

[10] S. Abtahi, b. Hariri, and S. Shirmohammadi, "Driver drowsiness monitor-ing based on yawning detection," *2011 IEEE International Instrumentation and Measurement Technology Conference*, 2011.

[11] I.-H. Choi, S. K. Hong, and Y.-G. Kim, "Real-time categorization of driver's gaze zone using the deep learning techniques," *2016 International Conference on Big Data and Smart Computing (BigComp)*, pp. 143–148, 2016.

[12] Fouzia, R. Roopalakshmi, J. A. Rathod, A. S. Shetty, and K. Supriya, "Driver drowsiness detection system based on visual features," *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, pp. 1344–1347, 2018.

[13] A. Kumar and R. Patra, "Driver drowsiness monitoring system using visual behaviour and machine learning," *2018 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE)*, pp. 339–344, 2018.

[14] S. Sathasivam, A. K. Mahamad, S. Saon, A. Sidek, M. M. Som, and H. A. Ameen, "Drowsiness detection system using eye aspect ratio technique," *2020 IEEE Student Conference on Research and Development (SCOReD)*, pp. 448–452, 2020.

[15] J. Ahmed, J. P. Li, S. A. Khan, and R. A. Shaikh, "Eye behaviour based drowsiness detection system," *2015 12th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, pp. 268–272, 2015.

[16] R. Dey and J. Paulose, "Driver drowsiness detection system based on visual features," *International Journal of Applied Engineering Research*, vol. 13, no. 2, pp. 1219–1226, 2018.

[17] M. Saradadevi and D. P. Bajaj, "Driver fatigue detection using mouth and yawning analysis," *IJCSNS International Journal of Computer Science and Network Security*, vol. 8, no. 6, 2008.

[18] Y. Katyal, S. V. Alur, and S. Dwivedi, "Safe driving by detecting lane discipline and driver drowsiness," *2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies*, pp. 1008–1012, 2014.

[19] L. Z, L. SE, L. R, C. B, and S. J., "Online detection of driver fatigue using steering wheel angles for real driving conditions," *Sensors (Basel)*, vol. 17, no. 3, 2017.

[20] G. Uzair, C. Pravin, G. Siddharth, G. Rohit, and F. Kausar, "Drowsiness detection and monitoring system," *ITM Web of Conferences*, 2020.

[21] W. Deng and R. Wu, "Real-time driver-drowsiness detection system using facial features," *IEEE Access*, vol. 7, pp. 118 727–118 738, 2019.

[22] U. Budak, V. Bajaj, Y. Akbulut, O. Atila, and A. Sengur, "An effective hybrid model for eeg-based drowsiness detection," *IEEE Sensors Journal*, vol. 19, no. 17, pp. 7624–7631, 2019.

[23] J. J. Yan, H. H. Kuo, Y. F. Lin, and T. L. Liao, "Real-time driver drowsiness detection system based on perclos and grayscale image processing," *2016 International Symposium on Computer, Consumer and Control (IS3C)*, pp. 243–246, 2016.

[24] I.belakhdar, W. Kaaniche, R. Djmel, and B. Ouni, "A comparison between ann and svm classifier for drowsiness detection based on single eeg channel," *2016 2nd International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, pp. 443–446, 2016.

[25] F. You, X. Li, Y. Gong, H. Wang, and H. Li, "A real-time driving drowsiness detection algorithm with individual differences consideration," *IEEE Access*, vol. 7, pp. 179 396–179 408, 2019.

[26] B. Alshaqaqi, A. S. Baquhaizel, M. E. A. Ouis, M. Boumehed, A. Ouamri, and M. Keche, "Driver drowsiness detection system," *2013 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA)*, pp. 151–155, 2013.

[27] A. Sinha, R. P. Aneesh, and S. K. Gopal, "Drowsiness detection system using deep learning," *2021 Seventh International conference on Bio Signals, Images, and Instrumentation (ICBSII)*, pp. 1–6, 2021.

[28] R. Bhargava, R. Panchal, P. Verma, and S. Sharma, "Drowsiness detection while driving using eye tracking," *International Research Journal of Engineering and Technology (IRJET)*, vol. 7, 2020.

[29] N. Radha, E. M. Malathy, R. Swathika, R. B. Jananie, and A. A. Silviya, "Drowsiness detection system using visual articulators," *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, pp. 1086–1091, 2021.

[30] M. A. Assari and M. Rahmati, "Driver drowsiness detection using face expression recognition," *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pp. 337–341, 2011.

[31] S. Kaplan, M. A. Guvensan, A. G. Yavuz, and Y. Karalurt, "Driver behavior analysis for safe driving: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 3017–3032, 2015.

[32] R. N. Khushaba, S. Kodagoda, S. Lal, and G. Dissanayake, "Driver drowsiness classification using fuzzy wavelet-packet-based feature-extraction algorithm," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 1, pp. 121–131, 2011.

Building an

Effective Dev Portfolio

Josh Comeau

Building an Effective Dev Portfolio

Written by Joshua Comeau

This book is self-published.

INTRODUCTION

According to Career Karma[†], about 34,000 people graduated from software-development bootcamp programs in 2019, in the US alone. This is in addition to all of the folks who self-taught through online programs like FreeCodeCamp, as well as folks graduating from college with Computer Science / Software Engineering degrees.

That's a lot of people competing for junior software development jobs!

In order to land an offer, you'll need to stand out from the crowd. A college degree from a prestigious university is one way to do this, but not everyone is privileged enough to be able to attend one. Prior work experience is another way, though there's a catch-22 here: if you need work experience to get a job, how can you possibly get work experience?!

Without a degree or previous work experience, **your portfolio** is the greatest asset you have. A couple of well-executed projects sends *such* a strong signal to potential employers. Portfolio projects show that you have the skills required to do the job.

We need to do some work to showcase those projects, though. Even the most amazing portfolio project might be overlooked if an employer only sees a couple screenshots and a Github link. We need to *guide* potential employers through our work, making sure to highlight the most interesting, impressive, salient parts. The good stuff. Stuff that isn't always obvious at first glance.

This book is an instruction manual for creating a portfolio site that stands out to prospective employers, in order to give you the best chance of landing an offer. It focuses on the practical tips and tricks I've learned over the years to help capture the attention of the folks who review applications.

[†] Source: https://careerkarma.com/blog/bootcamp-market-report-2020/

MOTIVATION

A few months ago, I offered to do volunteer portfolio reviews on Twitter, and got over 200 responses (https://twitter.com/JoshWComeau/status/1227213590274486275).

After reviewing a whole bunch of these portfolios, I realized I was giving the same feedback in 75%+ of cases. Most junior developer portfolios—even ones that were very well-built—were missing a lot of potential opportunities to stand out to employers.

I know about what employers are looking for because I've been involved in hiring for multiple organizations, including Khan Academy, Unsplash, Breather, and more. I've worked with HR departments to decide whether to interview someone or not. I've been on both sides of the table.

I've been using this information to help aspiring developers for years. I work with Concordia University (through a partnership with Journey Education) to help coach recent graduates and ensure that their applications are as polished and effective as possible.

For a while now, I've wanted to share this information more broadly. I had initially planned to write a blog post, but I quickly realized I had a lot to say about it! So instead, I've written a short book.

I'm passionate about this topic because I feel like it's so often underappreciated. While the portfolio site is just one piece of the job-hunting puzzle, it's a piece that can have an outsized impact. I don't want to oversell it—even the best portfolio site won't get you a job all on its own—but I've seen how a solid portfolio site can lead to more callbacks, more interviews, and ultimately more job offers. The goal of this book is to help you build a stellar portfolio that can increase your odds of success.

CHAPTER 1: FOUNDATIONS

WHAT IS A DEVELOPER PORTFOLIO?

A developer portfolio is a website that showcases the work that you've done as a developer. The idea is borrowed from the art world: photographers, for example, will often create a portfolio website that showcases their best work.

The goal of a developer portfolio is to present yourself and your work in the best possible light. Your portfolio should make employers excited to meet you. It should offer a glimpse into how awesome it would be to work with you!

Unless you already have extensive work experience, your side projects are the greatest asset you have. They are your trophies, and the portfolio site is the trophy case that highlights all the amazing stuff you've done.

Building side projects is a lot of work, and we want to make sure to squeeze as much job-seeking value out of them!

WHY DO I NEED ONE?

You may be wondering if you actually *need* a developer portfolio. You have a resume, and a LinkedIn profile. You can list your projects there, isn't that enough? Will employers actually take the time to visit a portfolio site?

Without a doubt, resumes and LinkedIn profiles matter, but I believe that a developer portfolio can be a kind of *secret weapon*.

LinkedIn and resumes are both focused on *work* experience. If you're early in your career, you don't have very much relevant experience. And while LinkedIn does let you add details about individual projects, the format makes it very difficult to share all the context. As we'll see, presentation matters, and the presentation on LinkedIn is less than ideal.

Portfolio sites let us guide potential employers through our projects, making sure they realize how *freaking cool* the stuff we built is, and how *relevant* our skills are for the roles

they're trying to fill. They allow us to make a strong case for our competence and experience.

Now, not all employers will take the time to look at our portfolio site. And, believe it or not, **this is a blessing in disguise.**

Most developers either don't have a portfolio site, or they haven't put much thought or effort into one. The reason for this is that "build a solid portfolio site" is rarely a top priority for recent bootcamp grads, or self-taught folks. Many developers skip this step, since they see it as optional.

We can imagine an alternative reality, one in which every developer has a polished portfolio, and every employer scrutinizes them rigorously. In this reality, it would be very hard to stand out from the crowd. We'd have to put in a *bunch* of work just to meet the minimum expectation! It wouldn't give us an *advantage* at all.

Happily, we don't live in that reality. In ours, *many* employers will check a portfolio site, but not all. For those that do, we'll have a huge advantage.

Unfortunately, job-seeking is competitive by nature; for every junior dev job-posting, there will be multiple candidates, and only one of them will get an offer. A polished, effective portfolio site can be just the leg-up you need.

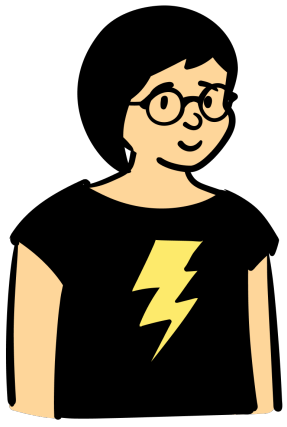
A portfolio site is a guided tour of your personal projects, a chance to show (rather than tell) that you have the skills needed for the job. Imagine if you're the only candidate to have a portfolio site listed with your application! I expect it'd be a serious competitive advantage.

Like I said: portfolio sites can be a secret weapon.

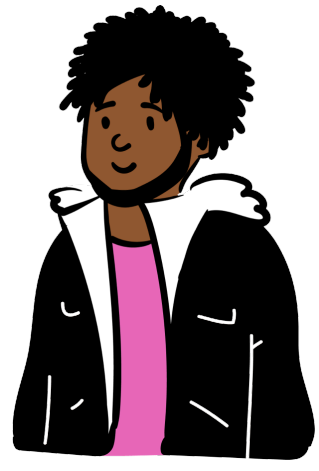
WHO IS THIS BOOK FOR?

I wrote this book for people who are trying to break into the tech industry, or who are looking to make early moves in their career. It assumes that you've been writing code for long enough to have built a couple projects, but not long enough to have an impressive work history. It assumes that you're looking to get a job working as a front-end or full-stack software engineer.

As I was writing this book, I kept 4 distinct personas in mind:

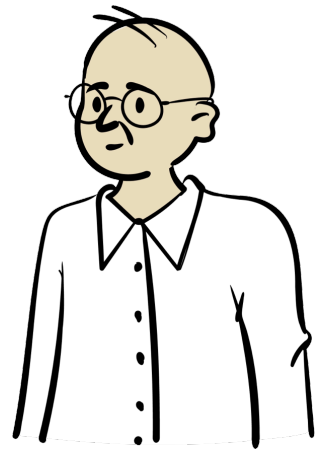


Morgan recently graduated from a 12-week coding bootcamp in Toronto. They have spent a few weeks polishing their final project from the course, and have started applying to positions.



Darius has been working as a junior front-end software developer for about 6 months in Austin. Unfortunately, he isn't enjoying his current role, and is looking to make a change. As he prepares to enter the job market again, he wants to give himself an edge by developing a solid portfolio site.

Priya is in her final year of Computer Science at UC Berkeley. She's participated in several hackathons and is particularly interested in full-stack development. She's contributed to several popular JS libraries. She's eager to get a job at a small start-up where she can have a large impact.



Levi is an office worker in Montreal who has been teaching himself web development through free online resources with React. It's been over a year of nights-and-weekend study, and he's even done a couple small freelance projects. To help with his office work, he created a small desktop application with Electron that lets him process invoices. He doesn't know if he's learned enough, but he's eager to change careers.

These people are all *early in their software development careers*. They all have *a body of work* they can share, and they're all hoping to *get a new job* working as a software developer.

I'm also focusing this book on front-end / full-stack software developers, but I expect that most of this content will also be relevant for back-end developers, mobile developers, or data engineers.

Caveats and Exceptions

- Software developers live and work all over the world, but I'm only familiar with hiring practices **in the United States and Canada.** If you're looking for a job in another part of the world, please understand that there may be differences that I am not aware of. Be skeptical of anything that doesn't sound right to you.
- This book assumes that you're **looking to get a job** as a software developer. If you're more interested in freelancing or starting your own company, the advice in this book is likely all wrong.
- If you don't yet have at least **1 solid personal project**, you should start with that. See the next section to get ideas for what to include.

CURATING PROJECTS FOR YOUR PORTFOLIO SITE

A portfolio site is a *showcase* of your work. But how do you know whether a project is worth showcasing? What sorts of projects qualify?

Ultimately, our goal is to show employers that we're competent at building stuff. There's a whole lot of different forms that this proof can take. Here are some examples, using our personas from earlier:

- **Morgan** spent 2 weeks at the end of their bootcamp building a restaurant reservation webapp, and they spent a few days afterwards tinkering and polishing it. The app is very much an MVP and it's missing a lot of features, but Morgan learned a lot building it.

Since graduating, Morgan has been volunteering at Code Dojo, a non-profit that runs bi-weekly workshops teaching programming fundamentals to kids. She's helped rework some of the curriculum.

- During the early stages of the Coronavirus pandemic, **Darius** spent a weekend building a Twitter bot using Ruby. It looks for people saying that coronavirus is "just like the flu", and automatically replies with links to articles that clarify this misunderstanding.

  Darius also spent 3 months in his first job contributing to a team that rolled out a notifications system.

- **Priya** has participated in several hackathons. For example, one time she teamed up with 3 other people to create an analytics platform that respects user privacy (A Google Analytics alternative with no GDPR banners!). She tackled the back-end side, ensuring that data is collected and processed effectively. The team completed a proof-of-concept, but have not taken any further steps to produce a real application.

  Priya has also contributed to several JS libraries, like PancakeJS and em-dash.

- **Levi** created a small Electron app to help him scan and catalog invoices at his office job. It was never released to the public, since it's tailor-made for his specific use-case, but he uses it every day, and has saved him about 2 hours of work each week.

  Levi also built a small single-page website for his local bakery, so that they could post their baking schedule, so that customers can always get fresh pies. 🥧

We're only scratching the surface — all kinds of different work qualifies. Morgan's volunteer work at a non-profit absolutely counts, even though there's no live demo or code samples. It still sends a strong signal about competence. Same thing applies for things like speaking at local meetups, being involved in the community, etc.

In my experience, certain kinds of projects are *especially* high-impact:

- ⭐ Did you build a project to solve a specific niche problem you had, like Levi's invoice-scanning app? Employers love to hear about projects like this, because it shows that you're able to apply your skills in creative ways, and tackle a project from start to finish.

- 🌎 Is your project “alive”? Is it shipped, and are people really using it for its intended purpose? It’s better to have a living, breathing project with real users (even if it’s only a handful) over a project that was built exclusively as a demo / for the portfolio.

- 🖥️ Have you contributed to a popular open-source library? Participating in the open-source community builds many of the “soft skills” that employers treasure! In order to get your change merged, you’ll need to be an effective communicator, be able to estimate tasks, and be good at receiving feedback.

  If you’ve contributed to open-source, definitely include it as a project! Just be sure to be very explicit about what your contributions were. Non-technical contributions, like adding documentation, are super valuable, and absolutely worth adding to your portfolio, but be careful; you don’t want to imply that you’ve contributed in ways that you haven’t.

If you don’t have any projects that fit these criteria, that’s OK! This list is meant to help people decide how to prioritize existing projects. It isn’t meant to exclude other kinds of projects.

PROJECTS NOT TO INCLUDE

Tutorial / Workshop exercises

Don’t include projects that were created by following a step-by-step tutorial. Same thing goes for workshops or exercises assigned by a bootcamp.

There are a few reasons for this:

- Projects are meant to show how you can use the skills you’ve accrued to build things in a self-directed, unguided way. Tutorials are a way to give you those skills, but the tutorial itself is not a demonstration of those skills.

- If you try to pass a tutorial project off as your own, this can backfire if the screener recognizes the project (more likely than you’d think!).

- Projects should be an accurate representation of where your skills are at. If you get hired on the belief that a tutorial project was your own invention, you’ll be assigned

work that could be well beyond your current skill level. This is a stressful nightmare scenario!

If you made *significant* extensions to the project, it might be alright to include. For example, if you were given an assignment to create a Tic-Tac-Toe game, and you decided to spend a week adding online multiplayer support. Or if you transformed it into a Sudoku game. A good rule of thumb: did you spend >50% of the *total time* on the extensions? If the bulk of your time was spent adding self-directed extensions, then I think it's worth adding. Just be sure to be explicit about which parts were original.

Non-Dev Projects

In an attempt to "pad" the number of projects, I've seen some folks add information about non-development projects like photography or carpentry.

There is a place for these kinds of details on your portfolio, but it shouldn't live amongst your other projects. You can include personal information in your About Me section.

Confidential Work

If you have previously worked as a junior dev / completed an internship, you may be tempted to include features that you worked on in that capacity. This can absolutely be a great idea, but you need to be careful and ensure that you're able to talk about it. Especially if you work for an agency, you can get in trouble; agencies often do work under a non-disclosure agreement.

When in doubt, ask your employer for clarification.

The Portfolio Site Itself

Some developers will list their portfolio site as a project on their portfolio site. In general, I believe that it's better not to do this; it gives the impression that you're filling space because you don't have other projects you can use.

The exception is if you invested time into parts of the site that are non-obvious from a user perspective. For example, did you build an "admin panel" to manage the content, like a mini home-built CMS? Did you code an analytics page to learn about traffic? Absolutely include it in this case!

HOW MANY PROJECTS TO INCLUDE?

At least 2. No more than 5.

Something which can be surprising: It's better to have 1 large polished project than 5 small projects.

You want to show that you can complete a non-trivial project from start to finish. Depth is more important than breadth, because it shows that you have the grit and determination to stick through the hard parts and finish a large project.

What if I have more than 5?

If you have a dozen projects that you feel are worth including, it's likely that you've gone *too wide* and *not deep enough*. Pick the 5 projects that best represent your skills and the work you want to do, and (if possible) consider spending a few days extending your largest project to be even larger.

Nobody will spend time looking at more than 5 projects. Most people will only look at 1 or 2. If you include 12 projects, a potential employer might only look at your least impressive one!

Portfolios are meant to highlight your *best* work, they aren't meant to be an exhaustive set of *all of* your work.

If you really want to include more than 5 projects, tuck the other projects behind a "show all" or "view archive" link. That way, you're setting a clear distinction between the "top-shelf" stuff and everything else.

What if I only have 1 project?

If you only have 1 significant project, you have a few options:

- Spend a weekend on a smaller second project. Consider building something relevant to your interests; if you're really into yoga, for example, build something that helps your

yoga practice. By picking something you're passionate about, you'll be more likely to want to work on it, and you'll wrap it up sooner.

- If you have an unfinished project, list it anyway, and note that it's "still under development".
- Is there a "spin-off" you could do of your first project? For example, if your first project was a full-stack web app that let users find food inspection results for local restaurants, could you maybe also create a Chrome Extension that spoke with the same server?
- Are there non-product things you've done that could be worth listing? Have you volunteered at any coding events? Have you contributed documentation to an open-source project? Think back through the time you've been learning to code, and see if anything pops up that could be used.

> *Ultimately, a portfolio site is only useful when you have projects to highlight. If you don't have at least 1 significant project and 1 minor project, consider spending some time filling in your portfolio. This can be an unfair burden on folks with caregiver responsibilities, or any other life situation that limits their time or energy. If it's unworkable, you might have to shift strategies; there are many ways into the industry! Unfortunately, this book's strategy assumes that you either already have suitable side projects, or else have the resources to create them.*

WHAT IF MY PROJECTS AREN'T GOOD ENOUGH?

It's very common for junior developers to feel like their work isn't impressive enough to get a job. You probably aren't super confident in your skills, and you might look at your projects and feel like they're nothing special / nowhere near as good as they could be.

This is all a *common and normal part* of entering the industry. Most of the senior developers who have successful careers started out exactly where you are; worried that they're not experienced or knowledgeable enough to get a job.

In fact, if you've spent many hours on a project, there's a very good chance that you've faced and overcome significant technical challenges. Projects often have *hidden complexity*, and

employers are eager to hire developers who are good at solving problems. The main purpose of a portfolio site is to show prospective employers how you ran into technical challenges, and overcame them. Your projects don't need to be flashy or feature-rich.

This is why a detailed portfolio is so important. LinkedIn lets you add a screenshot and a couple paragraphs about a project, but think about how incomplete that story is. Most junior-developer projects look the same on the surface; the interesting stuff is *below* the surface. And a portfolio site is a great chance to guide the reader through that interesting stuff.

Of course, there *is* such a thing as a project which isn't complex enough to warrant being put on a portfolio site. But folks tend to overestimate the level of complexity which impresses employers, when framed correctly. Remember, you're going for junior-level positions! Reasonable employers will understand that you're not an expert yet (and you probably don't want to work for unreasonable employers anyway, so nevermind what they think).

If you wait until you feel ready to start working as a developer, you'll never start!

If you identify as female, there's a good chance that you're underestimating your own skills[†], and overestimating the level required for these positions. If you're feeling like you're not as advanced as some of the men in your peer group (eg. fellow bootcamp alumni, members of the same "learning to code" community), it's worth considering if maybe your peers are overestimating their abilities.

[†] Source: *https://www.fastcompany.com/40554829/youll-never-guess-which-gender-tends-to-overestimate-their-own-stem-skills*

IN SUMMARY

- Development portfolios can be a **secret weapon**, precisely because very few aspiring developers put thought and energy into them.
- You want to highlight your best work in your portfolio. Between 2 and 5 projects is ideal.
- All sorts of projects are suitable for this section, but you should avoid:
    - Work created by following a tutorial or completing an exercise
    - Projects that aren't related to software development
    - Confidential work
    - The portfolio site itself, *unless* you can tell a compelling story about it that isn't just summarizing what the visitor has already seen

CHAPTER II: STRATEGY

UNDERSTANDING THE AUDIENCE

Before we start sketching or coding, we need to think about who we're making our portfolio for, and what we want to communicate to them.

When you apply to work at a company, there are two types of people who are likely to visit your portfolio site as part of a screening process:

1. HR (Human Resources) hiring managers
2. Software developers

Both types of people are trying to answer the same question—whether you're worth hiring—but they have very different sets of criteria.

Our job will be to create a site that communicates our awesomeness to both groups of people. To do that, we have to learn a little bit about what they're looking for.

> *This section is a generalization / oversimplification. Every company has their own process. Sometimes, HR hiring managers are technical. Other times, you'll only ever be interviewed by full-time software developers. It's a helpful generalization because we need to plan for the common case, but it's important to never assume that a real person fits neatly into one of these two categories.*

HR Hiring Managers

Unless you're applying to a very small startup, the company likely has an HR department. Depending on the size of the company, this department might have many employees who focus exclusively on hiring.

Generally, the first point of contact you'll have is with an HR person. This person likely isn't a software developer. Part of their job is to winnow the field; a typical junior developer job

posting might get up to 100+ applicants, and they'll want to reduce that number to something more manageable before scheduling interviews.

Here are some of the traits they're hoping we'll show in our application:

- **Competence.** Has this person built professional-looking projects before?
- **Enthusiasm.** Does this person seem eager to learn, excited for the kinds of work they would do on the job?
- **Fit.** Does this person have the right kind of personality for the company? Have they worked with the languages and frameworks we use here?

Critically, these folks are often very busy, since they have a lot of applicants to get through and many other job responsibilities to fulfill. They'll likely make a decision about moving forward in the process with you within 30-60 seconds. Our portfolio will need to capture their attention and communicate **competence, enthusiasm** and **fit** in a very short window of time.

Software Developers

It's important that our portfolio also has information that is relevant for software developers working at the company:

- At very early start-ups, your application will likely be screened by a developer, since they don't yet have an HR department
- Sometimes, HR hiring managers might reach out to a developer to help them evaluate a candidate during the screening process
- At all companies, you will almost certainly be interviewed by a software developer at the company, and this person may visit your portfolio beforehand, to learn about you.

The nice thing is that software developers were once in your exact shoes—they remember what it was like, searching for their first job!

Unlike HR hiring managers, developers generally aren't very interested in what languages or frameworks you know. In fact, for junior roles, developers care much more about your *potential* rather than what you happen to know already. They want to know if you'll be easy to mentor. They want to know if you have the grit to work through tough problems, and the humility to admit when you don't know something.

It can be hard to communicate this stuff through a portfolio site, but there are definitely things we should *avoid* doing. We'll discuss them more later in this chapter.

A GENERIC EXAMPLE

The main reason I'm writing this book is that, after reviewing hundreds of individual developer portfolios, I've realized that most junior developers aren't thinking strategically about how to stand out to these groups.

Let's take a look at a contrived portfolio I've built. As you review this portfolio, try and see it through the eyes of an HR professional, or a senior developer. Make some notes about what you like, what you don't like, and what's missing.

View it live: https://generic-portfolio.now.sh/

Squint at a screenshot, on the next page

Charlie Smith

About me Projects Contact

Full Stack Software developer in Anytown, USA

Looking for full-time opportunities

About me

I am a junior full stack software developer from Anytown. I am passionate about web development and about languages such as HTML, CSS, and Javascript.

I recently graduated from Apex College's 12-week intensive web development bootcamp and I am looking for a full-stack position.

When not coding, I enjoy hanging out with my friends.

Projects

Here are the projects I have completed:

Analytics app

Application dashboard. Built with React, Redux, React Router.

Travel booking

Website to book travel. Built with Vue.js, Vuex.

Marketing landing page

School project. Built with vanilla HTML and CSS.

Contact me

Let's work together!

Your name

Your email

Your message

Submit

Charlie Smith

Structurally, this portfolio has all the right pieces: it includes an "About me" section that tells the employer about the candidate, a list of projects they've built, and a contact form.

In fact, most of the portfolio sites I've reviewed have been similar to this one. If you've built a portfolio before, it's likely quite similar to this!

The main reason we have a portfolio is to showcase our strengths. This portfolio isn't really showcasing anything. It's not helping the candidate at all; I suspect that if this candidate applied to 100 jobs, and included a link to this portfolio in 50 of those applications, there would be *no difference* in terms of response rate / number of booked interviews. This portfolio may as well not exist.

This is a perfectly acceptable generic portfolio, but we aren't going for generic. We want to stand out, not blend in. This portfolio is missing opportunities to convey the stuff employers care about, like competence, enthusiasm, fit, and personality.

Let's go over it in depth.

About Me

This is the "about" section on this portfolio:

Here's the biggest problem with this section: if you swap out the names of places and schools, *it can apply equally well to every single bootcamp graduate*. The only personal touch is the last line, and it's hardly unique; everyone enjoys hanging out with their friends.

In terms of tone, it uses that "professional" tone people take when trying to get a job in the corporate world. I suspect that this tone might still be relevant when applying to work in a different industry, but it's the wrong tone for the vast majority of tech roles.

What should an "About Me" section look like? We'll discuss more in Chapter 4, but here's a quick example of how I might write this for myself, when I was just starting out.

> *Hi there! I'm Josh. I'm a passionate introvert who loves building things with code.*
>
> *My first experience with programming was when I was 12, but I didn't last long: I was a big fan of computer-animated TV shows like* Reboot*, and I decided I wanted to do "computer graphics". My mom bought me a C++ Reference Manual. It was several hundred pages, and totally inscrutable to me.*
>
> *Years later, I decided to try learning Python. It was a radically different experience. I distinctly remember the moment it clicked; I was watching* The Price is Right*, and they were spinning the big wheel. A contestant landed on 0.70, and it made me wonder: statistically, what was their best move?*
>
> *It occurred to me: I had the skills to build a simulation! So I threw together a Python script that would run thousands of simulations to work out what the right answer was. 15 minutes later, I had the answer. My burgeoning skillset was a superpower: I could derive answers that were previously off-limits to me. It was magic.*
>
> *Since then, I've been honing my skills and learning Javascript. I recently graduated from Apex College's 12-week intensive Web Development Bootcamp program. I'm seeking a full-time role where I can help a company achieve their goals.*

This "About Me" section wouldn't work for anyone else (well, unless they were willing to make up a big lie about their history!), and that's the point. It's true to who I am, and shares a compelling window into my personality and experience.

Your personality is likely different than mine, so your "About Me" section should be different too! You can be more formal, if that's where your comfort zone is. You don't have to share a quirky story about how you discovered programming. But it should be interesting, and it should open the door to more conversation. It should sound like something you might say in real life.

Let's look at this through the lens of an HR hiring manager. You have a stack of applications to get through, and you can only realistically give 10-20% of them interviews. You click through to Charlie's portfolio site, and you see the original generic "About me" section. You spend 3 seconds skimming it, and decide to skip to the next section, since there's nothing *of interest* there. Unless something else on the page catches their eye, they'll likely move straight on to the next applicant in the pool. By the time they get to the end, they'll have forgotten yours entirely.

Compare that to the revised description, with the anecdotes. As their eyes skim the section, they catch "*The Price Is Right*", which is a very unusual thing to spot on a portfolio site! They start reading from the beginning, and they get hooked in. They're interested to know more about how you solved the problem, and what the answer was (should the contestant spin again??). You've created a memory, and a mild emotional response. After getting through the stack of applicants, it's likely that yours will still be in the back of their mind. *Most* portfolio sites are generic, and it doesn't take much to have yours stand out.

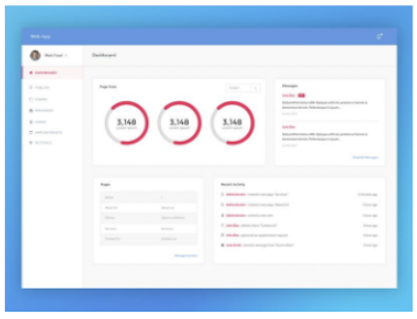
Hiring managers care about all the things you'd expect — work experience, volunteer experience, projects, education, skillset — but they're also looking for people they'd enjoy working with. A quirky (and, critically, inoffensive) anecdote can endear you to them. Someone in that stack might have had a lot more experience than you, but if they forget that the person exists, they won't get the job. You might.

Projects

In my generic portfolio example, the biggest missed opportunity is the projects section. Tragically, this is also the part most similar to common, real-world examples:

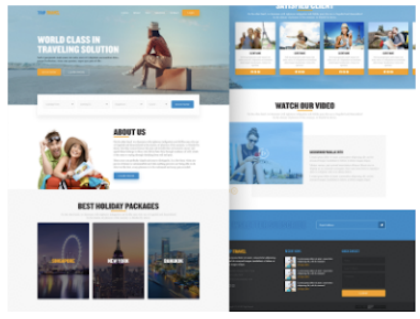
Projects

Here are the projects I have completed:



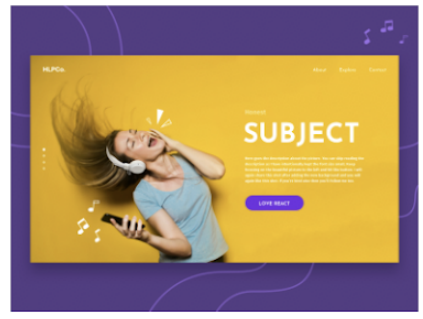
Analytics app

Application dashboard. Built with React, Redux, React Router.



Travel booking

Website to book travel. Built with Vue.js, Vuex.



Marketing landing page

School project. Built with vanilla HTML and CSS.

Each project is given the following information:

- A mini screenshot of the product
- A generic name
- 1 sentence description
- A list of technologies
- A link to the Github repo
- A link to a live demo of the product

You might feel like this is sufficient. After all, for those who are curious, they can follow the live link to get a demo, right? And software developers can follow the Github link to learn more about the implementation?

In truth, most people won't click on either icon. And those that do will likely come away unimpressed (or, at least, far less impressed than they should be!).

Be a tour guide

You don't want visitors to wander around on a live demo, or go spelunking in the codebase. We want to highlight the most impressive and interesting parts of our products!

With a live demo, you're taking a lot of risks:

- Have you tested your site on the browser and/or device the visitor is using?
- Will they follow exactly the same flow you've imagined? Unless you have a background as a senior UX designer, it's likely that they won't!
- Does your app require a signup, or any other high-friction setup? Most people don't have the patience to sign up for things.
- Have you found and fixed all possible edge-case bugs? Live demos going awry happens so much that it's cliché.

Even if everything goes flawlessly, think about what *isn't* conveyed in a live demo:

- Why is this project important to you? What inspired it? Why did you choose to build this?
- What are the major features that make it unique? How does it compare to existing products like it?
- What did you start with? Was this built from scratch? Did you have a team? If so, which parts did you do? Where did the design come from? Was there any collaboration?
- What was the hardest part of building this product? Where did you get stuck along the way?
- When you did get stuck, how did you resolve it? How did you overcome the obstacles you faced?

- What did you learn from doing this project? How has it affected the work you've done since then?

When it comes to the codebase, there are likely areas where you were able to come up with a clever solution to a hard problem, and other spots where you duct taped things together, or where things are working but you're not sure why. Every codebase has dusty closets. If a developer does check the link to your Github and start poking around, will they see the best bits?

Earlier, we spoke about how a portfolio lets us *squeeze value* out of our personal projects. We built a cool thing, but nobody knows about all the stuff that makes it compelling!

We want to **tell a story** about how a project came to be. It's much more interesting than showing a final product, without context. It also sends a much stronger signal to employers about what you can do!

It's probably still a good idea to include links to a live demo, mostly to serve as proof that you actually built the thing you're describing. But it should be de-emphasized. Most folks won't check. The people who do will likely take 10 seconds to say "yep, this is what they said it would be", and bail.

> *What if your project isn't a tangible product, like volunteer experience? Happily, the "be a tour guide" strategy is a natural fit for this kind of project. Follow the same process, answering the list of questions, and include photos when possible.*

Narration and Personality

You may be sensing a theme here: both the About Me section and the Projects section should have an element of storytelling. We don't just want to talk about where we are now, we want to share some history of how we got here.

Most folks didn't decide when they were 5 years old to become a programmer—we took a winding path that led us to where we are now. We want to capture that.

For bonus points, you can have fun with the details. I call them **sprinkles**—small UI flourishes that make your portfolio stand out, and become much more memorable. This can be things like animations, interactions, and surprising details.

We'll talk much more about this in the chapters ahead.

Design

This portfolio has a so-so design. It isn't hurting, but it isn't helping either. In Chapter 4, we'll talk more about how to improve the design of our portfolio sites without needing to become a great designer.

THINGS TO AVOID

Skill charts

Many developers will include a bar graph or some sort of data visualization showing their skill levels in various languages/frameworks:

Please, please don't do this. It's absolutely going to do more harm than good.

First, the numbers are meaningless. What does *85% Javascript* mean? Surely it can't mean that you know 85% of everything there is to know about the language; I've been writing JS since 2009 and I'm at maybe 50%.

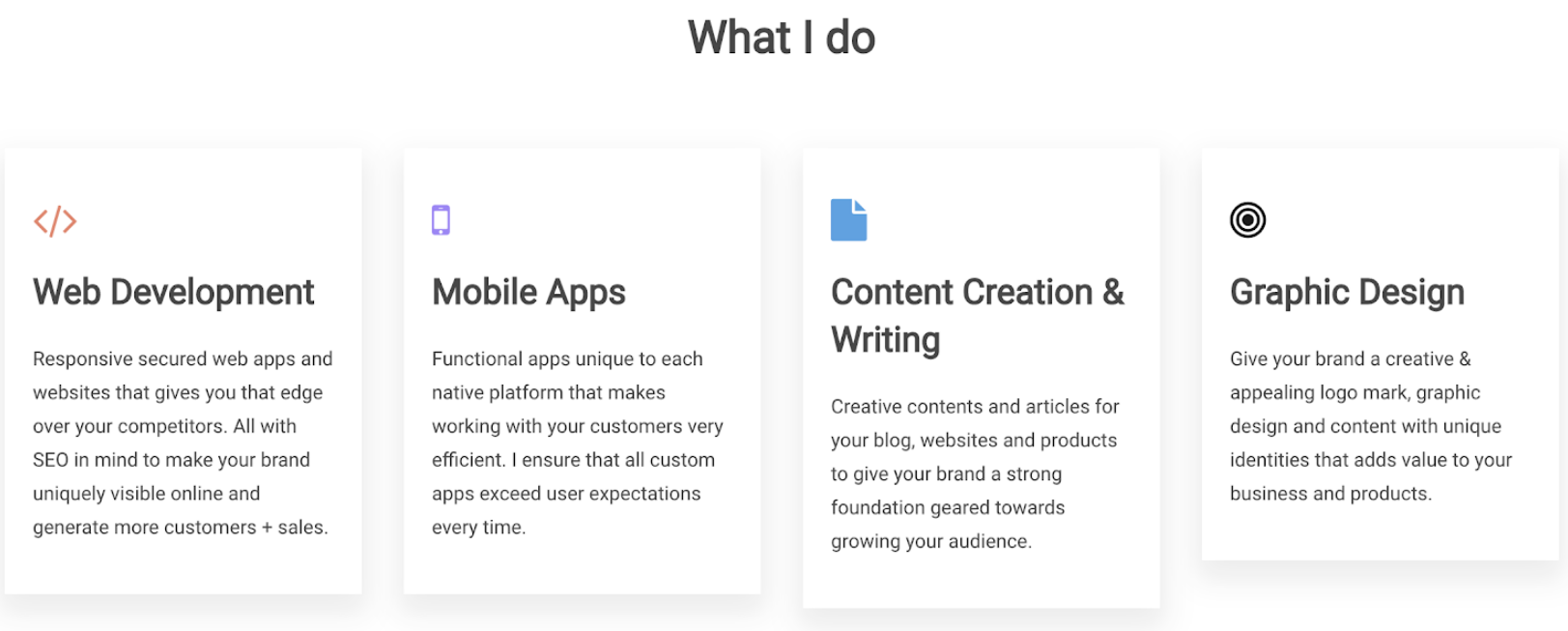
It's a no-win scenario. If the number you pick is too high to be realistic for someone of your level, you'll be seen as boastful and overconfident. If the number you pick *is* realistic, it will probably be low enough that it just doesn't look very good (who wants to hire someone with *20% Javascript*?)

These numbers are absolutely meaningless, and it's likely that the reader doesn't share your scale. I've never heard of one of these data visualizations helping a candidate, and I've seen them hurt the candidate quite often.

I know it can be tempting, because it can be a pretty fun element to build, from an engineering perspective. But it's a dangerous move. I advise skipping any sort of quantitative measure of your skill levels.

"What I Do"

Some portfolio sites have a "What I Do" section, which outlines a handful of high-level skills and abilities:



This sort of section is commonly seen on *freelancer* portfolios. If you're trying to start a consultancy or get short-term contract jobs, this can be a good way to highlight the services you offer.

This book is targeted at folks looking to get full-time software development jobs, though. In this case, it makes a bit less sense; you shouldn't be pitching your content-creation or graphic design skills if you're applying for a junior dev job.

While having complementary skills is a good thing, this sends a confusing message to prospective employers. it's a positioning problem; It makes it seem like you're not looking for full-time employment, because it fits a different mold.

I'd suggest maybe mentioning complementary skills on your resume, or in an interview. It doesn't need to be added to your portfolio site; save that real estate for your primary skillset.

Bravado

I've seen a few junior dev portfolios that include language like:

- *Head and shoulders above the rest*
- *Master of Javascript*
- *10x junior developer*

Most companies hire junior developers based on *potential*, not based on your current level of expertise. In fact, when a senior developer reviews your portfolio, they're hoping to see that you're eager to learn, humble, and looking for guidance; folks like that tend to be easier to mentor, and make for better teammates.

The problem with this language is that it gives the impression that you're finished learning. If you've mastered Javascript, how are you going to react when you submit some code to be reviewed, and a lot of changes are requested?

It gives the impression that you'll be hard to work with.

This is admittedly a rare problem; most folks *under*estimate their abilities, not *over*estimate them. But this still does happen sometimes.

IN SUMMARY

- We want to target our portfolio at two cohorts of individuals:
    - **HR Hiring Managers** are looking for competence, enthusiasm, and fit
    - **Software developers** are looking for folks with grit and determination, and who will be easy to mentor
- Structurally, our portfolio should feature 3 things on the homepage: an "About Me" section, a list of projects, and a contact form (or other contact method).
- In terms of tone, your portfolio site should reflect your personality. It should be specific enough that it would obviously be plagiarism if someone else tried to use your "About me" section. Avoid generic platitudes or corporate-speak.
- For your "Projects" section, live demos are overhyped and underutilized. We want to be a tour guide, guiding readers through the most impressive parts of our projects. **Tell a story.** We'll learn more about how to do this in the coming chapters (this is arguably the most important part of this book!)
- Be creative with **sprinkles**—small bits of flair that showcase your enthusiasm for development. This is especially relevant for folks who are primarily interested in front-end roles.
- **Avoid skill charts**, or any other quantitative representation of your skills.
- **Avoid bravado.** You want to make it clear that you're eager to learn from more experienced developers, not that you think you already know everything.

CHAPTER III: CASE STUDY

Of the hundreds of portfolios I have reviewed, the best one I've seen is by Julia Johnson, a student at Purdue University, and an intern at IBM.

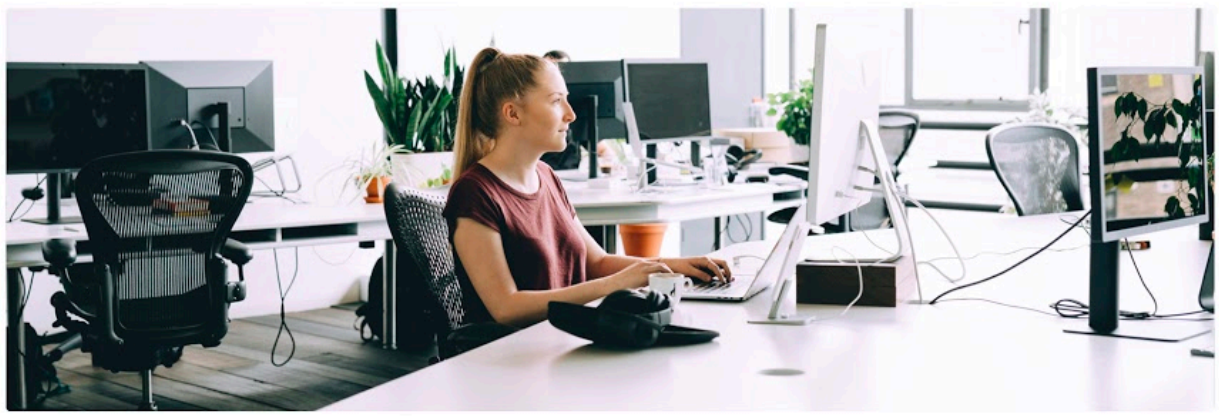
With Julia's permission, I've used her portfolio in this book as a case study. Let's look through some screenshots. As you read through at, keep in mind all the stuff we've been talking about. Try to see it through the eyes of a company looking to hire.

You can view these screenshots online at:

- http://joshwcomeau.com/images/effective-portfolio/julia-home.jpg
- http://joshwcomeau.com/images/effective-portfolio/julia-project.jpg

JU
LIA.

Hi there, I'm Julia.
Front End Developer,
Student and Minimalist



I'm a 20 year old student at Purdue University currently studying web development and design.

As I've grown as a developer, I've worked alongside senior designers and developers who have raised my standards for whats expected of any web application.

Through these experiences, I've had the opportunity to create memorable products that are not only enjoyable to use, but are written in code thats maintainable and easy to understand.

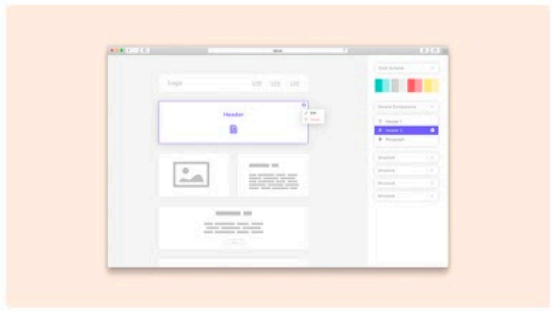
My Skills

Apart from the courses included in my degree, I've taken a number of online courses such as The Complete Javascript Course, Advanced CSS & Sass, React: the Complete Guide, Javascript 30, and I'm currently taking ES6 for Everyone

- HTML5
- CSS3
- Javascript ES6
- React
- React Native
- Styled-Components
- Git
- PHP & SQL

What I've been working on

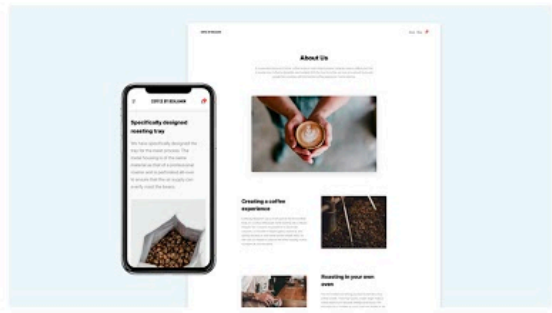
I like to stay busy and always have a project in the works.
Take a look at some of the applications, articles and
companies I've dedicated my time to.



Decore

A web application that builds custom starter
layouts for developers

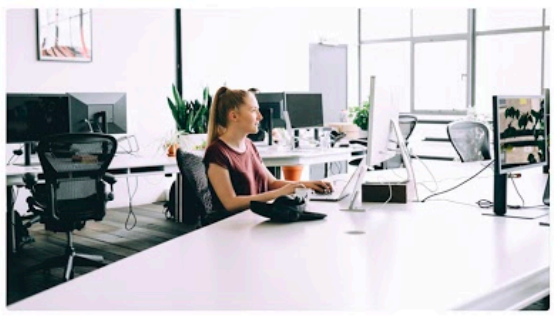
VIEW PROJECT›



Coffee By Benjamin

Developed a shopify e-commerce application
with React for a coffee roasting company

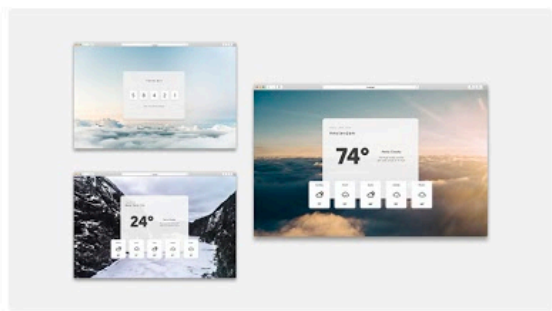
VIEW PROJECT ›



Working at Awkward

Read about my experience as an intern at
Awkward, a digital design agency.

READ STORY ›



Forecast

A 7 day weather application that sets weather
data based on user input.

READ EXPLANATION ›

Working at Awkward

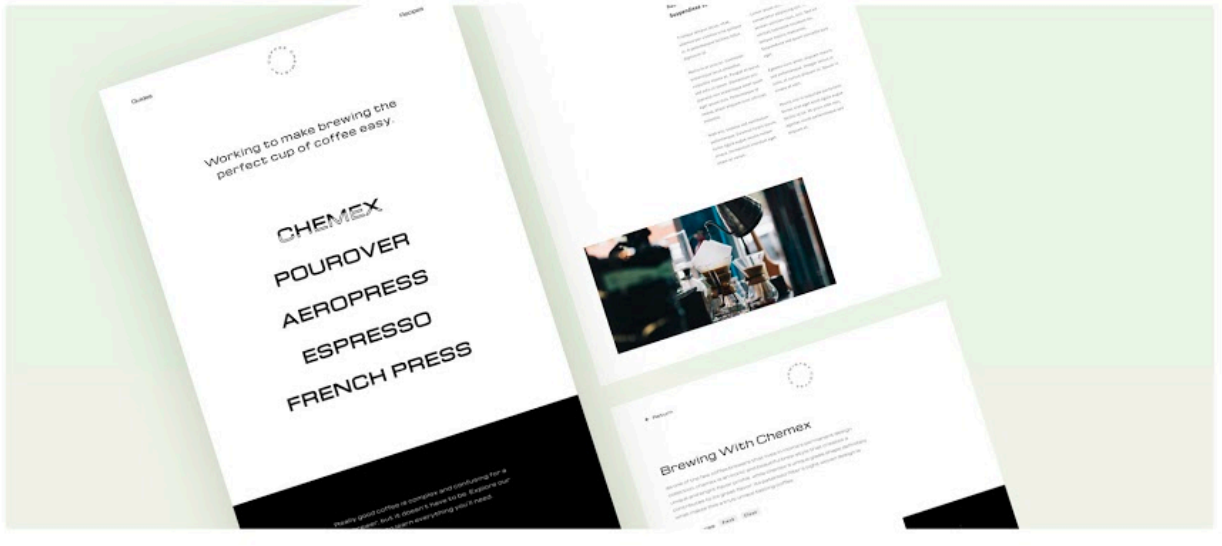
Read about my experience as an intern at Awkward, a digital design agency.

READ STORY ›

Forecast

A 7 day weather application that sets weather data based on user input.

READ EXPLANATION ›



Coffee Chemistry

My current work in progress, a web application that helps you brew the perfect cup of coffee.

COMING SOON

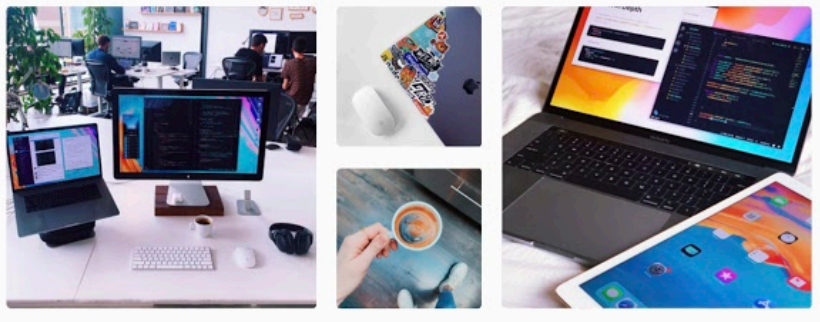
Let's Build Something Together

Feel free to reach out if you're looking for a developer, have a question, or just want to connect.

juliajohnson@purdue.edu

I'm a lot cooler on instagram

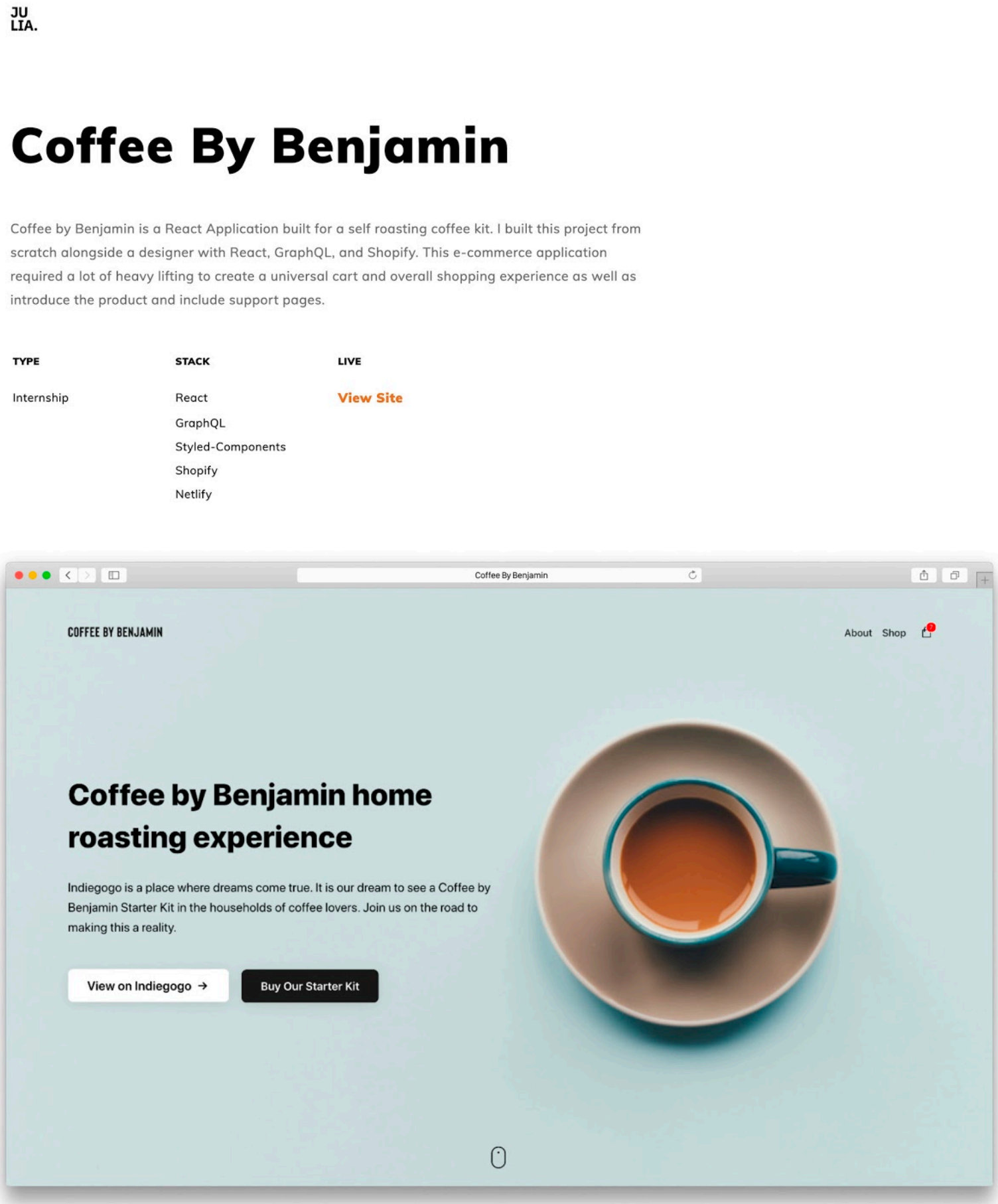
SEE MORE ›



JU
LIA.

github instagram

This is a great homepage, but the portfolio *really* shines when you click on one of the listed projects. Here are some screenshots from the second listed project:



Project Purpose and Goal

This project included 3 phases and iterations of the site. Phase 1 simply allowed users to enter their email to be alerted to when the product was released. Phase 2 was quite larger and is designed to introduce users to the Coffee By Benjamin product and answer any questions they may have. Phase three is by far the largest and most complex, as it includes the full shop and cart pages as well as the logic and backend that goes along with it.

I found that the best way to implement these 3 phases without having separate versions saved was to incorporate a feature flag that will pass the current state that should be displayed and then render content conditionally.

Web Stack and Explanation

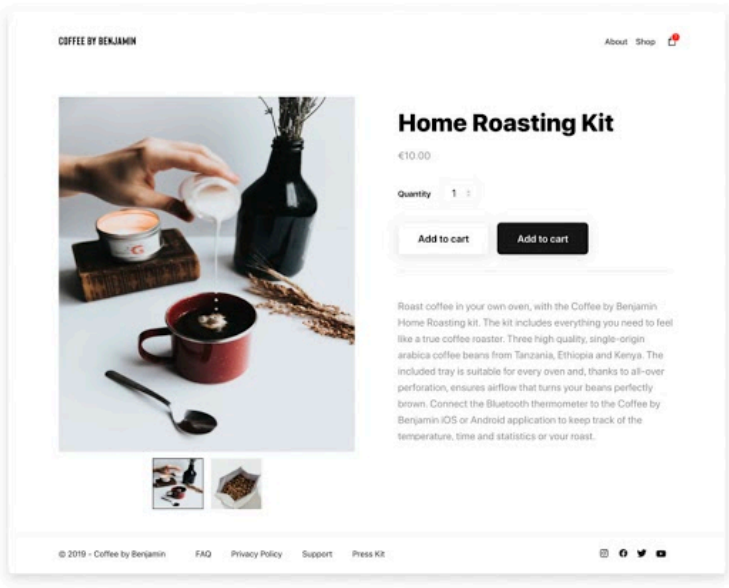
React

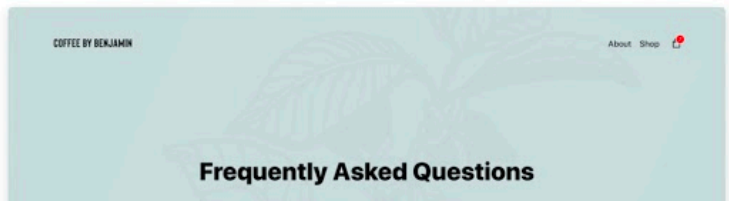
GraphQL

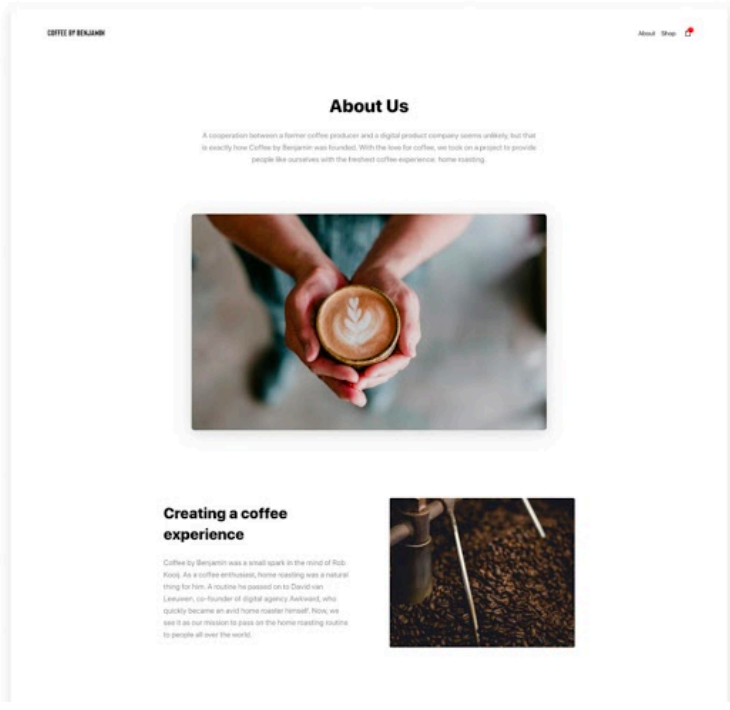
Netlify

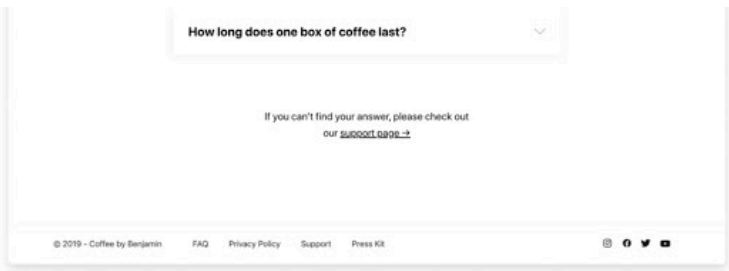
React made the most sense for the web application because it required to connect to graphQL and the Shopify-SDK for javascript ties into React very smoothly. The Shopify-Buy-SDK was chosen because of the ability for the client to modify the products without any complex coding knowledge.

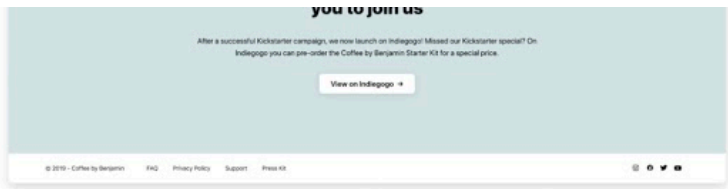
React hooks and session storage are also used throughout the project to maintain the user cart items and allows the cart count and other shopping data to be displayed universally without the need for Redux. Netlify is also an obvious choice for deployment because of its speed and reliability.







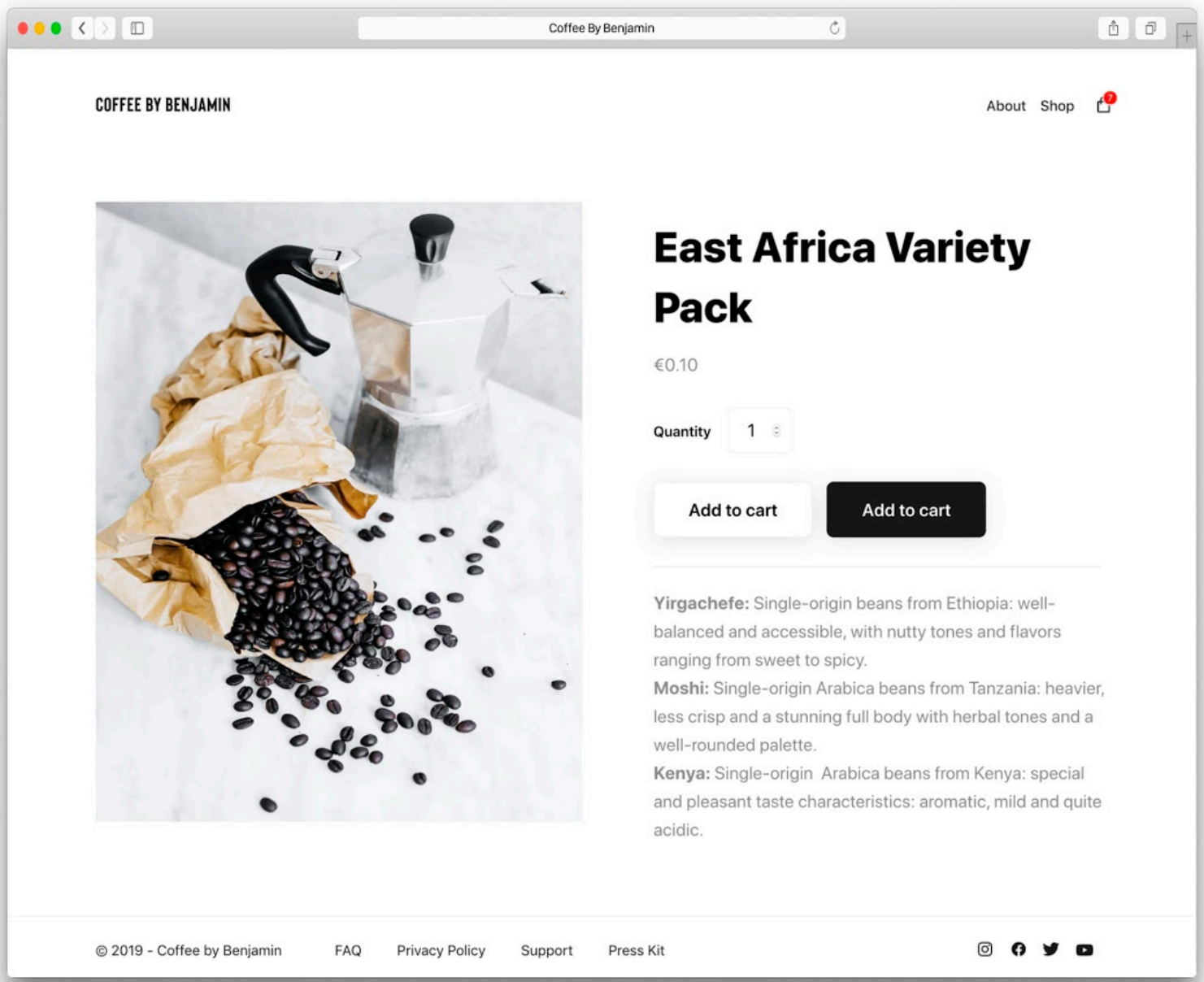




Problems and Thought Process

Like most projects, I ran into a few bumps along the way, one particularly difficult area was organization and structure of the code. Because of this project's size, I realized how important maintaining an organized structure would be.

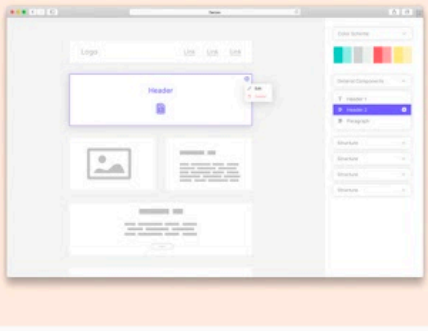
I worked hard to keep components as reusable as possible and utilized props for many slight variations. I also used styled-components, because the structure of CSS-in-js is much clearer and prevents overrides.



Lessons Learned

I could spend all day describing the lessons that I learned while working on this project, but the most important ones involved my newfound understanding of React Hooks, Git management, Feature Keys, and API integration. As my first large project using React, I learned a lot of lessons regarding code structure and organization. When I first began, I would write sloppy code and move on, but now I know how doing so can come back to bite you; I now spend a lot more time refactoring and improving every line I code I write, for the best readability and far fewer headaches.

Other Projects



Decore

A web application that builds custom starter layouts for developers

VIEW PROJECT›

Let's Build Something Together

Feel free to reach out if you're looking for a developer, have a question, or just want to connect.

juliajohnson@purdue.edu

Let's take a look at how this portfolio achieves the attributes we spoke of earlier.

Competence

The project page details a lot of thorny technical challenges, including:

- Using feature flags to incrementally release multiple versions of the product
- Using GraphQL and the Shopify SDK to fetch items from Shopify's headless API
- A focus on reusing code to boost maintainability

You'll notice that Julia never says things like "I'm a fantastic coder" or "I built an amazing project". Julia *shows* rather than *tells*.

Recruiters and HR professionals like to see specific keywords that match the tools and technologies a company uses. Julia lists her skills on the homepage, as well as technologies used for specific projects (in the example we're looking at: React, GraphQL, styled-components, Netlify). She even shares *why* she chose the tools that she did, getting a little extra value out of this section.

Enthusiasm

The impression from reading this site is that Julia really enjoys doing this kind of work – she's taken many courses beyond the content taught in her formal education, and she "always has a project in the works".

There are many ways to demonstrate enthusiasm, but doing it *implicitly* is always better than calling it out *explicitly* ("I am very excited to work as a developer…")

Telling a Story

Julia shares a small amount about her educational background and her priorities, and speaks briefly about the context around the *Coffee by Benjamin* project. She shares the plan of action for developing the application in 3 phases, and what she learned along the way.

I do think there is even more of an opportunity to share some more background. It would be great to hear why she chose to work on a coffee product in the first place, how she collaborated with the designer, and what roadblocks they hit along the way in non-technical

terms (eg. last-minute client requests, design hurdles, etc). Sharing these details can help add a bit of color to the picture, and make the narrative more compelling and memorable.

Humility and Grit

On Julia's homepage, she says:

> *As I've grown as a developer, I've worked alongside senior designers and developers who have raised my standards for what's expected of any web application.*

I love this line. It suggests that her skills are strong for her level, but also that she's easy to mentor. It's exactly the attitude that employers are looking for in junior candidates.

On the project page, Julia mentions that she faced some difficulties with organization and code structure. She was able to overcome these challenges by focusing on creating small, reusable components, and leveraging styled-components. Instead of seeing these challenges as problems, she saw them as learning opportunities; the project summary ends with:

> *I now spend a lot more time refactoring and improving every line I code I write, for the best readability and far fewer headaches.*

Sprinkles

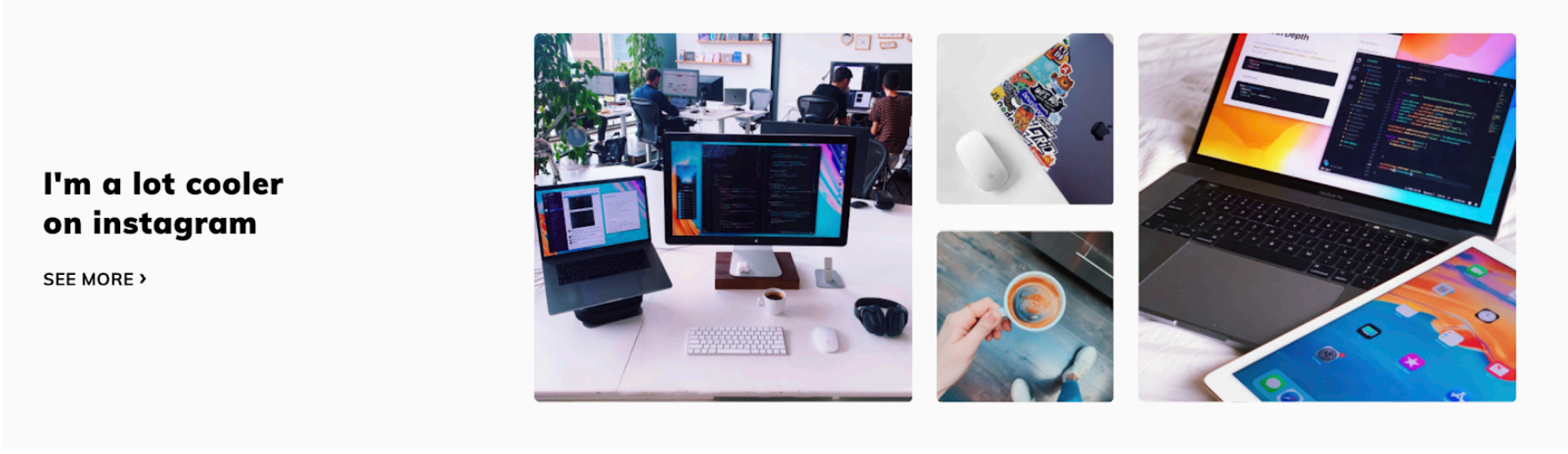
The page features an elegant sequence of on-load animations:

- The main heading fades in, and each line is staggered (They animate in order)
- A line extends, pushing the social media icons out
- The hero image fades in, after a delay

These individual animations work together to create an incredibly polished experience. It's a flourish that captures attention, and makes a sight more memorable.

You can view this animation yourself at https://www.juliacodes.com/ (though because this isn't my site, I can't guarantee it'll be the same now)

Linking to her personal instagram with a curated selection of photos makes this portfolio unique to Julia, a delightful touch:

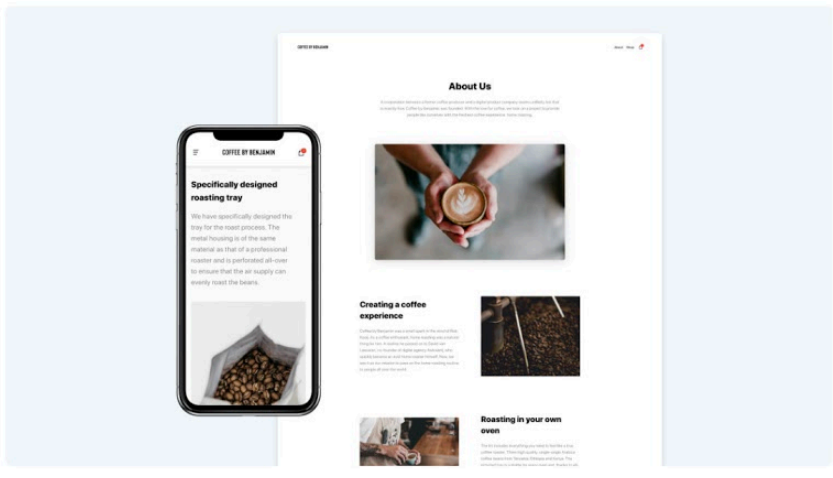


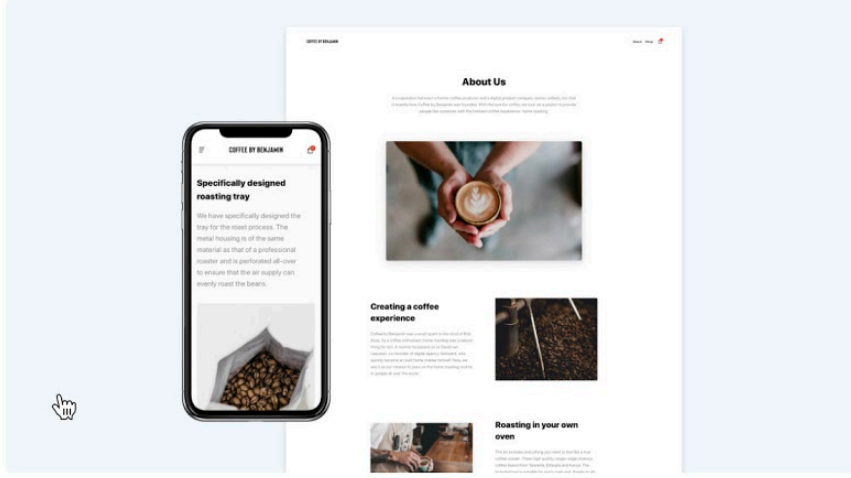
If you pop open the developer console, an easter egg is hidden within:

```
#      #       #######                                          ###
#      # #        #     #     # ###### #####   ######  ###
#      # #        #     #     # #       #     # #       ###
#######  #        #     ######  #####   #     # #####    #
#      # #        #     #     # #       #####   #
#      # #        #     #     # #       #   #   #       ###
#      # #        #     #     # ######  #     # ######  ###
```

```
This website was designed and built by Julia Johnson
```

Finally, the website has subtle animations on interactive elements. For example, project thumbnails zoom on hover:





It's hard to tell from these screenshots, but mousing over a project enlarges the image, and adds an underline to "VIEW PROJECT".

IN SUMMARY

Julia's portfolio is a delightful tour through her work, in a way that shines a light on the most impressive aspects. It's a great read.

It's impossible to say how much of a role her portfolio played in her job hunt, but she's clearly been very successful; she received several prestigious internship offers, and currently works at IBM.

Your portfolio should look very different from Julia's—because you're a different person—but it should include all the same signals. It should convey to employers that you're competent and enthusiastic. It should tell a story about your experience and your projects.

In the next chapter, we'll look at the practical things you need to know to build a portfolio site.

CHAPTER IV: BUILDING

We've covered a lot of the high-level theory around what makes an effective developer portfolio. Let's break it down further and talk about how to design and build one.

DESIGN

As a junior or aspiring software developer, does the design of your portfolio matter?

Unfortunately, I think it does. We don't need to have a stunningly-beautiful portfolio, but it needs to look reasonably well-designed.

Even though we aren't applying to work as a UI/UX designer, the people reviewing our portfolios are human, and humans like things that are well-designed. A poorly-designed site can make people feel anxious or uncomfortable, and we don't want to be associated with those emotions. Even if the reviewers make a conscious effort to disregard the design, I suspect it has an effect, and it ultimately colors their impression of us.

The Aesthetic-Usability Effect

Users are more forgiving of a website with usability problems if it looks pretty. Moreover, this happens subconsciously; people believe that if something ***looks*** *nice, it will* ***work*** *well, even if they've experienced usability issues firsthand. This is a well-studied † phenomenon.*

Software development is all about how something ***works****, not how something* ***looks****. The unfortunate reality is that people might judge our software-development skills based on how aesthetic our portfolios site is. Having a nice design can subconsciously suggest that we know how to build good software.*

† *Source: https://www.nngroup.com/articles/aesthetic-usability-effect/*

Design is hard. People study for years to become competent designers. You're likely already spending lots of time learning development, and you shouldn't be expected to invest a ton of time into design skills!

So how do we procure a nicely-designed portfolio site without learning all about design? The easiest approach is to copy existing designs. As we've discussed previously, though, we don't want to plagiarize an existing portfolio's design. Instead, I suggest looking for some design *templates*.

At least in my eyes, copying a template doesn't feel as slimy as copying a specific person's template. But we probably shouldn't create a mirror copy. My favourite trick is to find 2-3 sources of inspiration, and merge them into something unique. For example: you can take the layout and spacing from one design, the color scheme from a second design, and an interesting hero detail from a third.

You should always credit the sources that you take inspiration from, unless the inspiration is so subtle that nobody would be able to tell.

Finding a balance

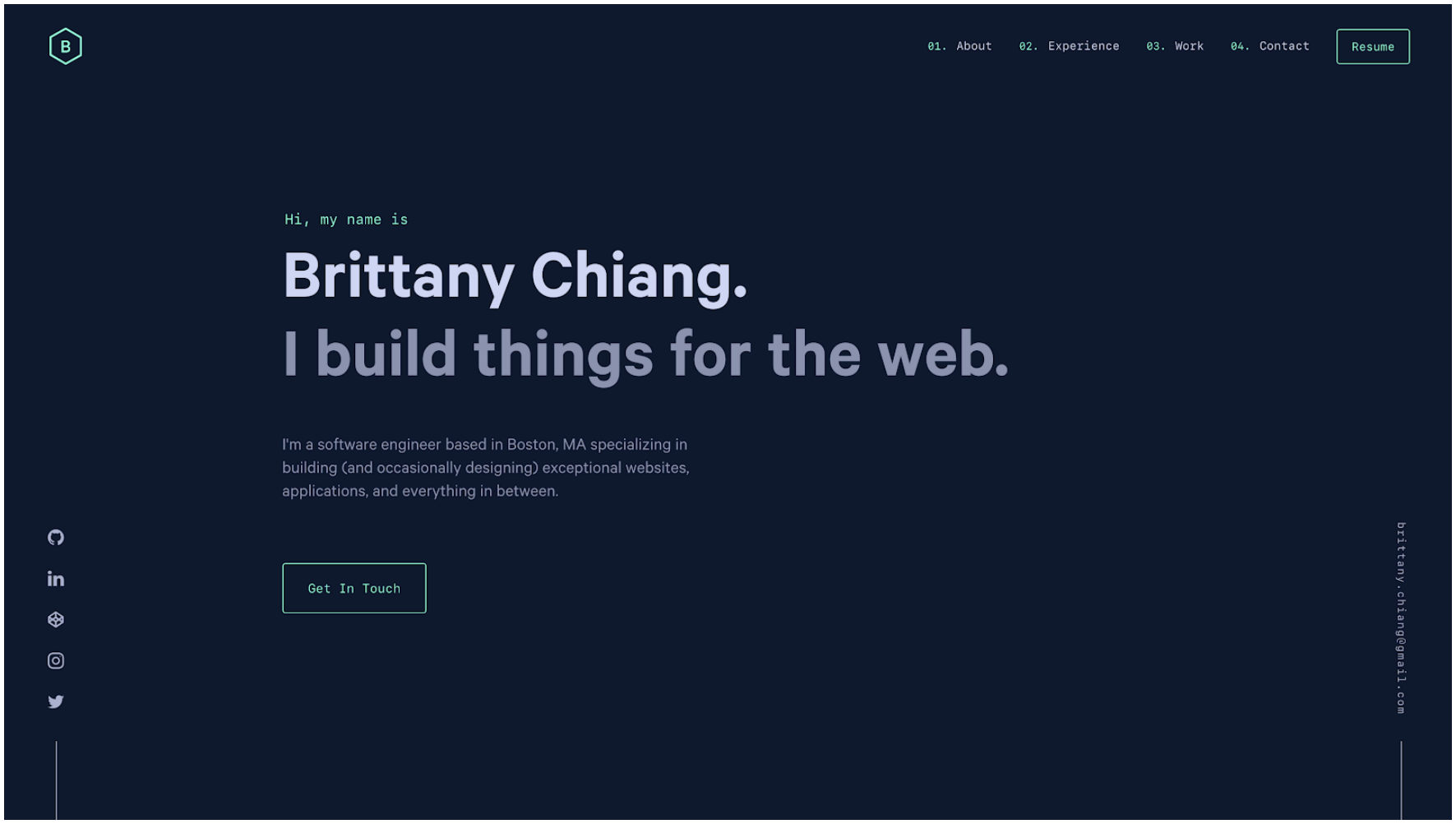
How blatant should you be? My personal standard is that it should be impossible to mistake your site for the thing you're copying from. It's alright if it looks *inspired* by the source, but it should always be clear that *you* built this, and you didn't just copy/paste the HTML and CSS.

The thing is, the further you deviate from an existing design, the more likely it is that you will inadvertently offset what makes the site look professional. This is especially true of spacing; I have no idea why, but even subtle shifts in spacing can break the polished feel of a template. My suggestion is to pick one template you really like and be vigilant about matching the sizes and spacings of elements.

For some example portfolio templates, check out Appendix A. Be creative, as well; feel free to take inspiration from a website you like, even if it's not a developer portfolio.

Crossing the line

One of the most well-known and well-respected portfolio designs comes from Brittany Chiang:



When I was reviewing junior developer portfolios, I saw **several** people steal this design, pixel for pixel.

This design is very unique, and recognizable. I suspect the people who copied it will have a hard time finding work, as recruiters *will* recognize that they're trying to pass off someone else's site as their own.

Please don't do this. Even if the portfolio is open-source, with a permissive license. It leaves a bad taste in the mouth. It's alright to take inspiration, but it should never be so blatant.

FRAMEWORKS AND TOOLING

Once you've conceived of a design, it's time to start building. How should you do it?

There is somewhat of an open debate as to whether developers should code their templates, or if it's acceptable to use a no-code tool like SquareSpace or Webflow. My personal opinion

is that if you're seeking a front-end or full-stack role, it's better to create your site with code.

It's likely that many HR professionals won't be able to tell the difference, but developers will know. Whether reasonable or not, some developers will think it's weird that you opted to use a site-builder when you're claiming to have front-end development skills.

I think there is a valid argument that site-builders are quicker to build with, and if you have a rich library of portfolio projects that can demonstrate your front-end skills, perhaps you'll be able to make the case that you were being efficient with your time. But you might not be able to make that case; you just won't hear back, and you'll never know why.

For this reason, I think it's best to create your site from scratch.

Which tools should you use to build your site? There are plenty of options, including:

- Vanilla HTML/CSS/JS
- 11ty (https://www.11ty.dev/)
- Gatsby (https://www.gatsbyjs.com/)
- Next (https://nextjs.org/)
- Jekyll (https://jekyllrb.com/)

It's probably best to pick the one you're already comfortable with, since we want to spend our time on the portfolio's substance.

Personally I would pick **Gatsby.** I am biased—I used to work for Gatsby!—but in my opinion it's a fantastic tool for the job. There is a rich library of Gatsby themes and plugins that can reduce the amount of time you have to spend building your site, while still being 100% built by you.

DOMAIN NAMES

If possible, you should buy a domain for your site, *yourname.com*. There are many domain registrars you can use, and registrations typically cost ~$10 USD per year. I primarily use Google Domains, I've had good experiences with them.

What if yourname.com is already registered by someone else? Here are some suggestions:

- Add a middle initial—my site is Josh**W**Comeau.com
- Use a different TLD (top-level domain, eg. *.com*):
    - *.co* and *.io* are decent generic choices
    - Use one specific to your country (eg. *.ca* in Canada)
    - You can use *.codes* or *.dev* for developer-specific options
- Add a suffix:
    - morgancodes.com
    - dariusdev.com
    - Priyathedeveloper.com

You should avoid the *.info* TLD — it's frequently associated with scams and spam. Also, avoid using a screen-name for the domain (eg. no *laserdude9000.com*), unless the screen-name is reasonably professional and a strong part of your identity.

HOSTING

Your portfolio site should probably be a static site, which means there's no need for a backend server. There are many amazing services you can use for hosting a static site, including:

- Vercel (https://www.vercel.com/)
- Netlify (https://www.netlify.com/)
- Github Pages (https://pages.github.com/)
- Surge (http://surge.sh/)

All of these services support custom domains. Research for your specific service to see how to connect them.

You shouldn't need to do anything with services like Heroku or DigitalOcean, nor should you need to pay for monthly hosting with companies like GoDaddy.

PERSONALIZING YOUR PORTFOLIO

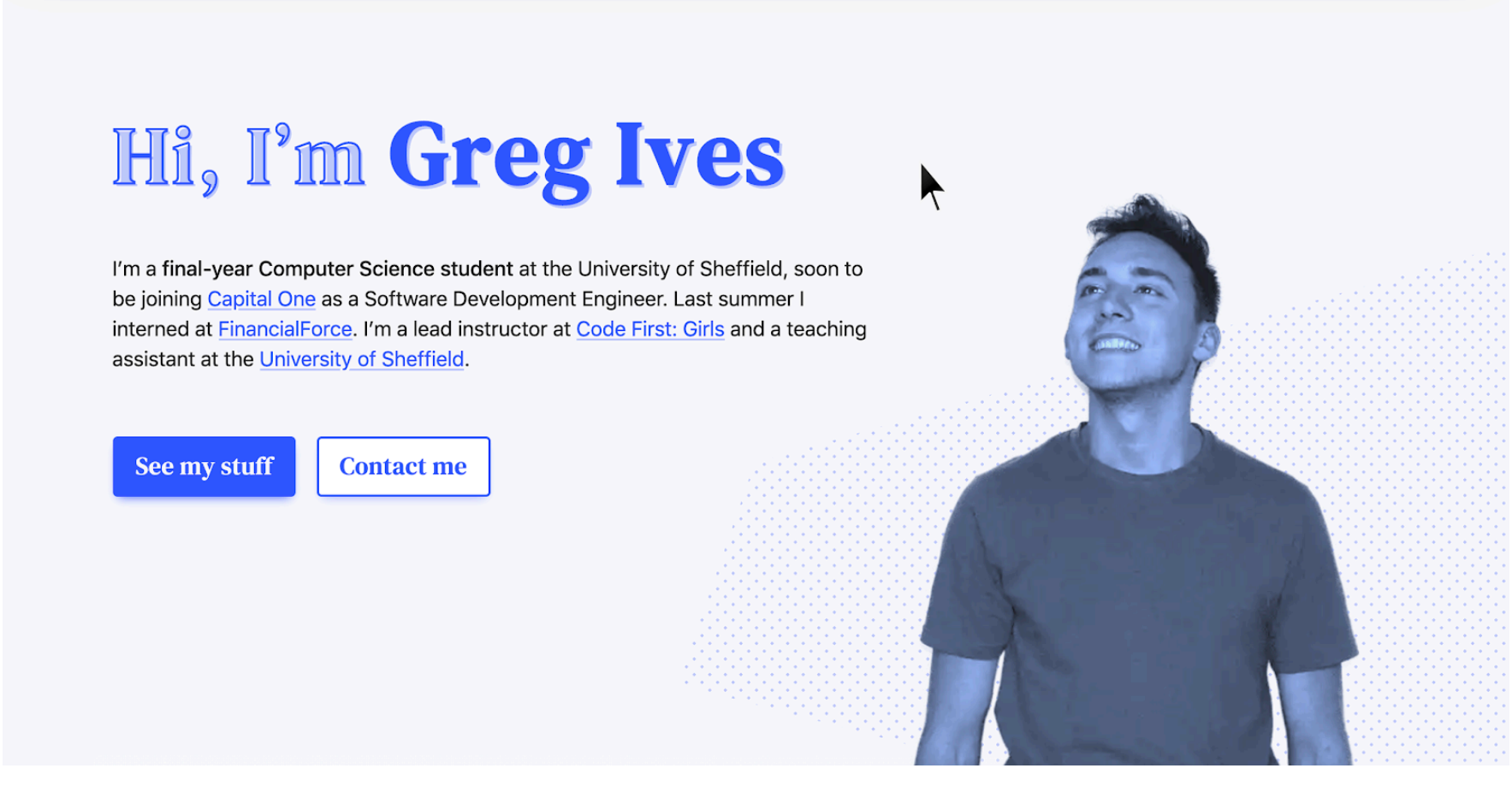
Even though I recommend picking a portfolio template as a starting point, please don't feel like you can't modify it! In fact, you'll likely need to make some modifications; portfolio templates are rarely set up exactly perfectly.

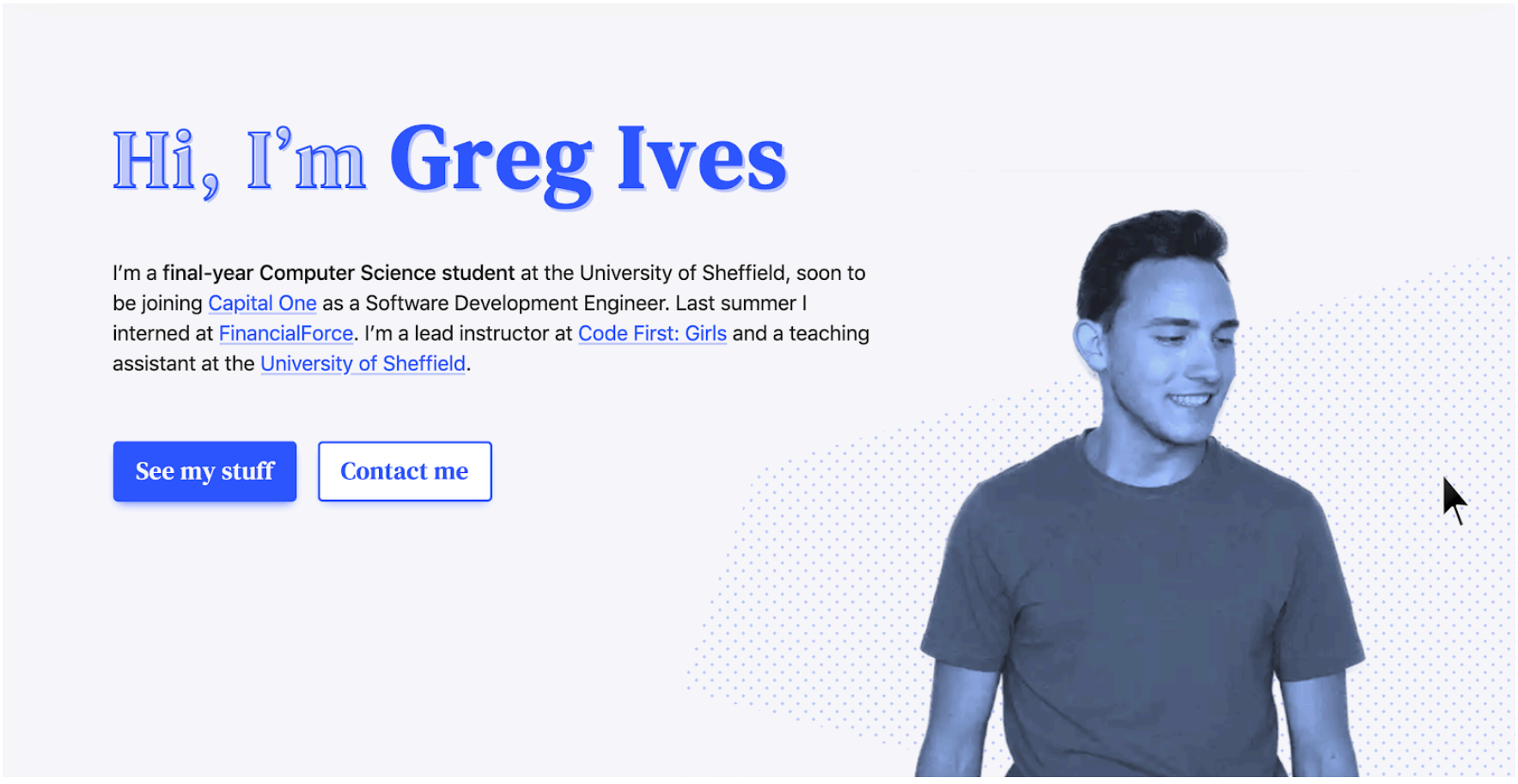
To be clear: our biggest lever for customizing our portfolio will be in the *copy* - how we describe ourselves and our projects. We don't need to make extensive UI customizations.

That said, it can be nice to add some sprinkles, to showcase our enthusiasm for UI development and inject a little whimsy and personality. Some ideas:

- Create a pixelated animated flag for your home country (or LGBTQ+ identity, if applicable)
- Add a dark/light toggle with micro-interactions
- Add a hover animation on interactive elements (links, buttons)
- Add an interaction where it isn't expected

One of my favourite examples of this is from Greg Ives (https://gregives.co.uk/). Like many portfolios, it features a photo of himself on the homepage. But as you move the cursor around the page, the photo updates, creating the illusion that he's watching the cursor move:





Cursor tracking is an advanced technique, and this is an exceptional example of a flourish; please don't feel like you have to compete with this. Small, creative details that reflect your personality can go a long way towards making your portfolio stand out and be memorable.

ACCESSIBILITY

It's important that our portfolio site conforms to accessibility standards. Our website should be usable by folks who aren't able to use a mouse and keyboard, as well as blind or colorblind people.

Accessibility is a broad topic, and it's outside of the scope of this book. Here are some resources to help, if you're not already familiar:

https://a11y.coffee/

https://webaim.org/resources/

https://websitesetup.org/html-tutorial-beginners/

You may wonder if this is really a valuable use of time. After all, it's unlikely that the hiring managers or developers will themselves have a disability, right? Do most candidates really submit fully-accessible developer portfolios?

In fact, most candidates don't; many of the junior-developer portfolios I've seen have been fairly inaccessible. But that's all the more reason to excel at this!

Even if the folks reviewing your portfolio don't have a disability themselves, they may know how to test a site to see if it's accessible. This is increasingly becoming a concern to companies, especially in the US, where the Americans with Disabilities Act (ADA) ensures that companies cannot discriminate against people with disabilities. The Supreme Court recently decided, by denying an appeal, that this law applies to websites. Companies are suddenly much more inclined to prioritize developers with accessibility experience.

More information: https://www.cnbc.com/2019/10/07/dominos-supreme-court.html

Additionally, it isn't fair to assume that no disabled users will try and access your website; it is estimated that 26% of the United States has at least 1 disability right now, and many will experience a disability, either temporary or permanent, at some point in their lives. All too often, developers neglect this segment of the population, though times are changing.

GENERAL LAYOUT

The design you choose to model your site after will affect the layout, but let's talk about some "must-have" elements. If your template is missing one of these elements, I *thoroughly* recommend adding it.

Your home page should include:

- A brief "About Me" section that highlights your background and personality.
- A list or grid of projects you've worked on.
- Contact information (ideally an embedded form, but a mailto link also works. Don't just include links to social media)

If you're comfortable with the idea, it can also help to add a photo of yourself in the top part of the homepage.

Each project in the list/grid should link to a "Project Details" page, which describes the project in greater depth. This is absolutely critical.

THE "PROJECT DETAIL" PAGE:

Adding "project detail" pages is pretty uncommon among developer portfolios, but it's super common among design portfolios.

The design community has realized that screenshots alone are insufficient. They need to tell a story about their work, to capture all the stuff not shown in the images.

> *This need is even greater for developers. The work that we do is all about the implementation, not the aesthetics! A screenshot tells very little about our role on the project. And the source code is far too granular, too low-level.*
>
> *If you only take one thing from this book, make it this: no matter how cool your project is, it cannot be summarized by a screenshot and a short paragraph. Guide the reader through your work!*

You might struggle with how to describe this project – filling a page about a project can be a very tall order!

Many junior developers *undervalue* their own projects. It's likely the projects you've built are more impressive than you think! Part of this comes down to marketing.

> *Years ago, I developed a bit of a reputation amongst friends and family for being good at helping people with their resumes, because I was able to frame their work experience in a positive light. Consider the following excerpt from a resume:*
>
> **Costco, retail sales, 2018-2020**
>
> - Sales associate
> - worked on the store floor, Electronics section.

This is an accurate but underwhelming description of the person's experience. I might rephrase as:

> **Costco, retail sales, 2018-2020**
>
> - Developed deep expertise in consumer electronics
> - Guided customers towards relevant purchases based on their needs, as well as any relevant promotions and sales
> - Operated modern POS systems to check customers out

- Maintained a consistently high approval rating from quarterly guest feedback review cards.

This description is also accurate (assuming that this hypothetical person actually did all of these things), but it sounds much more impressive.

We have a similar goal with our portfolio. We want to be completely honest, but we also want to highlight all of the details that make our project cool. Things that might seem trivial or mundane to you can actually be very impressive to potential employers!

To help get you started, here is a template for what the sections should be, and which information each section should hold.

The exact sections will depend on the nature of your project. Feel free to deviate if it doesn't quite fit your experience.

Introduction

- High-level summary of what the project is
- List of core functionalities / interesting features
- Your role in the project. were you exclusively doing development, or did you do design? If you worked in groups, what parts did you tackle?
- Technologies used
- Links to live demo + source code (if applicable)

Purpose and Goal

- Why did you build this project? Why is it important to you?
- What was the expected outcome of the project?
- What were the initial designs?
- Any other preliminary planning that you did which helps build a narrative

Spotlight

- What is the “killer feature” of your project? What feature does it have that took the most work, or was the most technically impressive? Some possible examples:
    - User authentication
    - A feed of items fetched from a database
    - A particularly tricky UI element (eg. autocomplete, calendar, drag-and-drop)
    - Anything you’re proud of!
- What were the technical hurdles that got in your way? Any major problems you hit during development?
- How did you solve those problems? What was the solution? Go deep here, and write with a developer in mind.

Current status

- This section is optional. *If* the project is actively being used by real people, talk a little bit about the current status, who uses it, why they use it, what they say to you about it, stuff like that.
- If the project was contrived specifically for the portfolio, omit this section.

Lessons Learned

- What did you learn doing this project? Feel free to list multiple things. Also feel free to cover non-technical lessons. It’s great to talk about how you learned to use an advanced feature of a framework or library, but it’s just as valuable to talk about project-management experience, or things you learned about shipping projects.
- If you used a framework or other libraries/tools, was it a good choice? How did it help? In which ways was it insufficient?
- Is your project accessible? What did you learn about accessibility, while building this project? Describing how you tested your project using keyboard navigation or a screen-reader can make for a really compelling story!
- How has this affected the work you’ve done since then? Real examples of how this project built your knowledge for future projects is fantastic.

If you have screenshots of your project (or photos of the process, eg early sketches), you should sprinkle them around this page. Weave them into the narrative, use them to highlight the things you're talking about in the text.

As I mentioned earlier, this template can vary widely depending on your specific project, but this is the level of detail you should aim for. It should take up 2-3 pages when printed (more if you have lots of images).

This is a lot of information, and people often won't spend that long reading. Because of that, **you need to be strategic**: avoid long paragraphs. Make use of many headings, and use lists within those headings to break up chunks of text. In essence, make it **easy to skim**. Some people will get drawn into our narrative and read every word, but we want to make sure the core messages come across even if someone browses quickly.

> ***Video demos?***
>
> *A trend I've seen occasionally is that developers will include a video walkthrough, showing off all the main features of the project.*
>
> *This is overall a good idea; it shows that the app is functional, while making sure that a "happy path" is followed. You don't need to worry about the user wandering off and finding a bug, or using a browser that isn't fully supported!*
>
> *It's not sufficient on its own though. A video will show what the app does, but not things like why you chose to build it, what technical problems you ran into, etc. Plus, videos are harder to skim than text; for that reason alone, many folks will skip watching videos.*
>
> *If you're particularly proud of specific functionality on your project, I suggest adding a video near the top, but repeating a lot of the information in text below, for folks who skip it.*

TONE AND COPY

Finding the right tone for the text you write can be tricky. The most important thing is that you want it to sound natural.

The biggest mistake I see in terms of tone is that developers adopt a “professional” voice that comes across as stuffy and corporate. I tried to model this voice in my “generic portfolio” example.

Why is this such a mistake? Think back to that generic portfolio; even though I asked you to review it carefully, do you remember any of the words from that site?

Formal corporate-speak is incredibly forgettable. It sounds more like an automated telephone robot than a real human. People have learned to tune that tone out, since it doesn't usually carry important information.

Here’s your goal: after you’ve written some text, read out loud, ideally to another person. Does it sound like something you’d say in real life? Rewrite it until it’s conversational.

This should be done within reason; if you curse a lot in your day-to-day life, maybe omit that part. But I’ve almost never seen a developer write *too* casually.

A note on privilege

The copy I write, whether on a portfolio site or a cover letter, tends to be very casual. It has worked well for me, but it would be negligent not to acknowledge the fact that as a white guy, it's likely that my professionalism is assumed. I don't have to prove it in how I write.

In my work as a career coach, I've worked with women and people of color, and I've always been cautious not to assume what worked for me would work for them. Unfortunately, tech is not a meritocracy.

I don’t believe that formal corporate-speak is helpful for anyone, but I also recognize that people might want to find a middle ground, and opt for copy that's a little less casual.

For folks who struggle with English.

If English isn’t your first language, or if you struggle with grammar and vocabulary, this can be an especially difficult challenge. I would suggest asking for proofreads from friends or peers. If you’re not able to find a proofreader,

> *feel free to shoot me your portfolio site on Twitter (@joshwcomeau). I can't make any promises about my future availability, but I'll do my best to either proofread it myself, or else help you find a proofreader.*

DEVELOPER BLOGS

Some portfolio sites feature developer blogs, where developers share tips about what they've learned.

I think that developer blogs are awesome, and I'm a big fan of the idea of learning in public (https://www.swyx.io/writing/learn-in-public/). But I'm not sure they should be combined.

As we've discussed, a developer portfolio is built to showcase your best work, and to do it in a very short amount of time. The folks who are looking at your portfolio after you've applied to work at their company likely don't have the time to read any of your blog posts.

I suspect the best strategy is to create an entirely separate blog for yourself (either on a platform like Dev (https://dev.to/), or doing it yourself with something like Ghost (https://ghost.org/) or Gatsby (https://www.gatsbyjs.org/)).

It's unfortunately true that most developer blogs only have one post: "How I built my developer blog". If you make it to your 5th or 6th blog post, you've done something truly uncommon! At that point, I think it'd be a great project to describe in your portfolio.

IN SUMMARY

- Come up with a professional-looking design by copying existing templates. By mixing and matching, we avoid plagiarizing. Whatever you do, don't fork another developer's portfolio and slap your name on it.
- Pick the tools you're already comfortable with to build your site.
- If possible, buy a domain name. You can host your site on free services like Netlify or Vercel.
- Add small flourishes that personalize your site.
- Prioritize accessibility. It's an essential piece of the frontend skillset, and one that employers are increasingly seeking out.
- Your homepage layout should consist of an About section, a Project Details list or grid, and a contact form (or contact details).
- Each project should link to a page that describes it in depth. Use the provided outline if you're not sure what to include.
- If you're interested in creating a developer blog, it should separated into its own thing. You may wish to list it as a project!

CHAPTER V:
PROMOTING YOUR PORTFOLIO

As we've already discussed, not all employers will look at portfolio sites. This chapter include some tips and tricks we can use to increase the number of employers who will see it.

As part of a formal application

Many application forms will ask you for your website. This should absolutely be your portfolio site!

Additionally, there is often a blank text box near the bottom, where applicants can include any information they think is relevant. This is a great place to encourage reviewers to check out your portfolio. You can mention that it includes details on the projects you've built.

Applications will ask you to attach your resume. You can include a link to your portfolio within your resume; I like to include mine by my contact info at the top.

We don't want to overdo it; some employers will never see it, and that’s ok. Our goal is to make sure that it's never hard to find, for the employers that are interested.

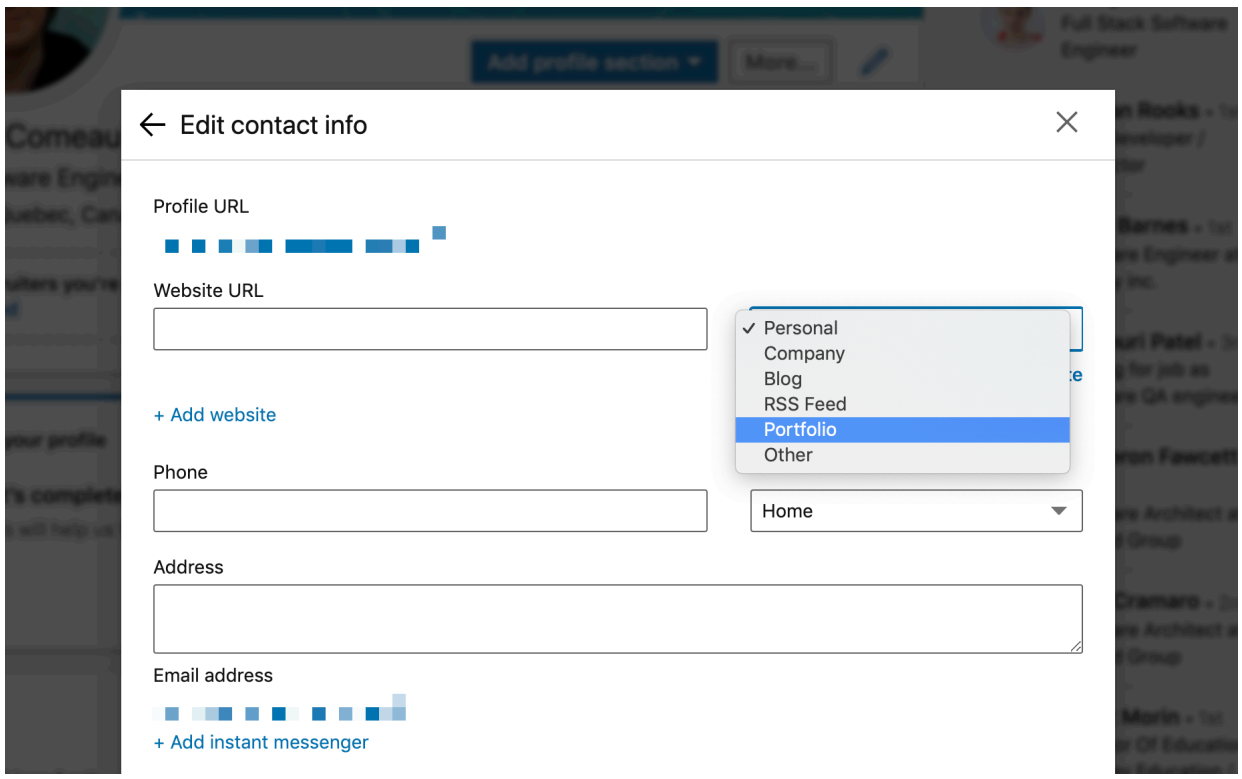
The further you get in your career, the easier it becomes to find work. The early jobs are the hardest ones, because you’re competing with so many other aspiring devs for a relatively small number of junior job postings. Having a huge advantage amongst even 10-20% of employers is potentially life-changing. And I suspect far more than 20% of employers will at least glance at your site.

As part of a networking effort

It’s estimated that between 70-80% of jobs are never advertised on job boards[†]. Networking unlocks a larger pool of possible jobs. This is often unwelcome news, especially for introverts like me. The silver lining is that it's much easier to show off our stellar portfolio site when we're meeting people through networking.

Networking is far beyond the scope of this book, but here are some quick ways to incorporate your portfolio into these efforts:

- Pin your portfolio site on your Github profile.
- Add your portfolio site to your social media profiles, especially LinkedIn:



- Add a link to your portfolio in your email footer. Every correspondence is a chance for the recipient to discover it.
- Put a link on your business card. I am writing this during the coronavirus pandemic, so this one doesn't feel particularly relevant right now, but once local events come back, this could be worthwhile.
- Work it into conversations, when appropriate. When speaking about your projects, you can say that a bunch more information can be found on your site.

[†] I found a few sources for this, like this Forbes article (https://www.forbes.com/sites/jacquelynsmith/2013/04/17/7-things-you-probably-didnt-know-about-your-job-search/#737b18ad3811) and this LinkedIn study (https://www.linkedin.com/pulse/new-survey-reveals-85-all-jobs-filled-via-networking-lou-adler/). Neither of them strike me as undeniably credible. Even if the exact number isn't quite right, the fact remains that there are lots of jobs that go unlisted.

CHAPTER VI: COVER LETTERS

Cover letters are entirely outside the scope of this book, but I'm squeezing it in anyway, because I think it's super important.

Many of the things we've discussed about portfolio sites apply to cover letters. In particular:

- Be conversational. Avoid the empty-calorie "corporate speak".
- Tell a compelling story. Your cover letter should be unique to you. It should feel dishonest if someone else was to copy your letter and replace your name with theirs.
- Keep the HR hiring managers in mind (and to a lesser extent, developers). You're targeting the same demographics.

There's one other cover-letter-specific thing which is super important: **it should be unique for the company you're applying to!**

Far too often, I've seen people copy/paste the same cover letter for each company they apply to, swapping out 1-2 details like the company name and the sector ("I am interested in the _____ industry").

It's OK to copy/paste the parts of the letter that are about your background, but most of the cover letter should be about how your skills and interests overlap with the company's. You should be talking about why you think you'd be a perfect fit for the role. Which specific attributes of the company align with your specific strengths.

As a rule of thumb, 75% of a cover letter should be unique to each company. Don't copy/paste more than 25%.

If you're applying to an agency or a company that doesn't have a specific product, this can be a bit harder, since there isn't an industry you can identify with. But you can learn a lot about a company's culture from their website (and any posted company values), and use that in your letter.

What if you genuinely don't have anything in common with the company you're applying to? Earlier in my career I considered applying to a sports stats site. I don't know anything about sports. This is clearly not ideal, but sometimes we have to settle for unideal companies early in our careers.

In these cases, you can either try to find a tangential interest, or else apply purely based on the culture. You could bend the truth a bit, and pretend to have an overlapping interest, but this is risky; what if someone starts talking to you in the interview about sports?! I'd consider this a last resort.

In Appendix B, I've included a few cover letters I've used which have led to an offer. As with portfolio sites, your cover letter should be unique to you, so please don't plagiarize my letters. But hopefully they can inspire you to write in a similar fashion.

CONCLUSION

I hope this book has helped give you some direction when it comes to your portfolio site!

This book doesn't include any complete examples (aside from the case study), because every person's portfolio site should be unique to them. Remember, the goal is to stand out, not to do what everybody else is doing!

The best portfolio sites are the ones that are able to tell a story about the person and their projects, in a way that makes you feel like you're getting to know the person.

I look forward to seeing what you create—feel free to let me know how it's going on twitter! You can reach me at @joshwcomeau.

A NOTE ON SHARING

If you'd like to share this book with other people, I kindly ask that you direct them to this website: https://joshwcomeau.com/effective-portfolio/

This provides valuable analytics about how many people are reading the book, and gives me a way to reach out to readers directly.

This book is and always will be free to download ✨

APPENDIX I: PORTFOLIO TEMPLATES

While I suggest building your own site from scratch, it can be interesting to take inspiration from the templates found on site builders.

We won't take any of the code from these templates, and ideally we won't copy too much from any single one. These are sources of inspiration.

All of these portfolios were meant for designers, not developers. I spent quite a bit of time trying to find developer portfolio templates, and none of them were suitable. Happily, many designer portfolios can be easily repurposed for developers.

This is definitely not a definitive list! Feel free to find inspiration in other places.

- Portfolio Starter — https://portfolio-starter-template.webflow.io/

  This is a great-looking minimal portfolio, with all the elements we need. Like many of the themes in this appendix, it's targeted at designers, but it works very well for our needs.

- Craig Portfolio — https://craig-roush-portfolio-template.webflow.io/

  This is a colourful, off-beat theme. I really like it, but it might not be to your taste. It is very recognizable, so you probably don't want to take *too* much inspiration from it.

  It has some elements that are more suitable for a freelancer: don't include the testimonials or the "companies worked with" bits.

- Alex Portfolio — https://alex-grant-template.webflow.io/

  This portfolio has an Apple vibe to it. In terms of layout, it's not great, because it buries the projects. But certain elements are quite nice.

- Dexter Portfolio — https://dexter-morgan.webflow.io/

  This portfolio is *very* designery. It probably won't be the right fit for most folks, but it might fit with your personality.

- Novo — https://novo-demo.squarespace.com/?nochrome=true

  This portfolio is very minimal and clean. it's image-heavy, which could be a problem if you don't have a lot of high quality images.

- Kester — https://kester-demo.squarespace.com/?nochrome=true

  Squarespace has a lot of portfolio templates, and they all kind of look and feel like this. It's nice and minimal, but image-heavy.

- "Art Director" — https://www.wix.com/website-template/view/html/1913

  This portfolio, like the previous one, depends on having lots of images for your projects. It's probably not the best portfolio template for most folks.

APPENDIX II:
COVER LETTERS

This section features cover letters that I've personally used in my applications, and which led to a job offer. I'm including them as an example of the kind of to that can stand out from a crowd, though it’s important to point out that my experience is not universal. Adopting a similar tone will not necessarily lead to more job offers for you.

I’m including them to show how personalized they are for the company.

LETTER 1: KHAN ACADEMY

Khan Academy is a non-profit organization dedicated to providing a free, world-class education for anyone, anywhere. Here's what I wrote:

Several years ago, I wanted to build my first web application. I had been doing light web development work (landing pages, contact forms) for a while, and wanted to see if I could build something more interactive.

I had a friend who was working as a tutor, teaching high-school math, and he wanted a way to quiz students online. I've always been passionate about education, and it seemed like an awesome project.

The MVP was very bare-bones, and it was held together with duct tape, but it worked. There was a simple auth system, students would complete quizzes, they would be graded and the results would be emailed to my friend.

The product only worked when you had a tutor available to guide you through the material, though. I wanted to build something that would help students who didn't have access to a tutor. I was exploring the idea of building it into a generalized product with built-in lessons when I discovered Khan Academy.

Khan Academy was almost exactly what I had wanted to build! I went through a few of the lessons, and was thoroughly impressed by the quality of the instructions (Sal Khan has such a talent for teaching) and the clean interface for the practice problems.

Ultimately I realized that I wouldn't be able to build anything that offered a better experience, and have since been building other web applications, but I never stopped being passionate about education, or wanting to build tools to help people learn.

I'd love to chat about why I think I'd be a great addition to the Khan Academy team :)

-Josh

LETTER 2: GLITCH

Glitches an online code platform designed for education and collaboration. Here's my letter:

Hi Glitch!

My name's Josh, and I'm a senior software developer. I currently work at Khan Academy, but I'm super interested in working at Glitch.

I've been building web things for almost as long as I can remember - I think my first website was a collection of my favourite GIFs when I was 11 or 12, back in the "Welcome to my homepage, it's under construction, sign my guestbook" days. It was rudimentary, but then so was everything else; it did about as much as every other website.

Of course, things have changed a lot since then. I've had the luxury of adapting with the internet, but newcomers nowadays have a much steeper hill to climb. I work part-time at a local coding bootcamp, and it's made me realize just how much stuff you need to know to be able to build something that fits in with today's web.

I think this is a necessary trade-off, because while things are more complicated, the stuff we build is so much more powerful. But I love the idea of making it more approachable. I'm really excited about the work Glitch is doing to enable folks to learn how to build, from the ability to "View Source" across the stack, to the community aspects like being able to request help from others when stuck on a bug, to making source control simple and enabled by default.

I think I'd be a great candidate to help build the community site for a few reasons:

- *I have strong technical skills with Javascript and React. Khan Academy is one of the oldest and largest production React applications, and I worked on a project to convert it to a modern single-page-app architecture. I've created and maintained several popular open-source React projects, such as a GUI project manager (Guppy) and an animation library (react-flip-move). I've spoken at several React conferences.*
- *I'm passionate about design and UX. I really like building whimsical, delightful experiences, and I believe that it's a huge boost to productivity when you can work closely with design and leverage your own intuition. I do all the design work on my side projects, and it's a ton of fun.*
- *I love building quirky, weird, experimental projects. For example, I'm currently working on an app that allows folks to create their own generative art. This gives me empathy for fellow tinkerers, and I'd derive a lot of satisfaction from knowing that my work at Glitch helps folks build similar things.*

Looking forward to hearing from you,

-Josh

LETTER 3: DIGITALOCEAN

DigitalOcean is a cloud computing platform developers can use to host their applications.

> *Hi there!*
>
> *My name is Josh, and I'm a senior front-end developer. I currently work for Khan Academy, but I'm super interested in working for DigitalOcean.*
>
> *I've been a DigitalOcean customer for many years, and I've really enjoyed how accessible it makes server provisioning and administration. As a mostly-front-end developer, DevOps is pretty far outside my wheelhouse, so it's been tremendously useful for me! Empathy is so important for building great products, and I have a built-in empathy for enabling folks to build products they believe in without getting drowned by the complexities of cloud computing.*
>
> *I think I'd be a great fit for the role for a few reasons:*
>
> - *I have strong technical skills with Javascript, HTML and CSS. I've been building things for the web in one form or another for over a decade (back when layouts were done with HTML frames and tables), and I've kept up with the industry. For the past few years, I've worked primarily with React.js. I championed the switch from Backbone to React at a previous job, have made significant architectural changes with React at my current job, and have produced a lot of tools for the open-source community, like an animation package and a GUI project manager.*
> - *I care about the details. I really like working closely with product and design to create amazing, polished user experiences, and will often go the extra mile to include whimsical touches. DigitalOcean's Love is what makes us great value really resonates with me.*

- *I'm passionate about learning! I really liked that the job posting emphasized that DigitalOcean believes in a growth mindset. I'm excited by new challenges and the opportunity to develop further.*

Looking forward to hearing from you,

-Josh

PAPER NAME

ACFrOgDyHSJMZZAs9FB8YZjBrKXFactS5lArTG8B4hzY2m7GY7wbr_yNz09K8qGQKt9dCLEj46BSObZ1KEETplZiBfsH6F6KOb19

AUTHOR

a b

WORD COUNT

5132 Words

CHARACTER COUNT

28416 Characters

PAGE COUNT

6 Pages

FILE SIZE

152.9KB

SUBMISSION DATE

Dec 2, 2022 2:58 PM GMT+5:30

REPORT DATE

Dec 2, 2022 2:59 PM GMT+5:30

45% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 37% Internet database
- 42% Publications database
- Crossref database
- Crossref Posted Content database

Driver Drowsiness Detection System using Visual Features and Machine Learning: A systematic literature review

Dr. Prajakta Khadkikar
Computer Engineering
Pune Institute of Computer Technology
Pune, India
pakhadkikar@pict.edu

Miss. Rutuja Banginwar
Computer Engineering
Pune Institute of Computer Technology
Pune, India
rutujabanginwar01@gmail.com

Mr. Imon Mandal
Computer Engineering
Pune Institute of Computer Technology
Pune, India
imonashokmandal@gmail.com

Miss. Urmila Matade
Computer Engineering
Pune Institute of Computer Technology
Pune, India
matadeurmila1712@gmail.com

Mr. Mayur Mote
Computer Engineering
Pune Institute of Computer Technology
Pune, India
mayurmote8055@gmail.com

***Abstract*—Over the past ten years, the number of motor vehicles has steadily grown in emerging nations. The reports of traffic accidents note that risky driving behaviors, such as drunken driving or fatigued driving, are responsible for most accidents. According to studies, sleepy driving is a contributing factor in 20% of all accidents. Drowsiness is a condition when the level of consciousness is lowered as a result of fatigue or lack of sleep, and it can make a driver fall asleep silently. Drowsy driving causes the driver to lose control of the vehicle, causing it to drift off the road, collide with an obstacle, or overturn. In this study, we carried out a Systematic Literature Review (SLR) to extract and synthesize the algorithms and features that have been used in drowsiness detection studies. Based on our search criteria, we retrieved 100 relevant studies, of which we have selected 30 studies for further analysis. We investigated these selected studies carefully, analyzed the techniques and features used, and found out the research gap as well. According to our analysis, the most used features are facial expressions like yawning, closing of the eyes, and head motions. After this observation based on the analysis of 30 papers, widely used techniques are Eye Aspect Ratio, Haar Cascade, Support Vector Machine(SVM), dlib library.**

***Index Terms*—feature extraction, visual features, machine learning , computer vision, drowsiness detection.**

I. Introduction

Drowsiness is one of the main factors that threaten road safety and cause serious injuries, deaths and economic losses. Lack of alertness, caused by the unconscious transition from wakefulness to sleep, leads to several serious traffic accidents. Drowsiness and distraction are two areas of study that researchers focused on when designing systems to monitor driver inattention. Systems designed for sleepiness analysis and detection can generally be divided into two categories: visual feature-based and non-visual feature-based. Techniques using visual features use computer vision approaches to detect drowsiness. The use of visual features focuses on the extraction of facial features such as the face, eyes and mouth. Analysis of eye and mouth conditions can provide observable cues for the detection process. Techniques using visual features can mainly be divided into four categories: eye state analysis, blink analysis, mouth and yawn analysis, and facial expression analysis. Image processing and machine learning techniques are the two main steps for visual feature detection and processing. Non-visual feature-based techniques are generally intrusive and fall into two categories: driver physiological analysis and vehicle parameter analysis. For example, in EEG technology, several sensors must be attached to the driver while driving. The acquisition system, processing system, and warning system are the three components or modules that make up the fundamental sleepiness detection system. Here, the video of the driver's front face is captured in the acquisition system and transferred to the processing block where it is processed online to detect drowsiness. A warning or alert is delivered to the driver by the warning system if drowsiness is discovered.

The rest of this document is organized as follows: Section 2 explains the background. Section 3 discusses the methodology. Section 4 discusses the various drowsiness detection techniques. Section 5 presents the results of SLR and Section 6 concludes this document.

II. Related Work

With a total road length of approximately 62.1 lakh kilometers, India has the world's second largest road network. This vast network serves as the nation's lifeline, transporting over 64.5% of all goods within the country and being the preferred mode of transportation for more than 90% of India's passenger traffic. Lack of consciousness while driving due to sleepiness is the root cause of all road accidents. If a drowsy driver is

informed in a timely manner, then a lot of road accidents can be avoided and many lives can be saved. There are different techniques and algorithms to detect drowsiness. In this review article, the research work that has already been published on detecting drowsiness has been investigated.

While searching for the research paper, we excluded the research works that were very old and only considered the latest research. Shruti Mohanty and et.al has published a paper to detect drowsiness using opencv, python, and dlib to calculate the Euclidean aspect ratio of eyes and mouth to detect drowsiness[3].

In 2020, Jagendra Singh proposed the paper on Learning based Driver Drowsiness Detection Model, In that the EAR is calculated for the eyes using the 6 landmarks around the eye ROI. A threshold is set for 0.3 and if the EAR becomes less than the threshold for at least 20 frames, an alert is generated to the driver predicting that they have drowsy eyes. If the landmarks around the eyes ROI and mouth ROI cannot be determined for at least 15 frames, an alert is generated to the driver insisting the driver pay attention to the road[8].

In 2020, Amin Azizi Suhaiman and et.al proposed the paper on "Development of an intelligent drowsiness detection system for drivers using image processing technique, in that they utilizes computerized camera to automatically track and process driver's eye using Python, dlib and OpenCV, as Image Processing depends on intensity of light thus affects the results of detection[5]. R. N. Khushaba and et.al proposed a paper on "Driver drowsiness classification using fuzzy wavelet-packet-based feature extraction algorithms in 2011. For that they have developed a fuzzy mutual information based wavelet packet transform model to estimate the drowsiness level from a set of electroencephalogram, electrooculogram, and electrocardiogram signals, as physiological approaches often require electrodes to be placed on the driver's body, which is inherently invasive and can be cumbersome for the driver[29].

The paper titled "Real-time detection of driver cognitive distraction using support vector machines" was proposed by Y. Liang and et.al, here they have used real time approach for detecting distraction using the driver's eye movements and driving performance data collected in a simulator environment called the in-vehicle information system. After that, for detecting the driver distraction the data were used to train and test both support vector machine (SVM) and logistic regression models. Thus it highly depends on the vehicle type, driver experience, and the road condition.Such systems can fail because the car doesn't provide important information if the driver falls asleep on a straight road[30]. Li, Gang and et.al proposed a paper on "Smartwatch Based Wearable EEG System for Driver Drowsiness Detection in 2015 where the smart watch and a headband containing sensors is used to identify drowsiness. The driver drowsiness level could be transformed to any probability value of 0 1 instead of discrete class labels aimed at detecting drowsiness as early as possible. This is limited by the driver's choice who may not want to wear a headband and may want to wear a different watch[27].

In 2009, Li, Lingling and et.al proposed paper on "Yawning detection for monitoring driver fatigue based on two cameras, in it they monitors yawning patterns of the driver using two separate cameras to acquire information of the upper part of the body in order to track the driver's mouth, here limitation is that hardware dependency is more[28].

A. B. Albu and et.al proposed a paper "A computer vision-based system for real-time detection of sleep onset in fatigued drivers" in 2008. After reading this paper we understand that real-time performance was achieved by focusing on one single visual cue (i.e. eye condition) and custom designed template-matching algorithm for online eye condition detection. Also additionally designed a computer vision-based sleep monitoring technique will be implemented through customer-designed DSPs or FPGAs built into the vehicle[11].

The study used a vehicle driving simulation system that was suggested in the 2010 work "An EEG-based technique for identifying drowsy driving conditions" by M. Liand et.al. EEG data from sober and drowsy drivers were obtained. FastICA arithmetic was used to remove EMG and EOG interference EEG. Fatigue index F from EEG exceeded equation can indicate fatigue management characteristics. The driving fatigue F value was significantly higher than normal driving, so it can be established to assess and evaluate degree of fatigue[12].

A study conducted in the paper "Detecting Driver Yawning in Successive Images" published in 2007 by Lu Yufeng and Wang Zengcai concludes that based on the location of the center of the nostrils and chin, we can detect driver yawning by calculating the vertical distance between them. They said the method was robust to variant lighting because the method was not based on color or intensity, which is fragile for lighting[13]. In 2011, S. Abtahi and et.al suggested in a study that it would be possible to choose a complex algorithm that would be one step closer to the actual implementation of the system under "Yawn Detection-Based Driver Drowsiness Monitoring" than the proposed methodology[14].

Choi and et.al published a paper in 2016 , "Real-time categorization of driver's gaze zone using the deep learning techniques". It used deep learning and facial expression recognition. The proposed system achieved a mean detection accuracy of over 95%. This approach has the disadvantage that it requires a huge amount of data to train the neural network to work with a high level of accuracy[26].

Fouzia and et.al presented the hardware-based driver drowsiness detection system based on visual features. This system makes use of the Raspberry Pi and the Python OpenCV library for drowsiness detection based on the blink count. The proposed model continuously monitors the driver's eye movement and warns them by turning on the vibrator when they are drowsy. A vibrator signal is generated to alert the driver if the eyes are detected to have been closed for a long period of time [1]. One of the main contributing factors to traffic accidents is drowsiness. To avoid this issue, Ashish and Rusha proposed a driver drowsiness detection system based on mouth and yawning analysis. First, facial landmarks on the detected face are identified, and then the eye aspect ratio, mouth opening ratio, and nose length ratio are computed to

detect drowsiness using adaptive thresholding based on these values[2]. The Drowsiness Detection System Using the Eye Aspect Ratio Technique was proposed by Sathasivam and et.al. The basic objective of the proposed method is to detect drowsiness in drivers and warn them by raising an alarm. This system uses a camera mounted in front of the driver to capture real-time video, then captured video frames are then used to detect drowsiness using the Eye Aspect Ratio (EAR) technique[24].

A study named "Eye Behaviour based Drowsiness Detection System" by Javed Ahmed, Jian-Ping Li and et.al was published in IEEE in 2015. The creation of a Drowsy Driver Detection System has made use of a non-intrusive computer vision-based concept. In order to specifically identify driver drowsiness, a system that uses the small camera and analyzes the driver's eyes while focusing directly on his or her face has been developed. When a driver's weariness is apparent, a warning sign is displayed to alert other road users. The procedure of locating the driver's eyes and determining whether they are open or closed is demonstrated in this paper. The system is able to find the margins of a face and determine where a person's eyes may be by using binary image data that was acquired. The proposed method ensures that the driver is dozing off if their eyelids are closed for five consecutive frames, at which point a warning signal is sent. The framework also functions in conditions with reasonable lighting and can detect when the eyes are hidden from view[19].

An improved algorithm for drowsiness detection for non-intrusive driving was published in 2018 by Ratnarup Dey and Joy Paulose in the International Journal of Applied Engineering Research. The paper discusses a car-integrated system that monitors the behaviour of the driver based on a variety of stimuli. For the purpose of detecting head movement, steering gripping, and drunk driving, sensor characteristics are employed. As a part of this work, a system that can detect driver drowsiness by evaluating any one, two, or all three of the following parameters head movement, alcohol influence, and steering grasping is proposed. When one, two, or all of these circumstances exist, it is assumed that the driver is drowsy[16].

III. Methodology

A. Review protocol

The initial step is to define the research questions. Relevant studies are chosen from the databases once the research questions are ready. Science Direct, Springer Link and Google Scholar were the databases used in this study. A set of exclusion and quality criteria were used to filter and evaluate the relevant research after they had been chosen. From the chosen studies, all relevant data are retrieved, and ultimately, the extracted data are synthesized to address the research objectives.

B. Research questions

The purpose of this literature review is to find out already published algorithms and techniques for drowsiness detection and evaluate their performance. To prepare this literature review following research questions have been defined.

RQ1:- Which technique has been used for drowsiness detection?

RQ2:- What algorithms have been used to identify drowsiness?

RQ3:- What are the different visual features that the system takes into account?

RQ4:- What is the research gap in the proposed algorithm?

C. Search strategy

An automated search handles the fundamental searching. The search's initial inputs were "machine learning", "drowsiness detection system" AND "driver". Three databases were searched in this operation. To find a wide range of papers, the search terms "driver" AND "drowsiness detection system" were used.

Following is a detailed explanation of the search terms for each database:

ResearchGate: The search term is ["machine learning" AND "driver drowsiness detection"] and ["driver drowsiness detection"]

Springer Link: The search term is ["drowsiness detection system" AND "yawning"] and ["drowsiness recognition" AND "facial features"]

Google Scholar: The search term is ["driver drowsiness detection" AND "eyes aspect ratio"] and ["image processing" AND "drowsiness detection" AND "machine learning"]

TABLE I
Search Stratergy

Database	# Of initially retrieved papers	# Of papers after exclusion criteria	Percentage of Papers (%)
ResearchGate	7	1	14
Springer	13	2	15
Google Scholar	80	27	34

IV. Drowsiness Detection Techniques

Depending on the technique being employed, the Driver Drowsiness detection system continuously observes the driver's physical behaviour, vehicle movement patterns, or environmental factors. Methods for detecting sleepiness are often divided into three categories.

1) Methods based on behavioural parameter data
2) Methods that are based on vehicle parameters
3) Methods using physiological parameters

A. *Behavioural Parameter-Based Techniques*

Computer vision algorithms and cameras were combined with behavioral-based strategies to extract behavioural information. The non-intrusive methods of sleepiness detection are behavioural parameters. The behavioural indicators used by these strategies to gauge driver weariness include eye closure ratio, eye blinking, head position, facial expressions, and yawning. Environmental elements, such as lighting, brightness, and road conditions, have an impact on the measurement's reliability and accuracy, which are issues with behavioural measures.

*1) **Eye Blink Detection Method**:* A real-time drowsiness detection system utilizing Dlib was proposed by Mohanty, Hegde, and Manikandan [3]. The Dlib model and Python are used to create a drowsy driver detection system in this study. The coordinates of the facial landmarks in the input video are mapped using Dlib's form detector, which is also used to track the aspect ratios of the eyes and mouth to look for signs of tiredness. For the eyes and mouth, respectively, predefined threshold values of 0.15 and 0.83 are used to compare the estimated aspect ratios with each. The average aspect ratio of the left and right eyes must be above the threshold in order to be considered drowsy. The driver is alerted if indicators of fatigue are noticed for an extended period of time. Testing films from a common public dataset as well as real-time video is done to evaluate the performance of the suggested system. The greatest recognition accuracy for video datasets for the proposed system was 96.71

*2) **Eye and Yawning Analysis**:* A real-time drowsiness detection system utilizing Dlib was proposed by Mohanty, Hegde, and Manikandan [3]. The Dlib model and Python are used to create a drowsy driver detection system in this study. The coordinates of the facial landmarks in the input video are mapped using Dlib's form detector, which is also used to track the aspect ratios of the eyes and mouth to look for signs of tiredness. For the eyes and mouth, respectively, predefined threshold values of 0.15 and 0.83 are used to compare the estimated aspect ratios with each. The average aspect ratio of the left and right eyes must be above the threshold in order to be considered drowsy. The driver is alerted if indicators of fatigue are noticed for an extended period of time. Testing films from a common public dataset as well as real-time video is done to evaluate the performance of the suggested system. The greatest recognition accuracy for video datasets for the proposed system was 96.71%.

*3) **Nose, Eye and Yawning Analysis Method**:* One of the main contributing factors to traffic accidents is drowsiness. Ashish and Rusha [2] suggested a driver sleepiness detection system based on mouth and yawning analysis to get over this problem. The eye aspect ratio, mouth opening ratio, and nose length ratio are computed to identify drowsiness using adaptive thresholding based on these values once facial landmarks on the observed face are first identified. To find the face in the frames, this technique employs a histogram of oriented gradients (HOG) and a linear support vector machine (SVM). Image normalisation and an ensemble of regression trees are used to estimate the landmark positions on the face. Fisher's linear discriminant analysis (FLDA) and Support Vector Machine (SVM) with linear kernel were then used for classification. The INVEDRIFAC dataset has been used to test the newly built algorithm. SVM and FLDA have been found to perform better than Bayesian classifiers. Both FLDA and SVM have a specificity of 1, while their respective sensitivity values are 0.896 and 0.956.

*4) **Mouth and Yawning Analysis**:* Saradadevi and Bajaj [4] suggested a mouth and yawning analysis-based driver fatigue detection system. The Viola-Jones classifier is used to first identify the driver's face from the input frames. After that, the mouth and yawning image datasets are trained using SVM. Finally, yawning and sleepiness are detected by classifying the mouth regions using SVM. A dataset of more than 100 typical films and 20 yawning photos is chosen for the experiment. The findings show that the proposed system performs better than the system based on geometric features.

B. Vehicular Parameter-Based Techniques

Vehicle attributes like frequent lane changes, variable vehicle speeds, steering wheel tilt, steering wheel grip force, etc. are used in vehicle parameter-based strategies to identify driver weariness. These precautions call for sensors on car parts like the steering wheel, accelerator, and brake pedals, among others. These sensors produce signals that are used to analyse driver sleepiness. Principal objective of these techniques is to analyse driving behaviours and spot a drop in driving performance brought on by fatigue and exhaustion. However, evaluating driver weariness based on vehicle movement has its limitations because measurement results are often influenced by outside variables like the geometry of the road and the weather.

*1) **Real Time Lane Detection System**:* In the modern era, traffic accidents are all too common, endangering both the lives of those on the road and their property. Road accidents can occur for a variety of causes, including: rash driving, inexperience, disobeying signs, jumping signals, etc. The Drivers' Drowsiness Detection system was suggested by Katyal et al. [32] as a solution to the problems. The system operates in two stages: first, it uses the hough transform to recognise lanes. Secondly, it looks for signs of tiredness in the drivers' eyes. The viola jones approach is used to detect faces first, followed by picture segmentation, otsu thresholding, and clever edge detection for eye detection. The data are used with the circle detection through transform method to find eyeballs and gauge level of exhaustion. Low lightning conditions won't affect its effectiveness. The results indicate that the suggested strategy is beneficial for drivers who take long journeys, drive at night, or drive after drinking.

*2) **Steering Wheel Angle For Real-Time Driving Conditions For DDT**:* Li et al. [31] suggested the online detection of Drowsiness Detection System to track the level of driver fatigue under actual driving conditions utilising steering wheel angles (SWA). The sensors mounted on the steering lever are where the SWA data is gathered. The system first extracts the approximate entropy (ApEn) features from fixed sliding windows on a time series of real-time steering wheel angles. Next, the system linearizes the ApEn features utilising the deviation of adaptive linear piecewise fitting approach. The technique then determines the warping distance between a series of sample data's linear characteristics. Finally, the system uses warping distance in accordance with the created decision classifier to determine the drivers' level of tiredness. The empirical analysis employs data gathered over 14.68 hours of actual driving time, evaluated on two levels of fatigue: drowsy and awake. Results indicate that the suggested system

is beneficial for preventing traffic accidents brought on by driver drowsiness and capable of operating online with an accuracy of 78.01

C. *Physiological Parameter-Based Techniques*

Based on the drivers' physiological data, such as heart rate, pulse rate, breathing rate, respiratory rate, and body temperature, the physiological parameters-based algorithms identify drowsiness. These biological indicators are more dependable and accurate at detecting tiredness because they focus on the physical state of the driver. Systems for detecting drowsiness based on physiological characteristics notice these modifications and warn the driver when he is getting close to falling asleep. Electrodes must be positioned on the drivers' bodies because these measures are intrusive. This technique can occasionally irritate drivers, making it challenging to use.

*1) **EEG-Based Driver Fatigue Detection**:* M. Li, Z. Cheng, and J.F. Fang proposed the paper "An EEG-based approach for diagnosing drowsy driving circumstances" [12]. In this study, the driving experience was simulated using a vehicle driving simulation system. Both while the drivers were awake and when they were asleep, EEG data was recorded. Using FastICA arithmetic, the interference between EMG and EOG and EEG was eliminated. The power spectral density of the EEG data was calculated after the signals were converted from the time domain to the frequency domain using FFT. The fatigue index F from EEG is calculated using the following equation:

$$F = \frac{E_\delta + E_\theta}{E_\alpha + E_\beta}$$

The fatigue index F identifies the signs of drowsy driving. Driving exhaustion has an F-value that is much higher than that of normal driving, making it a useful tool for assessing the level of fatigue.

*2) **Wearable Driver Drowsiness Detection System**:* A paper on a smartwatch-based wearable EEG system for driver sleepiness detection was proposed by Li, Gang, Lee, Boo-Leng, and Chung, Wan-Young [27] in 2015. The system uses a smart watch and a headband with sensors to detect tiredness. The Support Vector Machine-based Posterior Probabilistic Model (SVMPPM) for Driver Drowsiness Detection is demonstrated in this work with the goal of converting the level of drowsiness into a value between 0 and 1 rather than discrete labels. A Bluetooth-enabled EEG headgear and a commercial smartwatch were used to analyse the provided model in real-time. With 15 subjects used to create the model and 5 subjects used to test it, 20 participants in a one-hour repetitive driving simulation experiment were utilised to evaluate this model. The proposed system attained an accuracy of up to 91.25%, which was the highest.

V. Conclusion

This study showed that the selected publications use different features depending on the scope of the research and data availability. Each research investigates drowsiness detection by machine learning, while its features are different. Studies also vary in biological, vehicle, and behavioural parameters. The choice of functions depends on the availability of the data set and the goal of the research. According to studies multiple feature models did not necessarily offer the optimum performance for drowsiness detection. To find the model with the best performance, models with more and less features should be tested. Different studies have employed a variety of algorithms. The results show that no concrete conclusion can be drawn about which model is the best, but they clearly show that some machine learning models are used more than others.

TABLE II
Driver Drowsiness Detection Techniques

I. Behavioural Parameter-Based Techniques			
Ref	**Features**	**Approach/Techniques**	**Efficiency**
[1]	Eye Blinking	Raspberry Pi Python OpenCV library	NIL
[3]	Eye State and Yawning	Dlib library and Python	96.71%
[2]	Face components(nose length, eye closing, yawning)	Aspect Ratio method and SVM	89.6%
[4]	Mouth and Yawning Analysis	SVM	NIL
II. Vehicular Parameter-Based Techniques			
Ref	**Features**	**Approach/Techniques**	**Efficiency**
[31]	Lane Detection	Extracting Approximate Entropy from Steering Wheel Angle	78.01%
[32]	Steering Wheel Angle	Hough Transform, voila jones algorithm	NIL
III. Physiological Parameter-Based Techniques			
Ref	**Features**	**Approach/Techniques**	**Efficiency**
[12]	EEG (electroencephalogram)	Fatigue Index (F)	NIL
[27]	Wearable Devices	SVM and Posterior Probabilistic model	91.25%

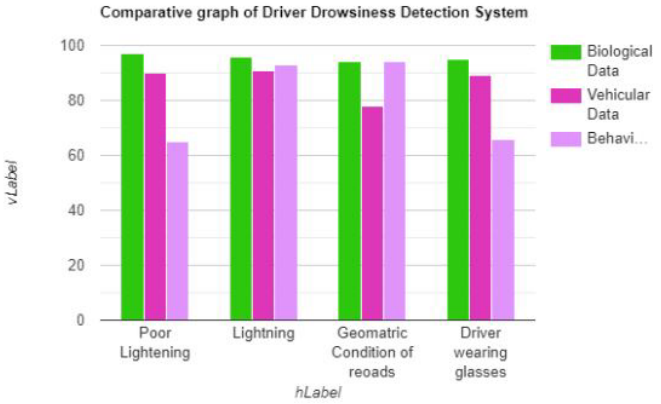


Fig. 1. Comparative graph Of Driver Drowsiness Detection System

The most used models are the techniques based on SVM, CNN, Aspect Ratio. Most studies used different machine learning models to test which model had the best detection.

REFERENCES

[1] Fouzia, R. Roopalakshmi, J. A. Rathod, A. S. Shetty and K. Supriya, "Driver Drowsiness Detection System Based on Visual Features," 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018, pp. 1344-1347, doi: 10.1109/ICICCT.2018.8473203.

[2] V. Kumar and R. Patra, "Driver drowsiness monitoring system using visual behaviour and machine learning," 2018 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE), 2018, pp. 339-344, doi: 10.1109/ISCAIE.2018.8405495.

[3] S. Mohanty, S. V. Hegde, S. Prasad and J. Manikandan, "Design of Real-time Drowsiness Detection System using Dlib," 2019 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE), 2019, pp. 1-4, doi: 10.1109/WIECON-ECE48653.2019.9019910.

[4] Mandalapu Saradadevi and Dr. Preeti Bajaj, "Driver Fatigue Detection Using Mouth and Yawning Analysis", IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.6, June 2008.

[5] A. A. Suhaiman, Z. May and N. A. A.Rahman, "Development of an intelligent drowsiness detection system for drivers using image processing technique," 2020 IEEE Student Conference on Research and Development (SCOReD), 2020, pp. 233-236, doi: 10.1109/SCOReD50371.2020.9250948.

[6] Bhole, Uzair & Chavan, Pravin & Gandhi, Siddharth & Gawde, Rohit Fakir, Kausar. (2020). "Drowsiness Detection And Monitoring System". ITM Web of Conferences. 32. 03045. 10.1051/itmconf/20203203045.

[7] W. Deng and R. Wu, "Real-Time Driver-Drowsiness Detection System Using Facial Features," in IEEE Access, vol. 7, pp. 118727-118738, 2019, doi: 10.1109/ACCESS.2019.2936663.

[8] J. Singh, "Learning based Driver Drowsiness Detection Model," 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), 2020, pp. 698-701, doi: 10.1109/ICISS49785.2020.9316131.

[9] O. Budak, V. Bajaj, Y. Akbulut, O. Atila and A. Sengur, "An Effective Hybrid Model for EEG-Based Drowsiness Detection," in IEEE Sensors Journal, vol. 19, no. 17, pp. 7624-7631, 1 Sept.1, 2019, doi: 10.1109/JSEN.2019.2917850.

[10] J. -J. Yan, H. -H. Kuo, Y. -F. Lin and T. -L. Liao, "Real-Time Driver Drowsiness Detection System Based on PERCLOS and Grayscale Image Processing," 2016 International Symposium on Computer, Consumer and Control (IS3C), 2016, pp. 243-246, doi: 10.1109/IS3C.2016.72.

[11] A. B. Albu, B. Widsten, T. Wang, J. Lan, and J. Mah ," A computer vision-based system for real-time detection of sleep onset in fatigued drivers" , 2008, doi: 10.1109/ivs.2008.4621133

[12] L. Li, Z. Cheng, and J.F. Fang ,"An EEG-based method for detecting drowsy driving state" ,2010, doi: 10.1109/fskd.2010.5569757.

[13] Lu Yuang ,Wang Zengcai, "Detecting driver yawning in successive images" ,2007 1st international conference on bioinformatics and biomedical engineering. Doi:10.1109/icbbe.2007.152

[14] S. Abtahi, b. Hariri, and S. Shirmohammadi, "Driver drowsiness monitor-ing based on yawning detection" ,2011,ieee international instrumentation and measurement technology conference. Doi:10.1109/imtc.2011.5944101

[15] Abelakhdar, W. Kaaniche, R. Djmel and B. Ouni ," A comparison between ann and svm classifier for drowsiness detection based on single eeg channel" ,2016, doi:10.1109/atsip.2016.7523132 .

[16] Ratnarup Dey, Joy Paulose," An improved algorithm for drowsiness detection for non-intrusive", 2018.

[17] Feng You, Xiaolong Li, Yunbo Gong, Haiwei Wang and Hongyi Li," A Real-time driving drowsiness detection algorithm with individual differences consideration" ,2019.

[18] Belal, Abdullah Salem, Mohamed el Amine, Meriem, Abdelaziz, Mokhtar," Driver drowsiness detection system", 2013.

[19] Javed Ahmed, Jian-ping li, Saeed Ahmed Kran, Riaz Ahmed Shaikh," Eye behaviour based drowsiness detection system",2015.

[20] Avigyan sinha, Aneesh R P, Sarada K Gopal," Drowsiness detection system using deep learning",2021.

[21] Rashi Bhargava, Ravina Panchal, Preeti Verma, Shivam Sharma," Drowsiness detection while driving using eye tracking",2020.

[22] N.Radha, e. M. Malathy, r. Swathika, r. B. Jananie a. A. Silviya," Drowsiness detection system using visual articulators" ,2021.

[23] M. A. Assari and M. Rahmati," Driver drowsiness detection using face expression recognition",2011.

[24] S. Sathasivam, a. K. Mahamad, s. Saon, a. Sidek, m. M. Som and h. A. Ameen," Drowsiness detection system using eye aspect ratio technique", 2020.

[25] Sinan kaplan, Mehmet Amac Guvensa, Ali Gokhan Yavuz, and Yasin Karal," Driver behavior analysis for safe driving: a survey", 2015.

[26] Jo-Ho Choi, Sung Kyung Hong and Yong-Guk Kim, "Real-time categorization of driver's gaze zone using the deep learning techniques," 2016 International Conference on Big Data and Smart Computing (BigComp), 2016, pp. 143-148, doi: 10.1109/BIGCOMP.2016.7425813.

[27] Li, gang and Lee, Boon-Leng and Chung, Wan-Young," Smartwatch based wearable EEG system for driver drowsiness detection",2015

[28] Li, Lingling and Chen, Yangzhou and Li, Chenlong," Yawning detection for monitoring driver fatigue based on two cameras",2009

[29] R. N. Khushaba, S. Kodagoda, S. Lal, and G. Dissanayake," Driver drowsiness classification using fuzzy wavelet-packet-based feature extraction algorithm",2011.

[30] C. Liang, M. L. Reyes, and J. D. Lee," Real-time detection of driver cognitive distraction using support vector machines",2007.

[31] Z. Li, S. E. Li, R. Li, B. Cheng, and J. Shi, "Online detection of driver fatigue using steering wheel angles for real driving conditions," Sensors, vol. 17, no. 3, p. 495, Mar. 2017.

[32] Y. Katyal, S. Alur, and S. Dwivedi, "Safe driving by detecting lane discipline and driver drowsiness," in Proc. IEEE Int. Conf. Adv. Commun., Control Comput. Technol., May 2014, pp. 1008–1012.

45% Overall Similarity

Top sources found in the following databases:

- 37% Internet database
- 42% Publications database
- Crossref database
- Crossref Posted Content database

TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	researchgate.net Internet	4%
2	Muhammad Ramzan, Hikmat Ullah Khan, Shahid Mahmood Awan, Ami... Crossref	3%
3	cse.anits.edu.in Internet	3%
4	ijartet.com Internet	2%
5	Sinan Kaplan, Mehmet Amac Guvensan, Ali Gokhan Yavuz, Yasin Karal... Crossref	2%
6	Jagendra Singh. "Learning based Driver Drowsiness Detection Model", ... Crossref	2%
7	ijircce.com Internet	2%
8	ijert.org Internet	1%
9	B. N. Manu. "Facial features monitoring for real time drowsiness detect... Crossref	1%

10	M. Kolodziej, P. Tarnowski, D. Sawicki, A. Majkowski, R.J. Rak, A. Bala, ... Crossref	<1%
11	Thomas van Klompenburg, Ayalew Kassahun, Cagatay Catal. "Crop yiel... Crossref	<1%
12	R P Janani, K. Lakshmi Narayanan, R. Santhana Krishnan, P. Kannan, R.... Crossref	<1%
13	Shruti Mohanty, Shruti V Hegde, Supriya Prasad, J. Manikandan. "Desig... Crossref	<1%
14	disco.uni-muenster.de Internet	<1%
15	Yuvraj Suryawanshi, Sushma Agrawal. "Driver Drowsiness Detection Sy... Crossref	<1%
16	ijana.in Internet	<1%
17	Wanli Li, Jing Huang, Guoqi Xie, Fakhri Karray, Renfa Li. "A survey on vi... Crossref	<1%
18	ece.uvic.ca Internet	<1%
19	bro.gov.in Internet	<1%
20	Alexey Kashevnik, Igor Lashkov, Andrew Ponomarev, Nikolay Teslya, A... Crossref	<1%
21	Ayman Altameem, Ankit Kumar, Ramesh Chandra Poonia, Sandeep Ku... Crossref	<1%

22	Gang Li, Boon-Leng Lee, Wan-Young Chung. "Smartwatch-Based Wear... Crossref	<1%
23	"Innovative Data Communication Technologies and Application", Sprin... Crossref	<1%
24	ijream.org Internet	<1%
25	Plínio M.S. Ramos, Caio B.S. Maior, Márcio C. Moura, Isis D. Lins. "Aut... Crossref	<1%
26	ijesc.org Internet	<1%
27	link.springer.com Internet	<1%
28	docplayer.net Internet	<1%
29	"Soft Computing and Signal Processing", Springer Science and Busines... Crossref	<1%
30	J. Vijay, J. Sai Krishna, M. Mohamed Thazeem, Jeremiah Asir, N. Srene... Crossref	<1%
31	Roaa Albasrawi, Fajer F. Fadhil, Mohammed Talal Ghazal. "Driver drow... Crossref	<1%
32	dblp.dagstuhl.de Internet	<1%
33	tutcris.tut.fi Internet	<1%

34	christuniversity.in Internet	<1%
35	sersc.org Internet	<1%
36	ieeexplore.ieee.org Internet	<1%
37	ijaret.org Internet	<1%
38	Chiagoziem C. Ukwuoma, Chen Bo. "Deep Learning Review on Drivers ... Crossref	<1%
39	Nadir N. Charniya, Vivek R. Nair. "Drunk driving and drowsiness detecti... Crossref	<1%
40	Riad Alharbey, Mohamed M. Dessouky, Ahmed Sedik, Ali I. Siam, Moha... Crossref	<1%
41	ijsrset.com Internet	<1%
42	research-publication.com Internet	<1%
43	scilit.net Internet	<1%
44	aicte-india.org Internet	<1%
45	ceur-ws.org Internet	<1%

46	eprints.utp.edu.my Internet	<1%
47	coursehero.com Internet	<1%
48	Aamir Nizam Ansari, Arundhati Navada, Sanchit Agarwal, Siddharth Pa... Crossref	<1%
49	S. Kailasam, M. Karthiga, R.M. Priyadarshini, K. Kartheeban, K. Anithad... Crossref	<1%
50	dokumen.pub Internet	<1%
51	Mohit Dua, Shakshi, Ritu Singla, Saumya Raj, Arti Jangra. "Deep CNN ... Crossref	<1%
52	jctjournal.com Internet	<1%
53	itm-conferences.org Internet	<1%
54	"International Conference on Intelligent Data Communication Technolo... Crossref	<1%
55	Ahmad F. Klaib, Nawaf O. Alsrehin, Wasen Y. Melhem, Haneen O. Bash... Crossref	<1%
56	Ibrahim, Masrullizam Mat, John J. Soraghan, Lykourgos Petropoulakis,... Crossref	<1%
57	SpringerBriefs in Computer Science, 2014. Crossref	<1%

58	content.sciendo.com Internet	<1%
59	doaj.org Internet	<1%
60	ejmcm.com Internet	<1%
61	Chao Zhang, Xiaopei Wu, Xi Zheng, Shui Yu. "Driver Drowsiness Detecti... Crossref	<1%
62	Ishan Vatsaraj, Yash Jungade, Nitee Dhuri, Rutwij Mulye, Shubham Dal... Crossref	<1%
63	Saravanaraj Sathasivam, Abd Kadir Mahamad, Sharifah Saon, Azmi Sid... Crossref	<1%
64	Yadong Xie, Fan Li, Yue Wu, Song Yang, Yu Wang. "Real-Time Detectio... Crossref	<1%
65	utupub.fi Internet	<1%
66	0-www-crossref-org.libus.csd.mu.edu Internet	<1%
67	Jithina Jose, J S Vimali, P Ajitha, S Gowri, A Sivasangari, Bevish Jinila. ... Crossref	<1%
68	Sigari, Mohamad-Hoseyn, Mahmood Fathy, and Mohsen Soryani. "A Dri... Crossref	<1%
69	export.arxiv.org Internet	<1%

70	ijisrt.com Internet	<1%
71	arxiv-vanity.com Internet	<1%
72	Ralph Oyini Mbouna, Seong G. Kong, Myung-Geun Chun. "Visual Analys... Crossref	<1%
73	"Image Processing and Capsule Networks", Springer Science and Busi... Crossref	<1%
74	Mukesh Kumar Kamti, Rauf Iqbal. "Evolution of Driver Fatigue Detectio... Crossref	<1%
75	Yan Wang, Lisheng Jin, Keyong Li, Baicang Guo, Yi Zheng, Jian Shi. "Dr... Crossref	<1%

Driver Drowsiness Detection System using Visual Features and Machine Learning: A systematic literature review

Rutuja Banginwar
Computer Engineering
Pune Institute of Computer Technology
Pune, India
rutujabanginwar01@gmail.com

Imon Mandal
Computer Engineering
Pune Institute of Computer Technology
Pune, India
imonashokmandal@gmail.com

Urmila Matade
Computer Engineering
Pune Institute of Computer Technology
Pune, India
matadeurmila1712@gmail.com

Mayur Mote
Computer Engineering
Pune Institute of Computer Technology
Pune, India
mayurmote8055@gmail.com

Prajkta Khadkikar
Computer Engineering
Pune Institute of Computer Technology
Pune, India
pakhadkikar@pict.edu

***Abstract*—Over the past ten years, the number of motor vehicles has steadily grown in emerging nations. The reports of traffic accidents note that risky driving behaviors, such as drunken driving or fatigued driving, are responsible for most accidents. According to studies, sleepy driving is a contributing factor in 20% of all accidents. Drowsiness is a condition when the level of consciousness is lowered as a result of fatigue or lack of sleep, and it can make a driver fall asleep silently. Drowsy driving causes the driver to lose control of the vehicle, causing it to drift off the road, collide with an obstacle, or overturn. In this study, we carried out a Systematic Literature Review (SLR) to extract and synthesize the algorithms and features that have been used in drowsiness detection studies. Based on our search criteria, we retrieved 100 relevant studies, of which we have selected 30 studies for further analysis. We investigated these selected studies carefully, analyzed the techniques and features used, and found out the research gap as well. According to our analysis, the most used features are facial expressions like yawning, closing of the eyes, and head motions. After this observation based on the analysis of 30 papers, widely used techniques are Eye Aspect Ratio, Haar Cascade, Support Vector Machine(SVM), dlib library.**

***Index Terms*—feature extraction, visual features, machine learning , computer vision, drowsiness detection.**

I. Introduction

Drowsiness is one of the main factors that threaten road safety and cause serious injuries, deaths and economic losses. Lack of alertness, caused by the unconscious transition from wakefulness to sleep, leads to several serious traffic accidents[1].

Drowsiness and distraction are two areas of study that researchers focused on when designing systems to monitor driver inattention. Systems designed for sleepiness analysis and detection can generally be divided into two categories: visual feature-based and non-visual feature-based[2]. Techniques using visual features use computer vision approaches to detect drowsiness. The use of visual features focuses on the extraction of facial features such as the face, eyes and mouth. Analysis of eye and mouth conditions can provide observable cues for the detection process. Techniques using visual features can mainly be divided into four categories: eye state analysis, blink analysis, mouth and yawn analysis, and facial expression analysis. Image processing and machine learning techniques are the two main steps for visual feature detection and processing. Non-visual feature-based techniques are generally intrusive and fall into two categories: driver physiological analysis and vehicle parameter analysis. For example, in EEG technology, several sensors must be attached to the driver while driving. The acquisition system, processing system, and warning system are the three components or modules that make up the fundamental sleepiness detection system. Here, the video of the driver's front face is captured in the acquisition system and transferred to the processing block where it is processed online to detect drowsiness. A warning or alert is delivered to the driver by the warning system if drowsiness is discovered[4].

The rest of this document is organized as follows: Section 2 explains the background. Section 3 discusses the methodology. Section 4 discusses the various drowsiness detection techniques. Section 5 presents the results of SLR and Section 6 concludes this document.

II. Related Work

With a total road length of approximately 62.1 lakh kilometers, India has the world's second largest road network. This vast network serves as the nation's lifeline, transporting over 64.5% of all goods within the country and being the preferred mode of transportation for more than 90% of India's passenger traffic. While roads continue to be synonymous

with development and growth in the country, they have also been a nemesis for users, with India having the dubious distinction of leading the global tally of annual deaths and injuries due to road accidents. There is an asymmetry between the number of vehicles and the number of deaths due to road accidents, with India's one percent global share of the number of vehicles accounting for nearly 11% of the deaths due to road accidents. Despite concerted efforts at all levels, road accidents continue to be the leading cause of deaths, disabilities, and hospitalizations in the country. Lack of consciousness while driving due to sleepiness is the root cause of all road accidents. If a drowsy driver is informed in a timely manner, then a lot of road accidents can be avoided and many lives can be saved. There are different techniques and algorithms to detect drowsiness. In this review article, the research work that has already been published on detecting drowsiness has been investigated.

While searching for the research paper, we excluded the research works that were very old and only considered the latest research. Shruti Mohanty, Shruti V Hegde, Supriya Prasad, and J. Manikandan has published a paper to detect drowsiness using opencv, python, and dlib to calculate the Euclidean aspect ratio of eyes and mouth to detect drowsiness.

In 2020, Jagendra Singh proposed the paper on Learning based Driver Drowsiness Detection Model, In that the EAR is calculated for the eyes using the 6 landmarks around the eye ROI. A threshold is set for 0.3 and if the EAR becomes less than the threshold for at least 20 frames, an alert is generated to the driver predicting that they have drowsy eyes. If the landmarks around the eyes ROI and mouth ROI cannot be determined for at least 15 frames, an alert is generated to the driver insisting the driver pay attention to the road.

The paper titled "Real-time detection of driver cognitive distraction using support vector machines" was proposed by Y. Liang, M. L. Reyes, and J. D. Lee[30], here they have used a real time approach for detecting distraction using the driver's eye movements and driving performance data collected in a simulator environment called the in-vehicle information system. After that, for detecting the driver distraction the data were used to train and test both support vector machine (SVM) and logistic regression models. Thus it highly depends on the vehicle type, driver experience, and the road condition.Such systems can fail because the car doesn't provide important information if the driver falls asleep on a straight road.

In 2009, Li, Lingling and Chen, Yangzhou and Li, Zhenlong[28] proposed paper on Yawning detection for monitoring driver fatigue based on two cameras, in it they monitors yawning patterns of the driver using two separate cameras to acquire information of the upper part of the body in order to track the driver's mouth, here limitation is that hardware dependency is more.

A. B. Albu, B. Widsten, T. Wang, J. Lan, and J. Mah[11] proposed a paper "A computer vision-based system for real-time detection of sleep onset in fatigued drivers" in 2008. After reading this paper we understand that real-time performance was achieved by focusing on one single visual cue (i.e. eye condition) and custom designed template-matching algorithm for online eye condition detection. Also additionally designed a computer vision-based sleep monitoring technique will be implemented through customer-designed DSPs or FPGAs built into the vehicle.

The study used a vehicle driving simulation system that was suggested in the 2010 work "An EEG-based technique for identifying drowsy driving conditions" by M. Li, Z. Cheng, and J.F. Fang[12]. EEG data from sober and drowsy drivers were obtained. FastICA arithmetic was used to remove EMG and EOG interference EEG. Fatigue index F from EEG exceeded equation can indicate fatigue management characteristics. The driving fatigue F value was significantly higher than normal driving, so it can be established to assess and evaluate degree of fatigue.

A study conducted in the paper "Detecting Driver Yawning in Successive Images" published in 2007 by Lu Yufeng, Wang Zengcai[13] concludes that based on the location of the center of the nostrils and chin, we can detect driver yawning by calculating the vertical distance between them. They said the method was robust to variant lighting because the method was not based on color or intensity, which is fragile for lighting. In 2011, S. Abtahi, B. Hariri, and S. Shirmohammadi suggested in a study that it would be possible to choose a complex algorithm that would be one step closer to the actual implementation of the system under "Yawn Detection-Based Driver Drowsiness Monitoring" than the proposed methodology .

Choi, In-Ho, Sung Kyung Hong, and Yong-Guk Kim[26] published a paper in 2016 , "Real-time categorization of driver's gaze zone using the deep learning techniques". It used deep learning and facial expression recognition. The proposed system achieved a mean detection accuracy of over 95%. This approach has the disadvantage that it requires a huge amount of data to train the neural network to work with a high level of accuracy.

Fouzia, Roopalakshmi, and Supriya [1] presented the hardware-based driver drowsiness detection system based on visual features. This system makes use of the Raspberry Pi and the Python OpenCV library for drowsiness detection based on eye blink count. The proposed model continuously monitors the driver's eye movement and warns them by turning on the vibrator when they are drowsy. A vibrator signal is generated to alert the driver if the eyes are detected to have been closed for a long period of time. One of the main contributing factors to traffic accidents is drowsiness. To avoid this issue, Ashish and Rusha [2] proposed a driver drowsiness detection system based on mouth and yawning analysis. First, facial landmarks on the detected face are identified, and then the eye aspect ratio, mouth opening ratio, and nose length ratio are computed to detect drowsiness using adaptive thresholding based on these values. The Drowsiness Detection System Using the Eye Aspect Ratio Technique was proposed by Sathasivam, Mahamad, and Ameen [24]. The basic objective of the proposed method is to detect drowsiness in drivers and warn them by raising an alarm. This system uses a camera mounted in front of the driver to capture real-time video, then

captured video frames are then used to detect drowsiness using the Eye Aspect Ratio (EAR) technique.

A study named "Eye Behaviour based Drowsiness Detection System" by Javed Ahmed, Jian-Ping Li, Saeed Ahmed Kran, and Riaz Ahmed Shaikh[19] was published in IEEE in 2015. The creation of a Drowsy Driver Detection System has made use of a non-intrusive computer vision-based concept. In order to specifically identify driver drowsiness, a system that uses the small camera and analyzes the driver's eyes while focusing directly on his or her face has been developed. When a driver's weariness is apparent, a warning sign is displayed to alert other road users. The procedure of locating the driver's eyes and determining whether they are open or closed is demonstrated in this paper. The system is able to find the margins of a face and determine where a person's eyes may be by using binary image data that was acquired. The proposed method ensures that the driver is dozing off if their eyelids are closed for five consecutive frames, at which point a warning signal is sent. The framework also functions in conditions with reasonable lighting and can detect when the eyes are hidden from view.

An improved algorithm for drowsiness detection for non-intrusive driving was published in 2018 by Ratnarup Dey and Joy Paulose[16] in the International Journal of Applied Engineering Research. The paper discusses a car-integrated system that monitors the behaviour of the driver based on a variety of stimuli. For the purpose of detecting head movement, steering gripping, and drunk driving, sensor characteristics are employed. As a part of this work, a system that can detect driver drowsiness by evaluating any one, two, or all three of the following parameters head movement, alcohol influence, and steering grasping is proposed. When one, two, or all of these circumstances exist, it is assumed that the driver is drowsy.

III. Methodology

A. Review protocol

The initial step is to define the research questions. Relevant studies are chosen from the databases once the research questions are ready. Science Direct, Springer Link and Google Scholar were the databases used in this study. A set of exclusion and quality criteria were used to filter and evaluate the relevant research after they had been chosen. From the chosen studies, all relevant data are retrieved, and ultimately, the extracted data are synthesized to address the research objectives.

B. Research questions

The purpose of this literature review is to find out already published algorithms and techniques for drowsiness detection and evaluate their performance. To prepare this literature review following research questions have been defined.

RQ1:- Which technique has been used for drowsiness detection?

RQ2:- What algorithms have been used to identify drowsiness?

RQ3:- What are the different visual features that the system takes into account?

RQ4:- What is the research gap in the proposed algorithm?

C. Search strategy

An automated search handles the fundamental searching. The search's initial inputs were "machine learning", "drowsiness detection system" AND "driver". Three databases were searched in this operation. To find a wide range of papers, the search terms "driver" AND "drowsiness detection system" were used.

Following is a detailed explanation of the search terms for each database:

ResearchGate: The search term is ["machine learning" AND "driver drowsiness detection"] and ["driver drowsiness detection"]

Springer Link: The search term is ["drowsiness detection system" AND "yawning"] and ["drowsiness recognition" AND "facial features"]

Google Scholar: The search term is ["driver drowsiness detection" AND "eyes aspect ratio"] and ["image processing" AND "drowsiness detection" AND "machine learning"]

D. Exclusion criteria

The papers were evaluated based on exclusion criteria to determine which ones should be excluded from the review. Listed below are the exclusion criteria (EC):

Exclusion criteria 1: Paper is not relevant to the drowsiness detection system

Exclusion criteria 2: Paper is not written in English

Exclusion criteria 3: Duplicate or already retrieved paper

Exclusion criteria 4: Full text of the paper is not available

Exclusion criteria 5: Paper has been published before 2007

TABLE I
SEARCH STRATERGY

Database	# Of initially retrieved papers	# Of papers after exclusion criteria	Percentage of Papers (%)
ResearchGate	7	1	14
Springer	13	2	15
Google Scholar	80	27	34

IV. DROWSINESS DETECTION TECHNIQUES

Depending on the technique being employed, the Driver Drowsiness detection system continuously observes the driver's physical behaviour, vehicle movement patterns, or environmental factors. Methods for detecting sleepiness are often divided into three categories.

1) Methods based on behavioural parameter data
2) Methods that are based on vehicle parameters
3) Methods using physiological parameters

A. *Behavioural Parameter-Based Techniques*

Computer vision algorithms and cameras were combined with behavioral-based strategies to extract behavioural information. The non-intrusive methods of sleepiness detection are behavioural parameters. The behavioural indicators used by

these strategies to gauge driver weariness include eye closure ratio, eye blinking, head position, facial expressions, and yawning. Environmental elements, such as lighting, brightness, and road conditions, have an impact on the measurement's reliability and accuracy, which are issues with behavioural measures.

*1) **Eye Blink Detection Method**:* A real-time drowsiness detection system utilizing Dlib was proposed by Mohanty, Hegde, and Manikandan [3]. The Dlib model and Python are used to create a drowsy driver detection system in this study. The coordinates of the facial landmarks in the input video are mapped using Dlib's form detector, which is also used to track the aspect ratios of the eyes and mouth to look for signs of tiredness. For the eyes and mouth, respectively, predefined threshold values of 0.15 and 0.83 are used to compare the estimated aspect ratios with each. The average aspect ratio of the left and right eyes must be above the threshold in order to be considered drowsy. The driver is alerted if indicators of fatigue are noticed for an extended period of time. Testing films from a common public dataset as well as real-time video is done to evaluate the performance of the suggested system. The greatest recognition accuracy for video datasets for the proposed system was 96.71

*2) **Eye and Yawning Analysis**:* A real-time drowsiness detection system utilizing Dlib was proposed by Mohanty, Hegde, and Manikandan [3]. The Dlib model and Python are used to create a drowsy driver detection system in this study. The coordinates of the facial landmarks in the input video are mapped using Dlib's form detector, which is also used to track the aspect ratios of the eyes and mouth to look for signs of tiredness. For the eyes and mouth, respectively, predefined threshold values of 0.15 and 0.83 are used to compare the estimated aspect ratios with each. The average aspect ratio of the left and right eyes must be above the threshold in order to be considered drowsy. The driver is alerted if indicators of fatigue are noticed for an extended period of time. Testing films from a common public dataset as well as real-time video is done to evaluate the performance of the suggested system. The greatest recognition accuracy for video datasets for the proposed system was 96.71%.

*3) **Nose, Eye and Yawning Analysis Method**:* One of the main contributing factors to traffic accidents is drowsiness. Ashish and Rusha [2] suggested a driver sleepiness detection system based on mouth and yawning analysis to get over this problem. The eye aspect ratio, mouth opening ratio, and nose length ratio are computed to identify drowsiness using adaptive thresholding based on these values once facial landmarks on the observed face are first identified. To find the face in the frames, this technique employs a histogram of oriented gradients (HOG) and a linear support vector machine (SVM). Image normalisation and an ensemble of regression trees are used to estimate the landmark positions on the face. Fisher's linear discriminant analysis (FLDA) and Support Vector Machine (SVM) with linear kernel were then used for classification. The INVEDRIFAC dataset has been used to test the newly built algorithm. SVM and FLDA have been found to perform better than Bayesian classifiers. Both FLDA and SVM have a specificity of 1, while their respective sensitivity values are 0.896 and 0.956.

*4) **Mouth and Yawning Analysis**:* Saradadevi and Bajaj [4] suggested a mouth and yawning analysis-based driver fatigue detection system. The Viola-Jones classifier is used to first identify the driver's face from the input frames. After that, the mouth and yawning image datasets are trained using SVM. Finally, yawning and sleepiness are detected by classifying the mouth regions using SVM. A dataset of more than 100 typical films and 20 yawning photos is chosen for the experiment. The findings show that the proposed system performs better than the system based on geometric features.

B. Vehicular Parameter-Based Techniques

Vehicle attributes like frequent lane changes, variable vehicle speeds, steering wheel tilt, steering wheel grip force, etc. are used in vehicle parameter-based strategies to identify driver weariness. These precautions call for sensors on car parts like the steering wheel, accelerator, and brake pedals, among others. These sensors produce signals that are used to analyse driver sleepiness. Principal objective of these techniques is to analyse driving behaviours and spot a drop in driving performance brought on by fatigue and exhaustion. However, evaluating driver weariness based on vehicle movement has its limitations because measurement results are often influenced by outside variables like the geometry of the road and the weather.

*1) **Real Time Lane Detection System**:* In the modern era, traffic accidents are all too common, endangering both the lives of those on the road and their property. Road accidents can occur for a variety of causes, including: rash driving, inexperience, disobeying signs, jumping signals, etc. The Drivers' Drowsiness Detection system was suggested by Katyal et al. [32] as a solution to the problems. The system operates in two stages: first, it uses the hough transform to recognise lanes. Secondly, it looks for signs of tiredness in the drivers' eyes. The viola jones approach is used to detect faces first, followed by picture segmentation, otsu thresholding, and clever edge detection for eye detection. The data are used with the circle detection through transform method to find eyeballs and gauge level of exhaustion. Low lightning conditions won't affect its effectiveness. The results indicate that the suggested strategy is beneficial for drivers who take long journeys, drive at night, or drive after drinking.

*2) **Steering Wheel Angle For Real-Time Driving Conditions For DDT**:* Li et al. [31] suggested the online detection of Drowsiness Detection System to track the level of driver fatigue under actual driving conditions utilising steering wheel angles (SWA). The sensors mounted on the steering lever are where the SWA data is gathered. The system first extracts the approximate entropy (ApEn) features from fixed sliding windows on a time series of real-time steering wheel angles. Next, the system linearizes the ApEn features utilising the deviation of adaptive linear piecewise fitting approach. The technique then determines the warping distance between a

series of sample data's linear characteristics. Finally, the system uses warping distance in accordance with the created decision classifier to determine the drivers' level of tiredness. The empirical analysis employs data gathered over 14.68 hours of actual driving time, evaluated on two levels of fatigue: drowsy and awake. Results indicate that the suggested system is beneficial for preventing traffic accidents brought on by driver drowsiness and capable of operating online with an accuracy of 78.01

C. *Physiological Parameter-Based Techniques*

Based on the drivers' physiological data, such as heart rate, pulse rate, breathing rate, respiratory rate, and body temperature, the physiological parameters-based algorithms identify drowsiness. These biological indicators are more dependable and accurate at detecting tiredness because they focus on the physical state of the driver. Systems for detecting drowsiness based on physiological characteristics notice these modifications and warn the driver when he is getting close to falling asleep. Electrodes must be positioned on the drivers' bodies because these measures are intrusive. This technique can occasionally irritate drivers, making it challenging to use.

*1) **EEG-Based Driver Fatique Detection**:* M. Li, Z. Cheng, and J.F. Fang proposed the paper "An EEG-based approach for diagnosing drowsy driving circumstances" [12]. In this study, the driving experience was simulated using a vehicle driving simulation system. Both while the drivers were awake and when they were asleep, EEG data was recorded. Using FastICA arithmetic, the interference between EMG and EOG and EEG was eliminated. The power spectral density of the EEG data was calculated after the signals were converted from the time domain to the frequency domain using FFT. The fatigue index F from EEG is calculated using the following equation:

$$F = \frac{E_\delta + E_\theta}{E_\alpha + E_\beta}$$

The fatigue index F identifies the signs of drowsy driving. Driving exhaustion has an F-value that is much higher than that of normal driving, making it a useful tool for assessing the level of fatigue.

*2) **Wearable Driver Drowsiness Detection System**:* A paper on a smartwatch-based wearable EEG system for driver sleepiness detection was proposed by Li, Gang, Lee, Boon-Leng, and Chung, Wan-Young [27] in 2015. The system uses a smart watch and a headband with sensors to detect tiredness. The Support Vector Machine-based Posterior Probabilistic Model (SVMPPM) for Driver Drowsiness Detection is demonstrated in this work with the goal of converting the level of drowsiness into a value between 0 and 1 rather than discrete labels. A Bluetooth-enabled EEG headgear and a commercial smartwatch were used to analyse the provided model in real-time. With 15 subjects used to create the model and 5 subjects used to test it, 20 participants in a one-hour repetitive driving simulation experiment were utilised to evaluate this model. The proposed system attained an accuracy of up to 91.25%, which was the highest.

TABLE II
DRIVER DROWSINESS DETECTION TECHNIQUES

I. Behavioural Parameter-Based Techniques			
Ref	**Features**	**Approach/Techniques**	**Efficiency**
[1]	Eye Blinking	Raspberry Pi Python OpenCV library	NIL
[3]	Eye State and Yawning	Dlib library and Python	96.71%
[2]	Face components(nose length, eye closing, yawning)	Aspect Ratio method and SVM	89.6%
[4]	Mouth and Yawning Analysis	SVM	NIL
II. Vehicular Parameter-Based Techniques			
Ref	**Features**	**Approach/Techniques**	**Efficiency**
[31]	Lane Detection	Extracting Approximate Entropy from Steering Wheel Angle	78.01%
[32]	Steering Wheel Angle	Hough Transform, voila jones algorithm	NIL
III. Physiological Parameter-Based Techniques			
Ref	**Features**	**Approach/Techniques**	**Efficiency**
[12]	EEG (electroencephalogram)	Fatigue Index (F)	NIL
[27]	Wearable Devices	SVM and Posterior Probabilistic model	91.25%

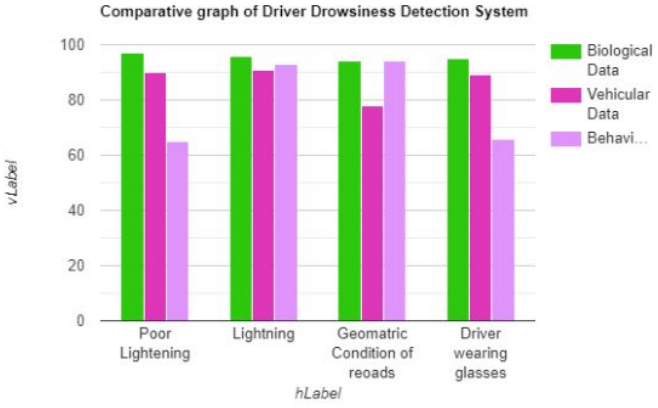


Fig. 1. Comparative graph Of DDS

V. CONCLUSION

This study showed that the selected publications use different features depending on the scope of the research and data availability. Each paper explores sleepiness detection using machine learning, but differs in features. Studies also vary in biological, vehicle, and behavioral parameters. The choice of functions depends on the availability of the data set and the goal of the research. Studies have also reported

that models with multiple features did not always provide the best performance for sleepiness detection. To find the model with the best performance, models with more and less features should be tested. Different studies have employed a variety of algorithms. The results show that no concrete conclusion can be drawn about which model is the best, but they clearly show that some machine learning models are used more than others. The most used models are the techniques based on SVM, CNN, Aspect Ratio. Most studies used different machine learning models to test which model had the best detection.

REFERENCES

[1] Fouzia, R. Roopalakshmi, J. A. Rathod, A. S. Shetty and K. Supriya, "Driver Drowsiness Detection System Based on Visual Features," 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018, pp. 1344-1347, doi: 10.1109/ICICCT.2018.8473203. **https://ieeexplore.ieee.org/abstract/document/8473203**

[2] A. Kumar and R. Patra, "Driver drowsiness monitoring system using visual behaviour and machine learning," 2018 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE), 2018, pp. 339-344, doi: 10.1109/ISCAIE.2018.8405495. **https://ieeexplore.ieee.org/abstract/document/8405495**

[3] S. Mohanty, S. V. Hegde, S. Prasad and J. Manikandan, "Design of Real-time Drowsiness Detection System using Dlib," 2019 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE), 2019, pp. 1-4, doi: 10.1109/WIECON-ECE48653.2019.9019910. **https://ieeexplore.ieee.org/abstract/document/9019910**

[4] Mandalapu Saradadevi and Dr. Preeti Bajaj, "Driver Fatigue Detection Using Mouth and Yawning Analysis", IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.6, June 2008. **http://paper.ijcsns.org/07_book/200806/20080624.pdf**

[5] A. A. Suhaiman, Z. May and N. A. A.Rahman, "Development of an intelligent drowsiness detection system for drivers using image processing technique," 2020 IEEE Student Conference on Research and Development (SCOReD), 2020, pp. 233-236, doi: 10.1109/SCOReD50371.2020.9250948. **https://ieeexplore.ieee.org/abstract/document/9250948**

[6] Ghole, Uzair & Chavan, Pravin & Gandhi, Siddharth & Gawde, Rohit Fakir, Kausar. (2020). "Drowsiness Detection And Monitoring System". ITM Web of Conferences. 32. 03045. 10.1051/itmconf/20203203045. **https://doi.org/10.1051/itmconf/20203203045**

[7] W. Deng and R. Wu, "Real-Time Driver-Drowsiness Detection System Using Facial Features," in IEEE Access, vol. 7, pp. 118727-118738, 2019, doi: 10.1109/ACCESS.2019.2936663. **https://ieeexplore.ieee.org/abstract/document/8808931**

[8] JJ. Singh, "Learning based Driver Drowsiness Detection Model," 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), 2020, pp. 698-701, doi: 10.1109/ICISS49785.2020.9316131. **https://ieeexplore.ieee.org/document/9316131**

[9] U. Budak, V. Bajaj, Y. Akbulut, O. Atila and A. Sengur, "An Effective Hybrid Model for EEG-Based Drowsiness Detection," in IEEE Sensors Journal, vol. 19, no. 17, pp. 7624-7631, 1 Sept.1, 2019, doi: 10.1109/JSEN.2019.2917850. **https://ieeexplore.ieee.org/abstract/document/8718312**

[10] J. -J. Yan, H. -H. Kuo, Y. -F. Lin and T. -L. Liao, "Real-Time Driver Drowsiness Detection System Based on PERCLOS and Grayscale Image Processing," 2016 International Symposium on Computer, Consumer and Control (IS3C), 2016, pp. 243-246, doi: 10.1109/IS3C.2016.72. **https://ieeexplore.ieee.org/abstract/document/7545182**

[11] A. B. Albu, B. Widsten, T. Wang, J. Lan, and J. Mah ," A computer vision-based system for real-time detection of sleep onset in fatigued drivers" , 2008, doi: 10.1109/ivs.2008.4621133 **https://ieeexplore.ieee.org/document/4621133**

[12] M. Li, Z. Cheng, and J.F. Fang ,"An eeg-based method for detecting drowsy driving state" ,2010, doi: 10.1109/fskd.2010.5569757. **https://ieeexplore.ieee.org/document/5569757**

[13] Lu yufeng ,Wang zengcai, "Detecting driver yawning in successive images",2007 1st international conference on bioinformatics and biomedical engineering. Doi:10.1109/icbbe.2007.152 **https://ieeexplore.ieee.org/document/4272636?arnumber=4272636**

[14] S. Abtahi, b. Hariri, and s. Shirmohammadi, "driver drowsiness monitor-ing based on yawning detection" ,2011,ieee international instrumentation and measurement technology conference. Doi:10.1109/imtc.2011.5944101 **https://ieeexplore.ieee.org/document/5944101**

[15] I.belakhdar, W. Kaaniche, R. Djmel and B. Ouni ," A comparison between ann and svm classifier for drowsiness detection based on single eeg channel" ,2016, doi:10.1109/atsip.2016.7523132 . **https://ieeexplore.ieee.org/document/7523132**

[16] Ratnarup dey, joy paulose," an improved algorithm for drowsiness detection for non-intrusive", 2018.

[17] Feng you, xiaolong li, yunbo gong, haiwei wang and hongyi li," a real-time driving drowsiness detection algorithm with individual differences consideration" ,2019.

[18] Belal, abdullah salem, mohamed el amine, meriem, abdelaziz, mokhtar," driver drowsiness detection system", 2013.

[19] Javed ahmed, jian-ping li, saeed ahmed kran, riaz ahmed shaikh," eye behaviour based drowsiness detection system",2015.

[20] Avigyan sinha, aneesh r p, sarada k gopal," drowsiness detection system using deep learning",2021.

[21] Rashi bhargava, ravina panchal, preeti verma, shivam sharma," drowsiness detection while driving using eye tracking",2020.

[22] N.radha, e. M. Malathy, r. Swathika, r. B. Jananie a. A. Silviya," drowsiness detection system using visual articulators" ,2021. **https://ieeexplore.ieee.org/document/9388534**

[23] M. A. Assari and m. Rahmati," driver drowsiness detection using face expression recognition",2011. **https://ieeexplore.ieee.org/abstract/document/6144162**

[24] S. Sathasivam, a. K. Mahamad, s. Saon, a. Sidek, m. M. Som and h. A. Ameen," drowsiness detection system using eye aspect ratio technique", 2020. **https://ieeexplore.ieee.org/abstract/document/9251035**

[25] Sinan kaplan, mehmet amac guvensan, ali gokhan yavuz, and yasin karal," driver behavior analysis for safe driving: a survey", 2015.

[26] Choi, in-ho, sung kyung hong, and yong-guk kim," real-time categorization of driver's gaze zone using the deep learning techniques",2016, doi: 10.1109/bigcomp.2016.7425813 **https://ieeexplore.ieee.org/document/7425813**

[27] Li, gang and lee, boon-leng and chung, wan-young," smartwatch based wearable eeg system for driver drowsiness detection",2015

[28] Li, lingling and chen, yangzhou and li, zhenlong," yawning detection for monitoring driver fatigue based on two cameras",2009

[29] R. N. Khushaba, s. Kodagoda, s. Lal, and g. Dissanayake," driver drowsiness classification using fuzzy wavelet-packet-based feature extraction algorithm",2011.

[30] Y. Liang, m. L. Reyes, and j. D. Lee," real-time detection of driver cognitive distraction using support vector machines",2007.

[31] Z. Li, S. E. Li, R. Li, B. Cheng, and J. Shi, "Online detection of driver fatigue using steering wheel angles for real driving conditions," Sensors, vol. 17, no. 3, p. 495, Mar. 2017.

[32] Y. Katyal, S. Alur, and S. Dwivedi, "Safe driving by detecting lane discipline and driver drowsiness," in Proc. IEEE Int. Conf. Adv. Commun., Control Comput. Technol., May 2014, pp. 1008–1012.

YawDD: A Yawning Detection dataset

Purpose

YawDD contains two video datasets of drivers with various facial characteristics, to be used for testing algorithms and models for mainly yawning detection, but also recognition and tracking of face and mouth. The videos are taken in real and varying illumination conditions.

- In the first dataset a camera is installed under the front mirror of the car. Each participant has three/four videos and each video contains different mouth conditions such as normal, talking/singing, and yawning. This dataset provides 322 videos consisting of both male and female drivers, with and without glasses/sunglasses, from different ethnicities, and in 3 different situations: 1- normal driving (no talking), 2- talking or singing while driving, and 3- yawning while driving.

- In the second dataset the camera is installed on the driver's dash. Each participant has a single video containing scenes with driving, driving while talking, and driving while yawning. This datast provides 29 videos consisting of both male and female drivers, with and without glasses/sunglasses, from different ethnicities.

Format and Available Data

The videos are in 640x480 24-bit true color (RGB) 30 frames per second AVI format without audio. The available data and their features are listed in table 1 and table 2. The total data size is about **5.1 Gigabytes** and can be accessed through ACM Multimedia Systems Conference Dataset Archive.

License and Usage

The videos are for non-commercial and research purposes only! No other usage is allowed! The videos are free and can be used in non-commercial and/or academic papers which study, design, and test algorithms and methods to detect face, facial features, and yawning. In addition, screen-shots of some (not all) videos can be used in such papers. Please check the *Allow Researchers to use picture in their paper* column in the above tables to see if you can use a screenshot of a particular video or not. **If for a particular video that column is "no", you are NOT allowed to use pictures from that video in your papers and publications.** Also, the following publication must be cited whenever making use of this dataset in any paper, publication, or report:

S. Abtahi, M. Omidyeganeh, S. Shirmohammadi, and B. Hariri, "YawDD: A Yawning Detection Dataset", Proc. ACM Multimedia Systems, Singapore, March 19 -21 2014.

How to Contribute Videos to the Dataset

We welcome the addition of more videos to the dataset. Please send an email to shervin@ieee.org if you would like to add more videos. Based on the format and quality of your videos, as well as the willingness of the volunteers in your videos to sign our licensing agreement, we might be able to add your videos to the dataset.

A PRELIMENERY PROJECT REPORT ON

DRIVER DROWSINESS DETECTION SYSTEM USING VISUAL FEATURES AND MACHINE LEARNING

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN PARTIAL THE FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE
OF

BACHELOR OF ENGINEERING
IN
COMPUTER ENGINEERING

SUBMITTED BY

Rutuja Sudhir Banginwar	Roll No:- **41303**
Imon Ashok Mandal	Roll No:- **41338**
Urmila Changdeo Matade	Roll No:- **41341**
Mayur Dnyandev Mote	Roll No:- **41343**

UNDER THE GUIDANCE OF

Dr. Prajakta A. Khadkikar

DEPARTMENT OF COMPUTER ENGINEERING

PUNE INSTITUTE OF COMPUTER TECHNOLOGY
DHANKAWADI, PUNE-43

SAVITRIBAI PHULE PUNE UNIVERSITY
2022-23

SAVITRIBAI PHULE PUNE UNIVERSITY
SCTR'S PUNE INSTITUTE OF COMPUTER TECHNOLOGY, PUNE
DEPARTMENT OF COMPUTER ENGINEERING
2022-2023

CERTIFICATE

This is to certify that the preliminary project report entitled
DRIVER DROWSINESS DETECTION SYSTEM USING VISUAL FEATURES AND MACHINE LEARNING
Submitted by

Rutuja Sudhir Banginwar	Roll No:- **41303**
Imon Ashok Mandal	Roll No:- **41338**
Urmila Changdeo Matade	Roll No:- **41341**
Mayur Dnyandev Mote	Roll No:- **41343**

are bonafide students of this institute and the work has been carried out by them under the supervision of **Dr. Prajakta A. Khadkikar** and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of the degree of **Bachelor of Engineering** (Computer Engineering).

Dr. Prajakta A. Khadkikar
Guide
Department of Computer Engineering

Dr. G.V. Kale
Head
Department of Computer Engineering

(Dr. S.T. Gandhe)
Principal
Pune Institute of Computer Technology

Place: Pune
Date:

ACKNOWLEDGEMENT

With great pleasure, we present the preliminary project report for the **"Driver Drowsiness Detection System Using Visual Features and Machine Learning"** project.

We would like to take this chance to express our gratitude to our guide **Dr. Prajakta A. Khadkikar** for providing us with all the support and advice we required. We are incredibly appreciative of her kind assistance. Her valuable suggestions were very helpful.

We are also grateful to **Dr. G.V. Kale**, Head of **Department of Computer Engineering, Pune Institute of Computer Technology** for her indispensable support and suggestions.

Last but not least, we want to express our gratitude to all of our professors, parents, and friends who have assisted us throughout the whole process.

Rutuja Sudhir Banginwar
Imon Ashok Mandal
Urmila Changdeo Matade
Mayur Dnyandev Mote
(B.E. Computer Engineering)

ABSTRACT

Over the past ten years, the number of motor vehicles has steadily grown in emerging nations. The reports of traffic accidents note that risky driving behaviors, such as drunken driving or fatigued driving, are responsible for most accidents. According to studies, sleepy driving is a contributing factor in 20% of all accidents. Drowsiness is a condition when the level of consciousness is lowered as a result of fatigue or lack of sleep, and it can make a driver fall asleep silently. Drowsy driving causes the driver to lose control of the vehicle, causing it to drift off the road, collide with an obstacle, or overturn. Several unfortunate accidents could have been prevented if the drowsy drivers had been informed in a timely manner. A number of drowsiness detection methods are available that monitor drivers' degrees of sleepiness while they are driving and warn them if they are not paying attention to the road. You may determine how drowsy someone is by observing their facial expressions, such as yawning, eye closure, and head movements. By using image processing algorithms, the system captures the videos and finds the driver's face in every frame. The system has the ability to recognise facial landmarks and computes the Eye Aspect Ratio (EAR) and Eye Closure Ratio (ECR) to calculate the driver's tiredness based on adaptive thresholding.

Contents

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

1.1 Motivation

India has the world's second largest road network, trailing only the United States, and the road network is critical to India's economic development and social well-being. As the number of highways and vehicles has increased, road accidents have become a major challenge for both the government and the general public. India leads the world in road fatalities, and the number of deaths from traffic accidents is rising. According to data, drowsiness is responsible for approximately 20% of all road accidents. As a result, it is critical to develop a low-cost system that can automatically detect driver drowsiness and alert the driver in a timely manner, eventually minimizing the number of fatalities.

1.2 Problem Definition

The project's objective is to recognise driver drowsiness and alert them when necessary so that accidents might be avoided. The goal of the research is to develop a model that can identify eye and mouth movement to ascertain whether or not a person is feeling sleepy. Driving will be safer thanks to the idea's immediate use in the automotive industry.

1.3 Technical Keywords

1) Machine Learning

2) Computer Vision

3) Drowsiness Detection

.

CHAPTER 2

LITERATURE SURVEY

Table 2.1: Literature survey

No.	Year	Author Name	Title Of Paper	Publication	Research Gap
1	2018	Fouzia, R. Roopalakshmi, Jayantkumar A. Rathod, Ashwitha S. Shetty, K. Supriya	Driver Drowsiness Detection System Based on Visual Features	IEEE	The inclusion of more features (like yawning detection) will further increase the accuracy of the system.
2	2018	Ashish Kumar, Rusha Patra	Driver Drowsiness Monitoring System using Visual Behaviour and Machine Learning	IEEE	Accuracy and performance can be further improved by using FLDA and SVM algorithm for classification.
3	2019	Shruti Mohanty, Shruti V Hegde, Supriya Prasad, J. Manikandan	Design of Real-time Drowsiness Detection System using Dlib detection	IEEE	People from different geographic locations have different facial structures, so the same threshold value of eye aspect ratio and mouth aspect ratio cannot be used for all people.

Table 2.2: Literature survey

No.	Year	Author Name	Title Of Paper	Publication	Research Gap
4	2020	Amin Azizi Suhaiman, Zazilah May, Noor A'in A.Rahman	Development of an intelligent drowsiness detection system for drivers using image processing technique	IEEE	Image Processing depends on intensity of light thus affects the results of detection.
5	2020	Ghole, Uzair Chavan, Pravin Gandhi, Siddharth Gawde, Rohit Fakir, Kausar.	Drowsiness Detection And Monitoring System	EDP Sciences	Haar cascade has a limitation of high false positive detection and less accurate than deep learning based technique.
6	2019	W. Deng and R. Wu	Real-Time Driver-Drowsiness Detection System Using Facial Features	IEEE	The accuracy of the eye closure recognition by CNN is poor. By using the angle of the eye to compensate for the CNN's limitations regarding eye closure recognition.

Table 2.3: Literature survey

No.	Year	Author Name	Title Of Paper	Publication	Research Gap
7	2021	Avigyan Sinha, Aneesh R P, Sarada K Gopal	Drowsiness Detection System Using Deep Learning	IEEE	Adding more features will increase the accuracy of the system.
8	2020	Rashi Bhargava, Ravina Panchal, Preeti Verma, Shivam Sharma	Drowsiness Detection While Driving Using Eye Tracking	IRJET	The accuracy of the system can be further improved by using more features like yawning detection.
9	2008	Mandalapu Saradadevi and Dr. Preeti Bajaj	Driver Fatigue Detection Using Mouth and Yawning Analysis	IJCSNS	More features like eye blink count and heart rate will increase the accuracy of the model.
10	2020	Jagendra Singh	Learning based Driver Drowsiness Detection Mode	IEEE	COCO sample and Haar technique has a less accuracy as compared to deep learning based technique.

CHAPTER 3

SOFTWARE REQUIREMENTS SPECIFICATION

3.1 Introduction

The Software Requirements Specification (SRS) will provide a detailed description of the requirements for the Driver Drowsiness Detection System. This SRS will allow a full understanding of what to expect from the engineered system. A clear understanding of the system and its functionality will enable the development of the right software for the end user. It will provide the foundation of the project. Based on the recorded specifications, the Driver Drowsiness System can be designed, constructed and tested.

3.1.1 Project Scope

The Drowsiness Detection System will be able to detect drowsiness in driver and warn the driver using an alarm. The model can be deployed by using components such as eye blink rate, yawning, etc. When all these parameters are applied it can improve the accuracy alot.

3.1.3 Assumptions and dependencies

1) Assumptions:

    a) Users have a high-resolution camera and a system that meets the minimum requirements.

2) Dependencies:

    a) The user should be familiar with machine learning concepts and how to use the application.

3.2 Functional requirements

The following functional specifications apply to the proposed system:-

3.2.1 Real-time video computation

The system should capture images at a rate of 10 frames per second.

3.2.2 Processing of frames

1) Captured images will be pre-processed, i.e. converted to grayscale.

2) This system will be responsible for searching the face and eye regions.

3.2.3 Capturing the number of frames in which the eyes are closed

The system determines whether the driver is drowsy based on whether these images exceed the threshold we set.

3.2.4 Generate output (alarm)

Based on the output, the system decides whether the driver is sleepy and generate an alarm.

3.3 Non-Functional requirements

3.3.1 Performance Requirement

1) The initial load time for the product must be minimal.
2) The client's hardware components will determine performance.

3.3.2 Safety Requirements

1) The requirements for the system must meets the customer's expectations in terms of the safe operation of the system.

3.3.3 Security Requirements

1) The system is activated when the ignition is turned on, driver will have no control of the system other than turning the alarm off.

3.3.4 Software Quality Attributes

1) Usability

   The system is easy to understand and very efficient to use.

2) Cost Effective

   The System should be cost-effective.

3) Reliability

   a) It shall be run properly at every time needed.
   b) Rate of Failure intensity should not be acceptable.

4) Availability

   System crashes should be minimum.

5) Maintainability

   a) The system shall respond in real time.

   b) Also responds to physical movements.

3.4 System Requirements

3.4.1 Software Requirements

1) Operating System: Windows 10/11
2) IDE: Jupyter Notebook or VS Code
3) Google Collab/Jupyter Notebook
4) Python
5) Python Libraries:
   - Numpy
   - Scipy
   - Dlib
   - OpenCV
   - Pandas
6) HTML, CSS, JS

3.4.2 Requirements Description

1) Python:
   A very basic form of the popular programming language Python is used for general programmable utilities. It is an effective and widely used programming language that is used for both small-scale and large-scale programming throughout the world. Python has programmable motorised boards. Different object-oriented, fundamental, utilitarian, and procedural programming perfection models are provided. The standard library is also broad and comprehensive. Python's advantageous properties will aid us in completing this task because of the following:

a) Python can run on any operating system .

b) Python's syntax is straightforward and easy to remember.

2) Numpy:
To work with arrays, one uses the Python module NumPy. Numerical Python is what it means. To replace arrays in Python, we have lists, however they take a long time to execute. Up to 50 times faster than conventional Python lists, NumPy strives to give an array object. NumPy's array object, referred to as ndarray, offers a number of support methods that make using ndarray relatively simple. In data science, where efficiency and resources are crucial, arrays are utilised frequently.

3) Pandas:
Pandas is based on two fundamental Python libraries: NumPy for mathematical operations and matplotlib for data display. By acting as a wrapper over both libraries, Pandas enables you to access many matplotlib and NumPy methods with less code. For instance, pandas'.plot() combines various matplotlib methods into a single method, enabling you to plot a create in several lines. Prior to the advent of pandas, the majority of analysts used Python for data collection and preparation before switching to a more specialised language, like R, for the remainder of their workflow. Pandas introduced Series, which have a list-like structure, and DataFrames, which have a table structure, two new types of data storage objects that simplify analytical processes and eliminate the need to switch tools.

4) Dlib:
The Dlib open source library, which includes a machine learning library and a number of software development tools, is one of the most robust and simple to use libraries available. In 2002, it was first made available. It is frequently utilised in numerous huge businesses, industries, and projects, among other things. Additionally, it contains a far larger variety of algorithms that play a bigger part in reality. Face recognition is the primary use of Dlib. Using the HOG (Histogram of Oriented Gradients) and CNN functions, they analyzed the object/face (Convolutional Neural Networks). In many modern applications, face recognition is employed extensively.

Steps for installation:

a) Install Python

b) Install CMake

c) Install Visual Studio Code.

d) Install cmake module.

e) Install dlib library.

f) The dlib library is installed successfully on your machine.

5) Open CV:
A large open-source library for image processing, machine learning, and computer vision is called OpenCV. Python, C++, Java, and other programming languages are among the many that OpenCV supports. It can analyse photos and movies to find faces, objects, and even human handwriting. The amount of weapons in your arsenal rises when it is integrated with different libraries like Numpy, which is a highly efficient library for numerical operations. Any operations that can be done in Numpy can be combined with OpenCV.

3.4.3 Hardware Requirements

a) Laptop/ Desktop : Intel i5 Generation Processor

b) RAM : 16GB

c) SSD : 500GB

d) Graphics Card

3.5 Analysis Model

Waterfall Software Development Life Cycle (SDLC) Model is used for the development of the system.

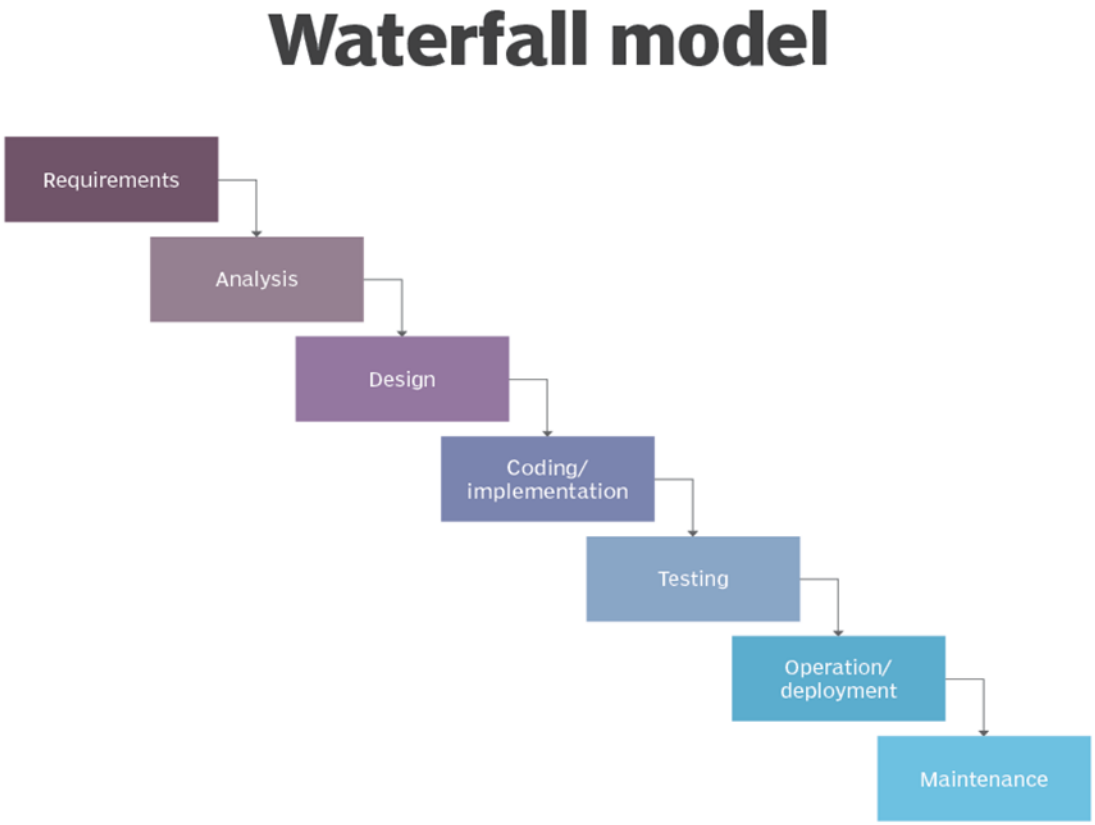


Figure 3.1: Waterfall Model for SDLC

According to the Waterfall paradigm, the successive phases are:-

3.5.1 Requirement Gathering and analysis

This stage involves gathering all potential system needs and documenting them in a requirement specification document.

3.5.2 System Design

This phase studies the initial phase's required specifications and prepares the system design. The system architecture as a whole is defined as a result of this system design, which also aids in determining the hardware and system requirements.

3.5.3 Implementation

The system is initially built in small programmes called units that are then combined in the following phase with input from the system design. Unit testing is the process of developing and testing each unit for functionality.

3.5.4 Integration and Testing

After each unit has undergone testing during the implementation phase, the entire system is merged. Following integration, the entire system is examined for errors and failures.

3.5.5 Deployment of system

The product is deployed in the customer environment or made available for purchase when functional and non-functional testing is complete.

3.5.6 Maintenance

A client environment is prone to a variety of issues. Patches are released to fix specific issues. Furthermore, updated versions of the product are released. Maintenance is carried out to effect these modifications in the consumer's environment.

3.6 System Implementation Plan

SR NO.	TASK	LABOUR HOUR/DAYS
1.	Topic Selection	2 Weeks
2.	Feasibility Study	1 Week
3.	Project Design	2 Weeks
4.	Develop Functional Specifications	10 Days
5.	Develop System Architecture	2 Weeks
6.	Develop Detailed Design Specifications	2 Weeks
7.	Data Collection and Environment Setup	2.5 Weeks
8.	Project Development	10 Weeks
9.	Perform Module Integration	3 Weeks
10.	Perform Testing	3 Weeks
11.	Post Project Review	1 Week

Table 3.1: Project Estimate

CHAPTER 4

SYSTEM DESIGN

4.1 Architecture diagram

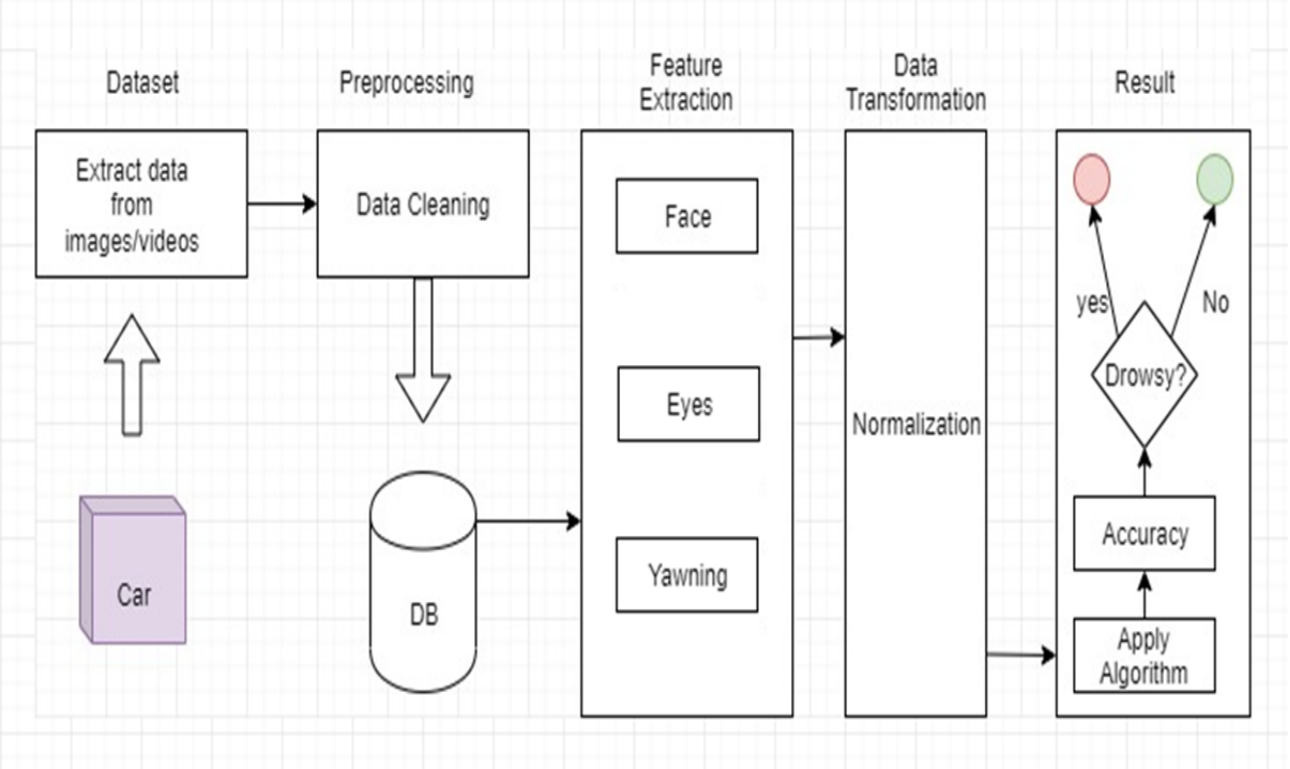


Figure 4.1: Architecture diagram

4.2 Data Flow Diagrams

4.2.1 Data Flow Diagram Level-0

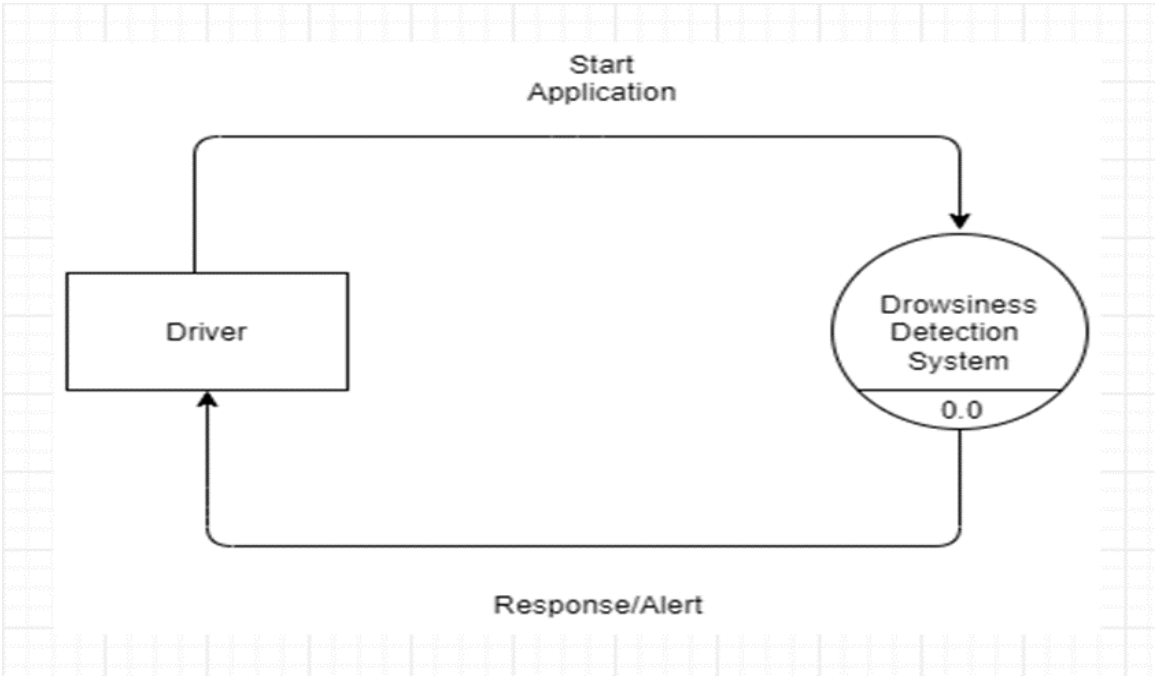


Figure 4.2: DFD level-0

4.2.2 Data Flow Diagram Level-1

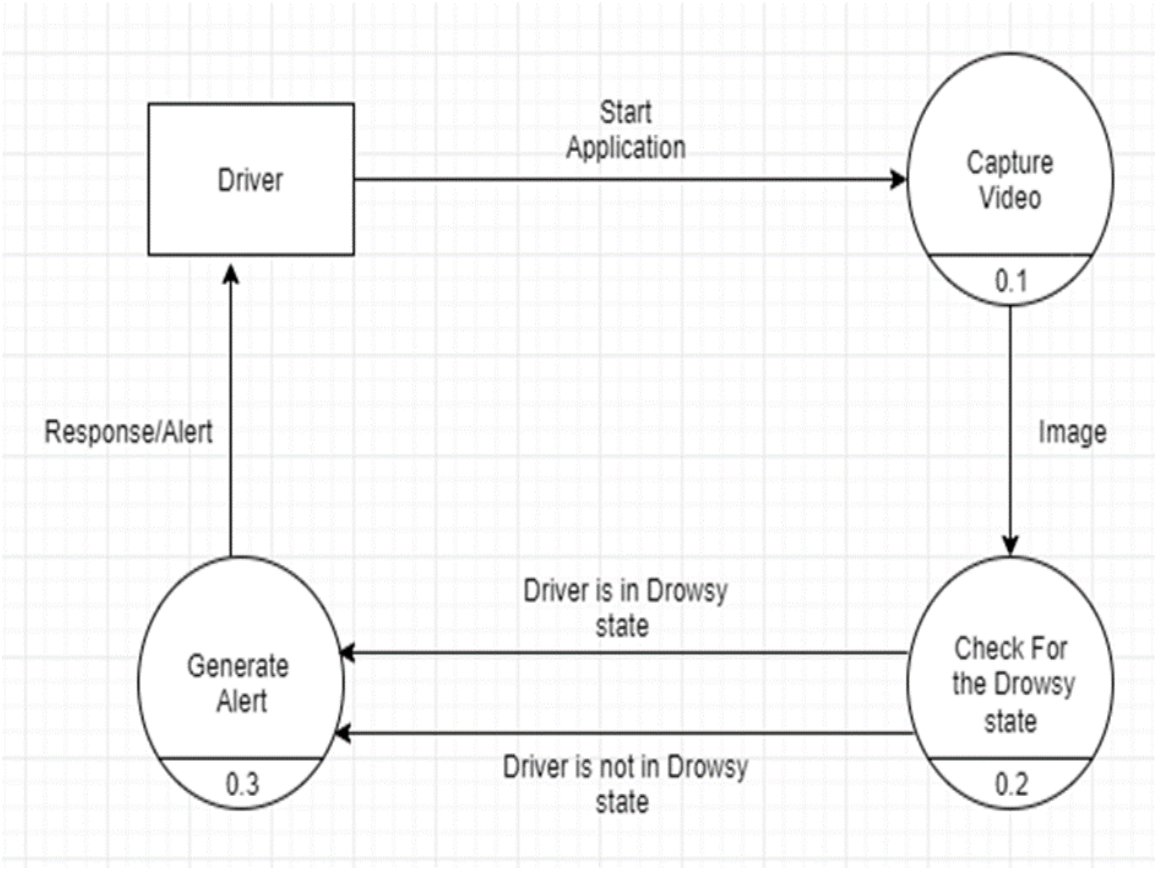


Figure 4.3: DFD level-1

4.2.2 Data Flow Diagram Level-2

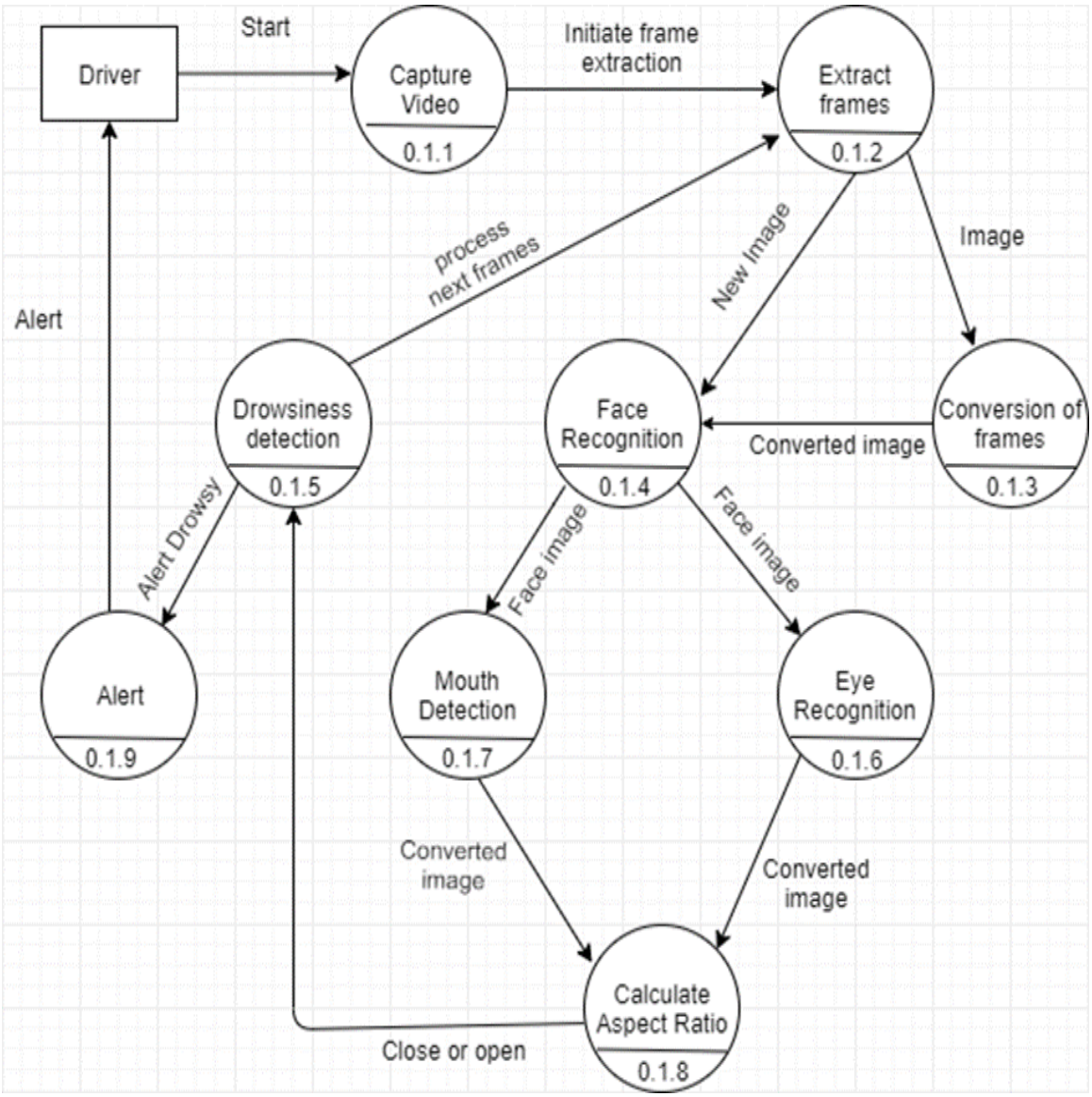


Figure 4.4: DFD level-2

4.3 Use Case Diagrams

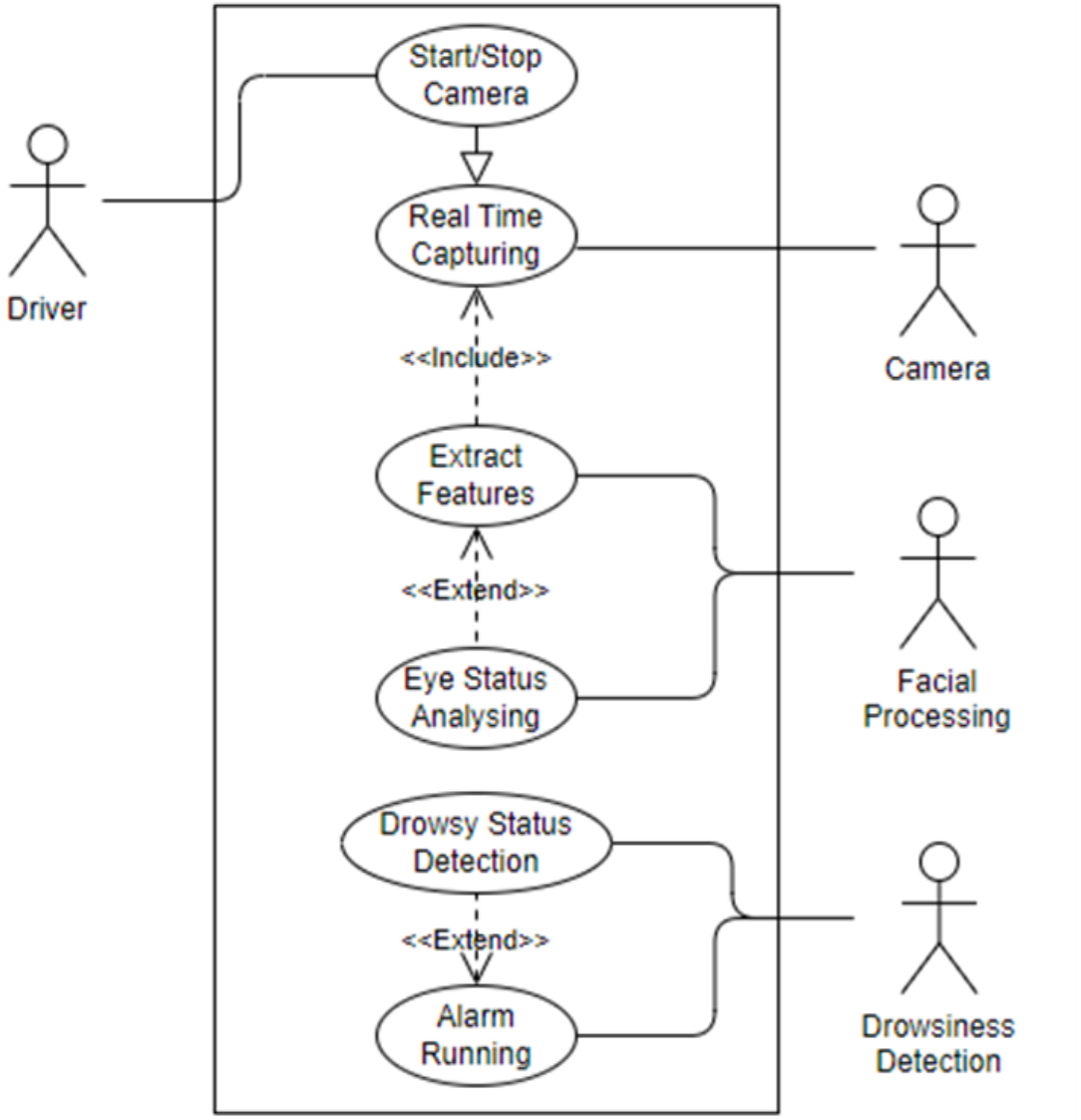


Figure 4.5: Use Case Diagram

4.4 Sequence diagram

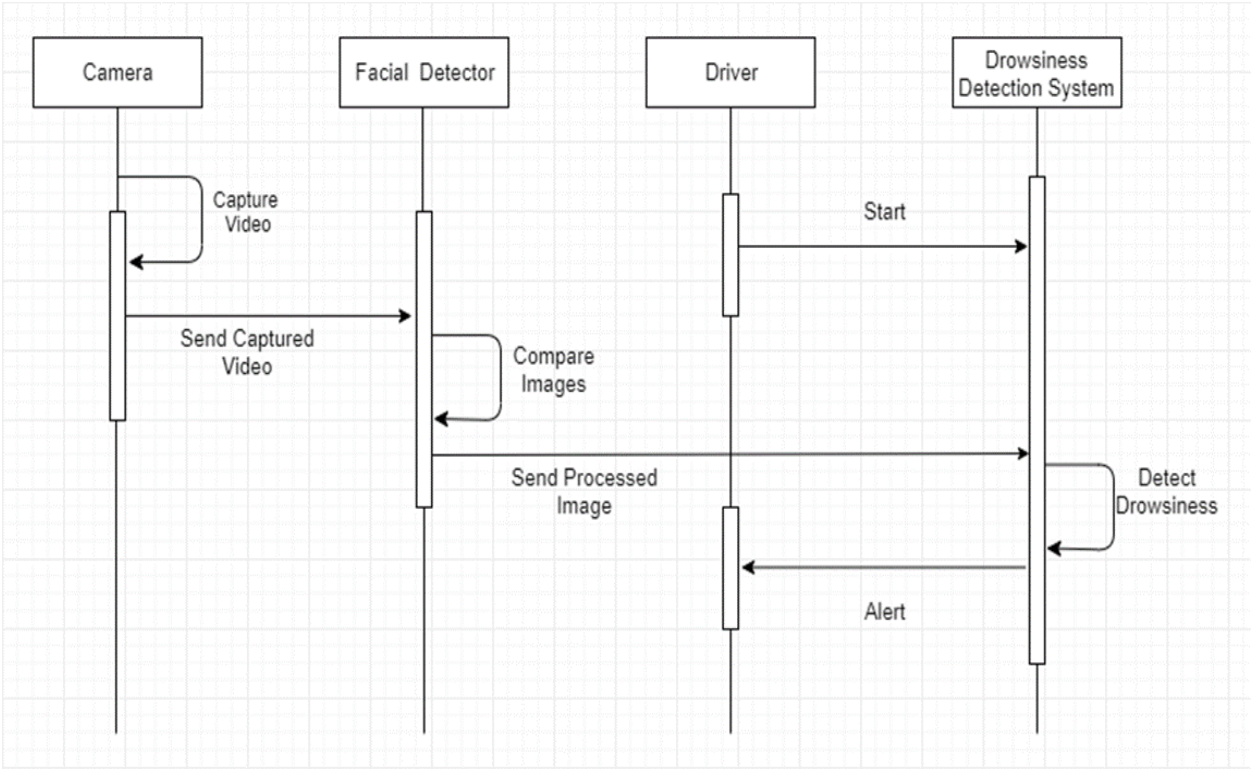


Figure 4.6: Sequence diagram

4.5 Algorithm

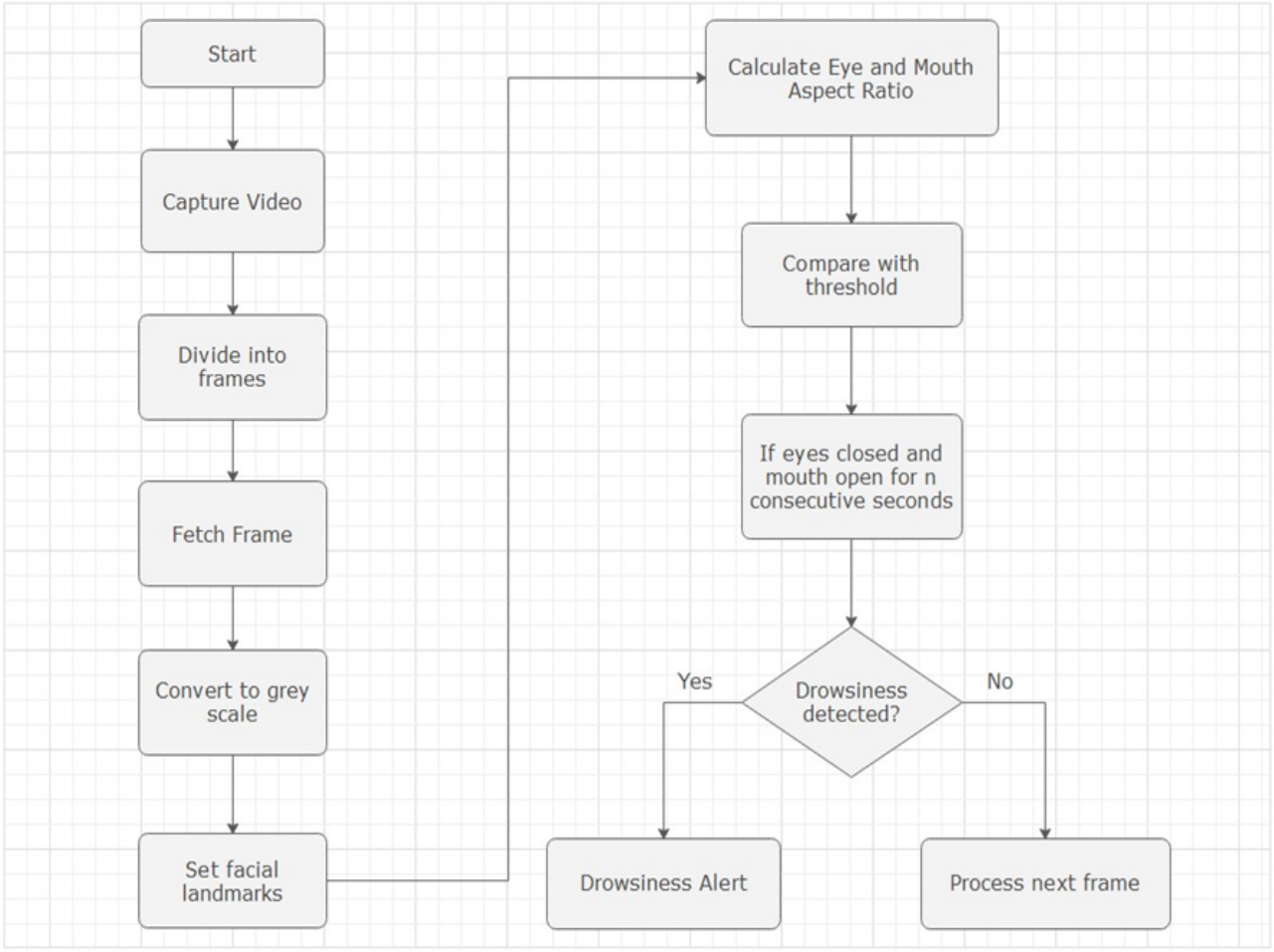


Figure 4.7: Algorithm

CHAPTER 5

OTHER SPECIFICATIONS

5.1 Advantages

There are number of advantages of drowsiness detection system which are as follows:

1) To avoid road accidents by alerting the driver at the appropriate time.
2) To detect fatigue and drowsiness while driving.
3) To examine mouth and eye activity in video images to recognise the signs of drowsiness.
4) To investigate the physical changes caused by fatigue and drowsiness.

5.2 Limitations

There are certain limitations to the Application which are as follows:

1) Dependency on hardware devices: The model's accuracy may be affected by the quality of the camera used to capture the video.
2) The weather and illumination conditions might have an impact on a model's performance.
3) Real Time output Generation: For real time output generation it will require high computations.

5.3 Applications

Applications of our system mentioned below:

1) It can be used in Commercial Vehicles/ Trucks, Cars, etc.
2) In online classes where students feel drowsy and inattentive during the class. Security guards cabin.
3) Operators in nuclear power plants where continuous monitoring is necessary.
4) Military applications where high intensity monitoring is needed.
5) Transportation business where almost daily accidents occur due to driver fatigue.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Conclusion and Future Work

The software is quick and efficient to identify drowsiness. The alarm sounds to warn the driver when the eyes are closed or the mouth is opened for n consecutive frames. In regular cars, driver safety can also be attained by installing a drowsiness detecting system. This method can both lower traffic accidents and provide tools for drowsy driving detection.

Future research may concentrate on the use of external factors for measuring fatigue, such as vehicle states, sleeping patterns, weather, mechanical data, etc. We can add more features, such as head movement and heart rate, to further increase the accuracy.

CHAPTER 7

BIBLIOGRAPHY

[1] Fouzia, R. Roopalakshmi, J. A. Rathod, A. S. Shetty and K. Supriya, "Driver Drowsiness Detection System Based on Visual Features," 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018, pp. 1344-1347, doi: 10.1109/ICICCT.2018.8473203. **https://ieeexplore.ieee.org/abstract/document/8473203**

[2] A. Kumar and R. Patra, "Driver drowsiness monitoring system using visual behaviour and machine learning," 2018 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE), 2018, pp. 339-344, doi: 10.1109/ISCAIE.2018.8405495. **https://ieeexplore.ieee.org/abstract/document/8405495**

[3] S. Mohanty, S. V. Hegde, S. Prasad and J. Manikandan, "Design of Real-time Drowsiness Detection System using Dlib," 2019 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE), 2019, pp. 1-4, doi: 10.1109/WIECON-ECE48653.2019.9019910. **https://ieeexplore.ieee.org/abstract/document/9019910**

[4] Mandalapu Saradadevi and Dr. Preeti Bajaj, "Driver Fatigue Detection Using Mouth and Yawning Analysis", IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.6, June 2008. **http://paper.ijcsns.org/07**$_{b}$*ook*/200806/20080624.*pdf*

[5] A. A. Suhaiman, Z. May and N. A. A.Rahman, "Development of an intelligent drowsiness detection system for drivers using image processing technique," 2020 IEEE Student Conference on Research and Development (SCOReD), 2020, pp. 233-236, doi: 10.1109/SCOReD50371.2020.9250948. **https://ieeexplore.ieee.org/abstract/document/9250948**

[6] Ghole, Uzair & Chavan, Pravin & Gandhi, Siddharth & Gawde, Rohit Fakir, Kausar. (2020). "Drowsiness Detection And Monitoring System". ITM Web of Conferences. 32. 03045. 10.1051/itmconf/20203203045. **https://doi.org/10.1051/itmconf/20203203045**

[7] W. Deng and R. Wu, "Real-Time Driver-Drowsiness Detection System Using Facial Features," in IEEE Access, vol. 7, pp. 118727-118738, 2019, doi: 10.1109/ACCESS.2019.2936663. **https://ieeexplore.ieee.org/abstract/document/8808931**

[8] JJ. Singh, "Learning based Driver Drowsiness Detection Model," 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), 2020, pp. 698-701, doi: 10.1109/ICISS49785.2020.9316131. **https://ieeexplore.ieee.org/document/9316131**

[9] U. Budak, V. Bajaj, Y. Akbulut, O. Atila and A. Sengur, "An Effective Hybrid Model for EEG-Based Drowsiness Detection," in IEEE Sensors Journal, vol. 19, no. 17, pp. 7624-7631, 1 Sept.1, 2019, doi: 10.1109/JSEN.2019.2917850. **https://ieeexplore.ieee.org/abstract/document/8718312**

[10] J. -J. Yan, H. -H. Kuo, Y. -F. Lin and T. -L. Liao, "Real-Time Driver Drowsiness Detection System Based on PERCLOS and Grayscale Image Processing," 2016 International Symposium on Computer, Consumer and Control (IS3C), 2016, pp. 243-246, doi: 10.1109/IS3C.2016.72. **https://ieeexplore.ieee.org/abstract/document/7545182**

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 25, 2019, accepted April 26, 2019, date of publication May 1, 2019, date of current version May 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2914373

A Survey on State-of-the-Art Drowsiness Detection Techniques

MUHAMMAD RAMZAN[1,2], HIKMAT ULLAH KHAN[3], SHAHID MAHMOOD AWAN[1], AMINA ISMAIL[3], MAHWISH ILYAS[2], AND AHSAN MAHMOOD[4]

[1]School of Systems and Technology, University of Management and Technology, Lahore 54782, Pakistan
[2]Department of Computer Science and Information Technology, University of Sargodha, Sargodha 40100, Pakistan
[3]Department of Computer Science, COMSATS University Islamabad at Wah Campus, Wah Cantt 47040, Pakistan
[4]Department of Computer Science, COMSATS University Islamabad at Attock Campus, Attock 43600, Pakistan

Corresponding author: Hikmat Ullah Khan (hikmat.ullah@ciitwah.edu.pk)

ABSTRACT Drowsiness or fatigue is a major cause of road accidents and has significant implications for road safety. Several deadly accidents can be prevented if the drowsy drivers are warned in time. A variety of drowsiness detection methods exist that monitor the drivers' drowsiness state while driving and alarm the drivers if they are not concentrating on driving. The relevant features can be extracted from facial expressions such as yawning, eye closure, and head movements for inferring the level of drowsiness. The biological condition of the drivers' body, as well as vehicle behavior, is analyzed for driver drowsiness detection. This paper presents a comprehensive analysis of the existing methods of driver drowsiness detection and presents a detailed analysis of widely used classification techniques in this regard. First, in this paper, we classify the existing techniques into three categories: behavioral, vehicular, and physiological parameters-based techniques. Second, top supervised learning techniques used for drowsiness detection are reviewed. Third, the pros and cons and comparative study of the diverse method are discussed. In addition, the research frameworks are elaborated in diagrams for better understanding. In the end, overall research findings based on the extensive survey are concluded which will help young researchers for finding potential future work in the relevant field.

INDEX TERMS Digital image processing, driver drowsiness, sensors, fatigue detection, supervised learning, classification, support vector machine (SVM).

I. INTRODUCTION

Drowsiness or fatigue is one of the main factors that threaten the road safety and causes the severe injuries, deaths and economical losses. The increased drowsiness deteriorates the driving performance. Lack of alertness, generated by the unconscious transition from wakefulness to sleep, leads to several serious road accidents. The U. S. National Highway Traffic Safety Administration (NHTSA)[1] reports that drowsy driving resulted in almost 100,000 road accidents and more than 1,500 deaths per year. A driver's fatigue can have multiple causes such as lack of sleep, long journey, restlessness, alcohol consumption and mental pressure. Each of which can lead to serious disaster. Nowadays, road rage is in the multiples of the past, which causes stress on drivers. Therefore, previous transportation system is not enough to handle these hazards on roads. Thus, by embedding the automatic fatigue detection systems into vehicles, several deadly accidents can be prevented. The drowsiness detection system continuously analyzes the drivers' attention level and alerts the driver before the arrival of any serious threat to road safety.

Due to the hazards that fatigue create on the roads, researchers have developed various methods to detect driver drowsiness and each technique has its own benefits and limitations. To conduct a valuable review of Drowsiness Detection Techniques (DDT) and appropriate classification methods, we build search strings to gather relevant information. We keep our search focused on publications of well reputed journals and conferences. We established a multi-stage selection criteria and assessment procedure.

The associate editor coordinating the review of this manuscript and approving it for publication was Auday A. H. Mohamad.

[1]https://crashstats.nhtsa.dot.gov/Api/Public/Publication/811754.Last Access:2018-11-14

Based on devising criteria, 41 research papers are filtered out from a detail search of 1020 research papers. We found a great rising in the trends of drowsiness detection systems, but still there is space for further improvement in present measures of drowsiness detection.

In this literature review, we (1) classify the existing models into three major categories and then review each model in chronological order sharing its novelty, main features and limitations (2) discuss Hybrid approaches (3) explore, top supervised learning techniques used for drowsiness detection have been explored (4) present the pros and cons and provide a comparative study of techniques are discussed (5) elaborate the research methods in the form of frameworks for better understanding.

This paper is subdivided into seven main sections. Section 2 describes the research methodology for performing effective analysis of drowsiness techniques, detailed analysis and evaluation of selected papers is discussed. In section 3, Detailed review of drowsiness detection techniques is presented in the form of tables. In section 4, the comparative study of DDT is presented. Section 5 illustrated the hybrid approaches of DDT. Classification methods used for drowsiness detection are listed in Section 6. A comparative study of classification is discussed in section 7 and the conclusion of this study are discussed in the end.

II. RESEARCH METHODOLOGY

The purpose of this systematic review paper is recognition and categorization of the best possible techniques, measures, tools and classification methods for drivers' drowsiness detection. The systematic reviews help to recognize what we know in the concerned domain. All the data gathered from primary studies is categorized. Once the systematic review of empirical studies is done, we gather relevant information and identify the research gaps in the existing research studies [1]. The population of systematic review consists of research papers relevant to drowsiness detection.

A. DATA ACQUISITION AND SELECTION

A systematic and well organized, search is conducted to extract meaningful and relevant information from the buckets of data. Research papers, case students, NHTSA on road accidents and reference lists of related publication were examined in detail. The websites containing information regarding to road safety, dangers of driver's fatigue, reasons of fatigue, and techniques of drowsiness detection are all searched. Keywords used to search information relevant to drowsiness are listed in TABLE 1. The search string is as follows

- (Drows* OR Fatigu*) AND (Biological* OR physiological*)
- (Drows* OR Fatigu*) AND (Vehicle * OR Automobile*)
- (Drows* OR Fatigu*) AND (Behavioral*)
- (Drows* OR Fatigu*) AND (Classif**)
- (Drows* OR Fatigu*) AND (SVM*)

TABLE 1. Description of search words.

Search Word	Set of keywords
Drows*	Drowsiness, Drowsiness detection, Drowsiness technique, drowsy driver
Fatigu*	Fatigues driver, Fatigue detection, Fatigue detection techniques
Biological*	Biological parameters, biological-measures, biological conditions
Physiological*	Physiological parameters, Physiological measures, Physiological conditions
Vehicle*	Vehicular measures, vehicular movements
Automobile *	Automobile measures, Automobile movements
Behavioral*	Behavioral parameters, behavioral measure
Classif*	Classifiers, Classification techniques, classification methods
SVM*	SVM classifier, support vector machine
CNN*	CNN classifier, Convolutional Neural Network
HMM*	HMM classifier, Hidden Markov model

- (Drows* OR Fatigu*) AND (CNN*)
- (Drows* OR Fatigu*) AND (HMM*)

We perform our initial study on search engines such IEEE explore, ACM, Springer, Google Scholar, IJCS and MDPI to extract information relevant to drowsiness detection. We found 1020 research papers in primary search.

The initial search procedure produces 1020 research papers; from those we have selected 105 papers based on title relevant to our study. Abstract of selected papers is examined which lead to extraction of further 74 research papers. Then these extracted papers are studied thoroughly, and 41 research papers are filtered out as our primary study. The complete selection process is illustrated in Figure 1.

Thorough study of full research papers seeks the answers of certain quality control questions. Current systematic research follows the quality assessment questions like a) Is the paper relevant to the research domain? and b) are the papers published in well reputed journal or conference? The detailed analysis of drowsiness detection systems is categorized based on drowsiness measures and classification methods. A complete details of existing drowsiness systems are summarized in tabular form. Answers of research questions are generated as well as pros and cons found in studies are evaluated, suggestions and conclusions are drawn.

III. DROWSINESS DETECTION TECHNIQUES

A detailed review of mentioned drowsiness detection techniques and their pros and cons are discussed in this section. Furthermore, the comparative analysis of such techniques is performed on different types of driving conditions The Driver Drowsiness detection system continuously monitors the drivers' physical behavior, vehicular movement pattern or environmental conditions based on the technique being used.

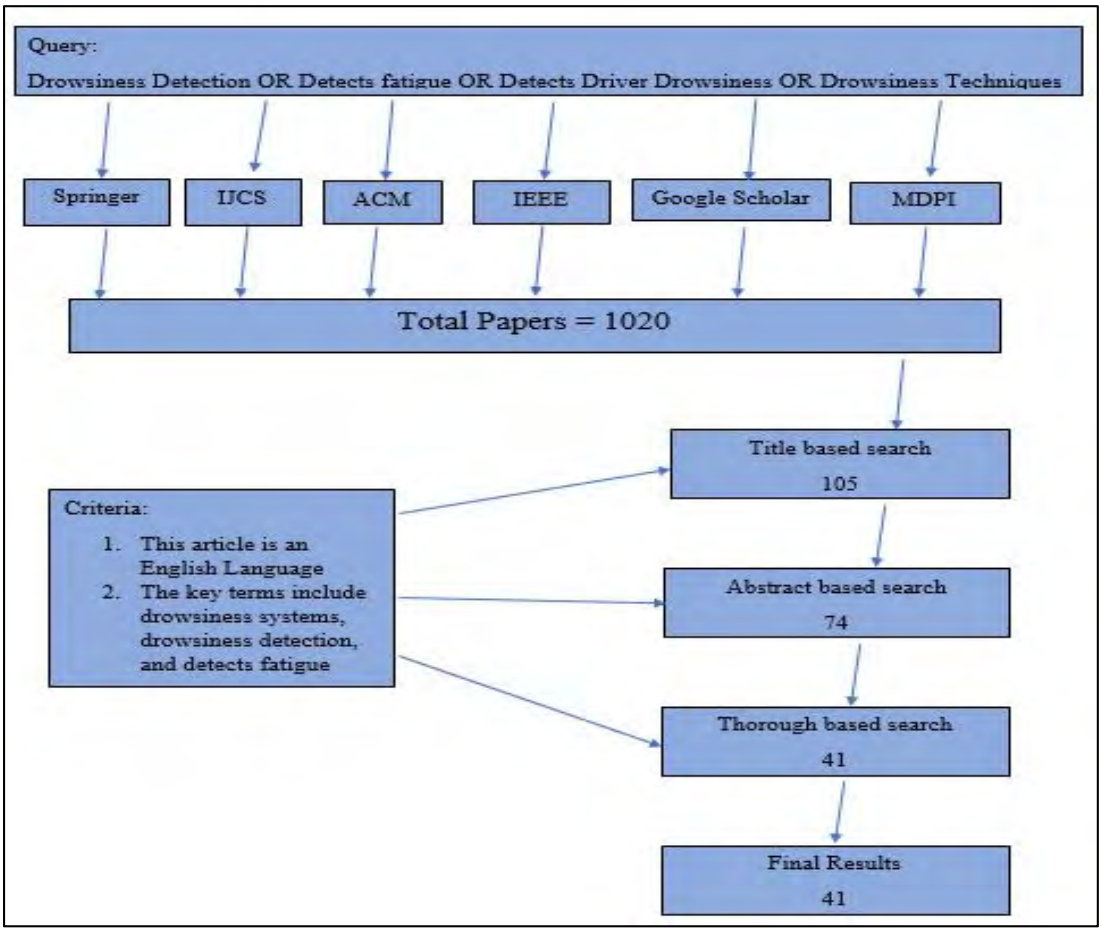


FIGURE 1. The research papers selection process.

Drowsiness detection methods are generally classified into three main categories:

1) Behavioral parameter-based techniques
2) Vehicular parameters-based techniques
3) Physiological parameters-based techniques

Figure 2 illustrates the basic architecture of Drowsiness Detection Techniques

A. BEHAVIORAL PARAMETER-BASED DDT

Behavioral parameters are non-invasive measures for drowsiness detection. These techniques measure drivers' fatigue through behavioral parameters of driver such as eye closure ratio, eye blinking, head position, facial expressions, and yawning. The Percentage of eye Closures (PERCLOS) is one of the most frequent used metrics in drowsiness detection based on eye state observation. PERCLOS is the ratio of eye closure over a period, and then on the result of PERCLOS, eyes are referred as open or closed. Yawning based detection systems analyze the variations in the geometric shape of the mouth of drowsy driver such as wider opening of mouth, lip position, etc. Behavioral based techniques used cameras and computer vision techniques to extract behavioral features. The general framework of process in behavioral pattern-based drowsiness detection techniques is presented in *Figure 3*.

A list of drowsiness detection system based on behavioral patterns is presented in Table 2.

The problems associated with behavioral measures are environmental factors, such as the illumination, brightness, and road conditions influence the credibility and accuracy of measurement [2].

1) EYE TRACKING AND DYNAMIC TEMPLATE MATCHING

To avoid road accidents, real time driver fatigue detection system based on vision [3] is proposed. Firstly, system detects the face of driver from the input images using HSI color model. Secondly, Sobel edge operator is used to locate the eyes positions and gets the images of eye as the dynamic template for the tracking of eye. Then the obtained images are converted to HSI color model to decide that whether the eyes are close or open to judge the drowsiness of driver. The experiments use four test videos for the tracking of eyes and face detection. The proposed system is compared with the labeled data which is annotated by the experts. The average

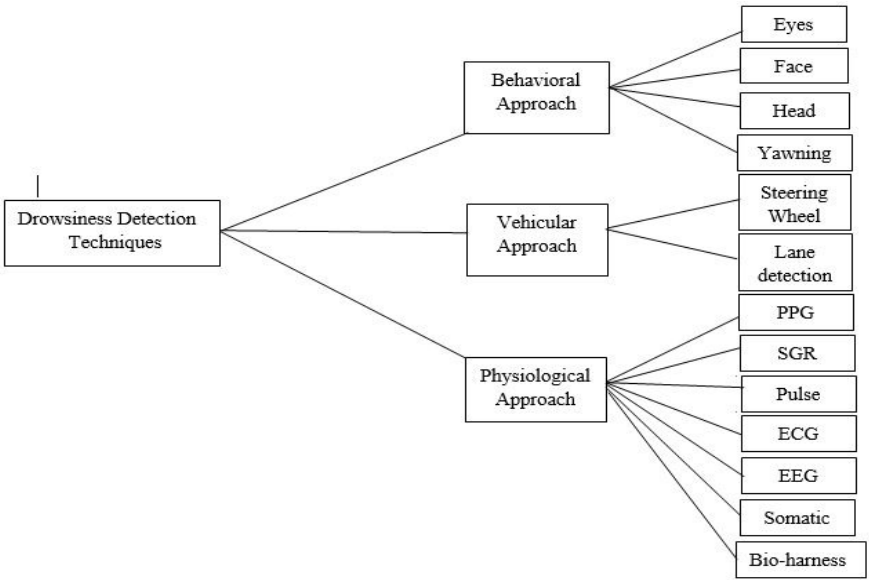


FIGURE 2. Architecture of drowsiness detection techniques.

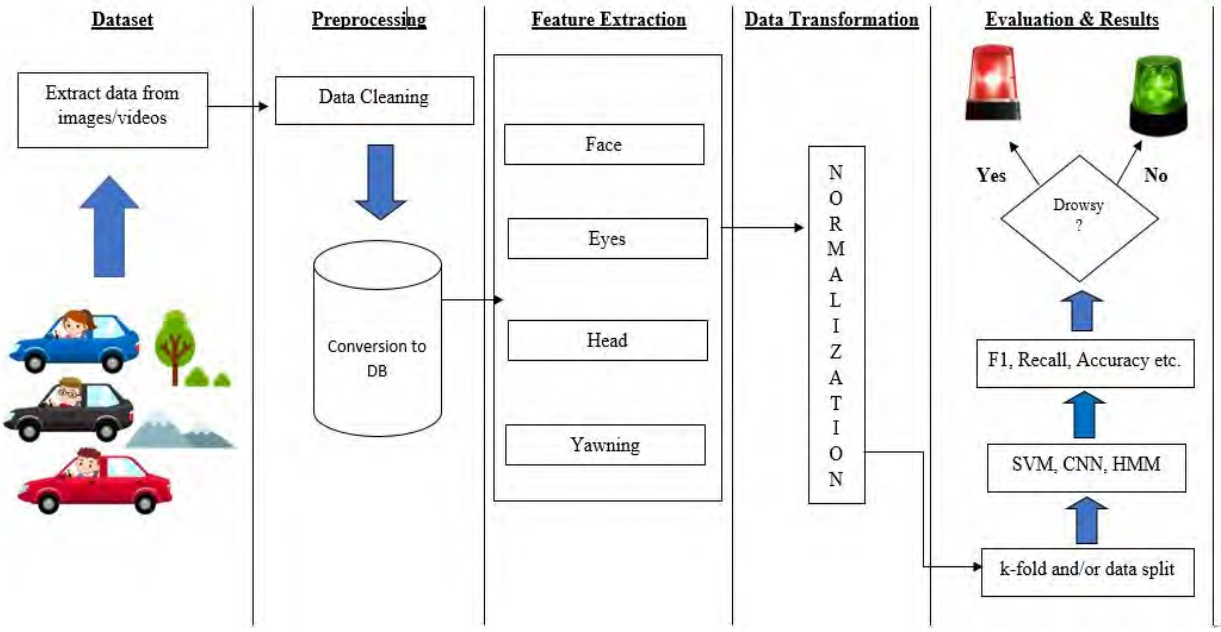


FIGURE 3. A general framework of Behavioral Pattern-based techniques.

correct rate of proposed system reaches up to 99.01 % and the precision to 88.9 %.

2) MOUTH AND YAWNING ANALYSIS

Fatigue is the major reason for road accidents. To avoid the issue, Saradadevi and Bajaj [4] proposed the driver fatigue detection system based on mouth and yawning analysis. Firstly, the system locates and tracks the mouth of a driver using cascade of classifier training and mouth detection from the input images. Then, the images of mouth and yawning are trained using SVM. In the end, SVM is used to classify the regions of mouth to detects the yawning and alerts the fatigue. For experiment, authors collect some videos and select 20 yawning images and more than 100 normal videos as dataset. The results show that the proposed system gives better results as compared to the system using geometric features. The proposed system detects yawning, alerts the fatigue earlier and facilitates to make the driver safe.

3) FACIAL EXPRESSIONS METHOD

Laboratory condition using Finite Element Analysis is used by the researchers which is a complex system that contains

TABLE 2. Behavioral-parameter based drowsiness detection systems.

Ref	Behavioral Features	Method & Classifier	Description	Efficiency
[3]	Eye tracking	HSI color model, Sobel edge detection method	HSI color model is independent of brightness. Sobel edge detection to locate eyes.	88.9
[4]	Yawning	Cascade classifier. SVM, Viola-Jones algorithm	applied SVM to train mouth and yawing images. Detects fatigue using cascade.	Nil
[5]	Face components (eyebrow raising, eye closing, yawning)	Background subtraction method, Horizontal projection technique	Used infrared light-based hardware to monitor the eye closing and eyebrows rising to detect drowsiness levels of driver.	Nil
[6]	Yawning	SVM, CHT	Used three steps face extraction, mouth region selection and wide, mouth open detection using SVM. CHT is applied to the results proposed wide, mouth open detector.	98%
[7]	Head movement, eye closeness duration	Viola Jones algorithm, WNC classification	Detects drowsiness based on eye blinking and head pose. A novel WNC based supervised method is proposed.	88.57%
[8]	Eye blinking and yawning	Haarcascade classifier, Active Contour Method, Viola Jones Algorithm	Used eye blinking duration and yawning to detect drowsiness. The System give better results if drivers are without glasses and moustache.	Nil
[9]	Eye state and blinking rate	Viola-Jones method & CART method, Binary cascade classifiers	Used cascade object identifier from vision tool box of mat- lab to eye region. Average drowsiness is calculated, and closed eye is regarded as zero.	90%
[10]	Eye state	Viola-Jones technique, Standard Ada-Boost (Adaptive Boosting) training method & PERCLOS method	Provides user friendly GUI. System requires compact hardware to execute on mid– range, microprocessor or it may be implemented on smart phone having hardware and software requirements.	95%
[12]	Eye blinking	Viola Jones algorithm, Adaboos computational approach, Haar cascade classifier, Luminosity Algorithm & Harris Corner Detector	Non-intrusive and work well in real time. Requires clear visibility of the eye. In poor lightening or sun glasses, system become unable to detect eye region and fails.	94% (in good lightening condition)
[13]	Head tilting and eye blinking frequency	Haar Cascade Classifier	Detection using eye blinking rate and head level, but efficiency declines in poor lightening and with sun glasses.	99.59

the database of facial expression as a template and detect the drowsiness on the basis of results from database. Similarly, Assari and Rahmati [5] present the hardware-based Driver Drowsiness Detection system based on facial expressions. The hardware system uses infrared light as it has giving many benefits like ease of use, independent of lightning conditions of environment. The system firstly uses the technique of background subtraction to determines the face region from the input images. Then using horizontal projection and template matching, facial expressions are obtained. After that in the tracking phase, elements found earlier are followed up using template matching and then investigates the incidence of sleepiness using the determination of facial states from the changes of the facial components. Changing in the three main elements such as eye brow rising, yawning and eye closure for the certain period are taken as the initial indications for drowsiness and the system generates the alert. The experiment is performed in the real driving scenario. For testing, images are acquired by the webcam under different conditions of lighting and from different people. The results investigate that the system produces appropriate response in the presence of beard or glasses and mustache on the face of driver.

4) YAWNING EXTRACTION METHOD

Fatigue or drowsiness is the major reason for road accidents. To prevent the issue, Alioua *et al.* [6] proposed the efficient system for monitoring the driver fatigue using Yawning extraction. Firstly, face region is obtained from the images using Support Vector Machine (SVM) technique to reduce the required cost. The proposed method is used to localize the mouth, edge detection technique is used to detects facial edges, then compute vertical projection on the lower half face to detect the right and left region boundaries and then compute the horizontal projection on the resulting region to detect the upper and lower limit of mouth and mouth localized region is obtained. Finally, to detect the yawning, Circular Hough Transform (CHT) is executed on the images of mouth region to identify the wide-open mouth. If the system finds notable number of continuous frames where the mouth is widely open, system generates the alert. The results are compared with the other edge detectors like Sobel, Prewitt, Roberts, Canny. The experiment uses 6 videos representing real driving conditions and results are presented in the form of confusion matrix. The proposed method achieves 98% accuracy and outperforms all other edge detection techniques.

5) EYE CLOSURE AND HEAD POSTURES METHOD

Teyeb *et al.* [7] proposed the Drowsy Driver Detection using Eye Closure and Head postures. Firstly, video is captured using webcam and for each frame of video, following operations are performed. To detect the ROI (face and eyes), viola-jones method is used. The face is partitioned in to three areas and the top one presenting the aye area is browsed by the Haar classifier. Then to detect the eye state, Wavelet Network based on neural network is used to train the images

then the coefficients learning images is compared with the coefficients of the testing images and tells which class it belongs. When the closed eye is identified in the frames then the eye closure duration is calculated, if the value exceeds the pre-defined time then the drowsiness state is detected. Then the developed system estimates the head movements which are: left, right, forward, backward inclination and left or right rotation. The captured video is segmented into frames and extract the images of head and determines the coordinates of image. Then the images are compared to determine the inclined state of head and same case with other head postures. Finally, the system combines the eye closure duration and head posture estimation to measure the drowsiness. To evaluate the system, experiment is performed on 10 volunteers in various situations. And results show that the systems achieve the accuracy of 80%.

6) REAL TIME ANALYSIS USING EYE AND YAWNING

Kuamr *et al.* [8] proposed the real time analysis of Driver Fatigue Detection using behavioral measures and gestures like eye blink, head movement and yawning to identify the drivers' state. The basic purpose of the proposed method is to detect the close eye and open mouth simultaneously and generates an alarm on positive detection. The system firstly captures the real time video using the camera mounted in front of the driver. Then the frames of captured video are used to detect the face and eyes by applying the viola-jones method, with the training set of face and eyes provided in OpenCV. Small rectangle is drawn around the center of eye and matrix is created that shows that the Region of Interest (ROI) that is eyes used in the next step. Since the both eyes blink at the same time that's why only the right eye is examined to detect the close eye state. If the eye is closed for certain amount of time it will be considered as closed eye. To determine the eye state, firstly the eye ball color is acquired by sampling the RGB components on the center of eye pixel. Then the absolute thresholding is done on the eye ROI based on eye ball color and intensity map is obtained on Y-axis that show the distribution of pixels on y-axis. Which gives the height of eye ball and compared that value with threshold value which is 4 to distinguish the open and close eye. After that, if the eye blink is detected in each frame it will be considered as 1 and stored in the buffer and after the 100 frames, eye blinking rate is calculated. Then to detect the yawning motion of the mouth, contour finding algorithm is used to measure the size of mouth. If the height is greater than the certain threshold. It means person is taking yawning. To evaluate the performance of the proposed system, system has been measured under different conditions like persons with glasses, without glasses, with moustache and without moustache for 20 days in different timings. The performs best when the drivers are without glasses and moustache.

7) EYE BLINK DETECTION METHOD

Ahmad and Borolie [9] proposed the Driver Drowsiness System based on non-intrusive machine-based concepts. The system consists of a web camera which is placed in front of the driver. Online videos as well as saved videos for simulation purposed are considered. Firstly, camera records the facial expressions and head movements of the driver. Then the video is converted into frames and each frame is processed one by one. Face is detected from frames using Viola-jones algorithm. Then the required features like eyes, mouth and head from face are extracted using cascade classifier. Region of interest on face is indicated by rectangles. Here the main attribute of detecting drowsiness is eyes blinking, varies from 12 to19 per minute normally and indicates the drowsiness if the frequency is less than the normal range. Instead of calculating eye blinking, average drowsiness is calculated. The detected eye is equivalent to zero (closed eye) and non-zero values are indicated as partially or fully open eyes. The equation (1) is used to calculate the average.

$$\%d = \frac{No.\ of\ closed\ eyes\ found}{no.\ of\ frames} * 100 \tag{1}$$

If the value is more than the set threshold value, then system generates the alarm to alert the driver. Moreover, yawning is also considered to generate the alert. Online and offline are videos are used for experiment which are performed on two different systems. The results show that the system achieves the efficiency up to 90%.

8) EYE TRACKING SYSTEM

Nguyen *et al.* [10] proposed an eye tracking system to detect drowsiness. The system includes camera, simple alarm and the laptop having developed (proposed) software and provides user-friendly GUI. Web cam is connected to laptop through USB port. Firstly, the system receives images from the webcam, system adjust the brightness and contrast and converts them into frames. Secondly, to detect the face region to locates the eye region, top-down model approach is used. If the face is not found in the input images, the system continues to take the images from the webcam until the face is found. After face detection, eye region is extracted using viola-jones algorithm and Ada-Boost (Adaptive Boosting) [11]. Algorithm firstly extracts face features like eyes and nose when Haar-like features are applied on the sub-image and Ada-Boost classifier is used for training of features. The system uses the popular method of PERCLOS for drowsiness detection. The fatigue level of driver S is calculated as S=H/L, where H is the height and L is the length of drivers' eye. Each frame of input video is categorized using the measured value of S. PERCLOS is measured as given in Equation (2).

$$PERCLOS = \frac{No.\ frames\ of\ closed\ eyes}{3\ \text{min}\ interval\ of\ all\ frame - blinking\ time} \tag{2}$$

0.15 is used as the highest drowsiness level for PERCLOS alarm. When the level of drowsiness reaches to severe, system activates the alarm to alert the driver. The experimental results show that it works in all lighting conditions.

TABLE 3. Vehicular parameter-based drowsiness detection systems.

Ref	Vehicular features	Method & Classifier	Description	Accuracy
[3]	Lane detection, eye blinking duration	Hough Transformation, Voila Jones method, Otsu thresholding, Canny edge detection, Circle detection Hough Transform	Detects lane changing pattern and eye closeness to detect drivers' drowsiness level. Works good in low lightening conditions as well.	Nil
[17]	Steering wheel behavior	Temporal detection window	Uses novel approach of time series analysis of steering wheel angular velocity to detect drowsiness.	Nil
[18]	Steering wheel angle	SWA-based fatigue detection method, binary classifier, DTW & ApEn based Adaptive Piecewise Linear Approximation	Detects driver drowsiness in real time using SWA data collected from sensors.	78.01%
[19]	Steering wheel angle, yaw angles	Approximate entropy (ApEn) features, Back propagation neural network classifier	Detects driver drowsiness using steering wheel and yaw angles in real time driving.	88.02%

9) EYE BLINK MONITORING METHOD

Drowsy or sleepy driver is the main reason for road accidents. To handle the problem, Rahman *et al.* [12] proposed an eye blinking based monitoring method to determine the drowsiness of driver. Firstly, receives the video from capturing device and converted into frames. Face region is detected from the frames using voila-jones technique. After the identification of face, ROI is set to face region, Viola-Jones cascade classifier technique is again applied to this region to eyes detection. Cascade classifier uses Haar-like features to detect eyes. Both eye regions are extracted for further processing. Then the proposed Eye-blink detection technique is applied. The method firstly converts the color eye image into grayscale using Luminosity algorithm. Then to detect the two-upper eye corner and one lower eye lid points, Harris corner detector is used. After the points identification, mid-point value between the upper two corner points is calculated. Let (x_1, y_1) be the coordinates of left upper corner and (x_2, y_2) be the right upper corner. The mid-point is calculated as follows:

$$\left(\frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2}\right) \tag{3}$$

Then using distance formulae, midpoint from lower eye lid is calculated given by Pythagoras theorem as given in Equation (4).

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 + y_1)^2} \tag{4}$$

finally, the decision is made for the state of eyes using the distance d value. if the value is zero or closed to zero the eye is considered as closed else open. The blink duration is ranges from 0.1-0.4 seconds for normal person so, if the driver is drowsy the blink rate is beyond this interval. 2 seconds is set as threshold value, if the eyes are closed for more than two seconds the system generates the alarm to alert the driver. The proposed algorithm is tested under different lighting conditions. Under normal or bright lighting, system achieves high accuracy but gives poor result in bad lights. The proposed method is compared with the previous methods like eye blink monitoring based on mean sift algorithm, face and eye monitoring based on neural networks & visual information, computer vision & machine learning algorithm, electrooculogram and vehicle-based measures. The proposed system achieves the accuracy of 94 % and less complex as compared to others.

10) EYE CLOSENESS DETECTION METHOD

Khunpisuth *et al.* [13] creates an experiment the calculates the drowsiness level of driver using Raspberry Pi camera and Raspberry Pi 3 model B. Firstly Pi camera captures video and to detect face regions in the images, Haar cascade classifier from Viola-Jones method is used. Several images are trained in different lights. The percentage of 83.09 % is achieved based on the case study with 10 volunteers. Blue rectangle shows the Region of Interest (ROI) that is face. Again, Haar cascade classifier is applied on the last obtained frame which reduces the size of ROI. After the face detection, drowsiness level is calculated using eye blink rate. Eye region is detected using template matching on the face and authors uses three templates to check the eye blink and aye area. CV_TM_CCOEFF_NORMED from OpenCV is considered as it gives improved results than other methods of template matching. The integration of eyes and face detection permits the checking of an eye blinking and closeness rate. If the eyes are closed, then the value of closed eye is higher than the open eyes and opposite case if eyes are open. Authors assumed that the Haar cascade classifier will work if the face is front facing position. That why authors proposed the method to rotate the tilted face back in to the front-facing position. Firstly, determines whether the head is tilt or not then calculates the degrees of rotation (angle). After the accurate detection of face and eyes, drowsiness level of driver is determined. If the drivers blink eyes too frequently, he system indicates he drowsiness. When the level reaches to one hundred, a loud sound will be generated to alert the driver. The proposed method is compared with Haar cascade and results shows that the proposed method achieves the accuracy of 99.59 %. It works in all lighting conditions and able to detect the face wearing glasses.

B. VEHICULAR PARAMETER-BASED TECHNIQUES

Vehicular parameter-based techniques try to detect driver fatigue based on vehicular features such as frequent lane

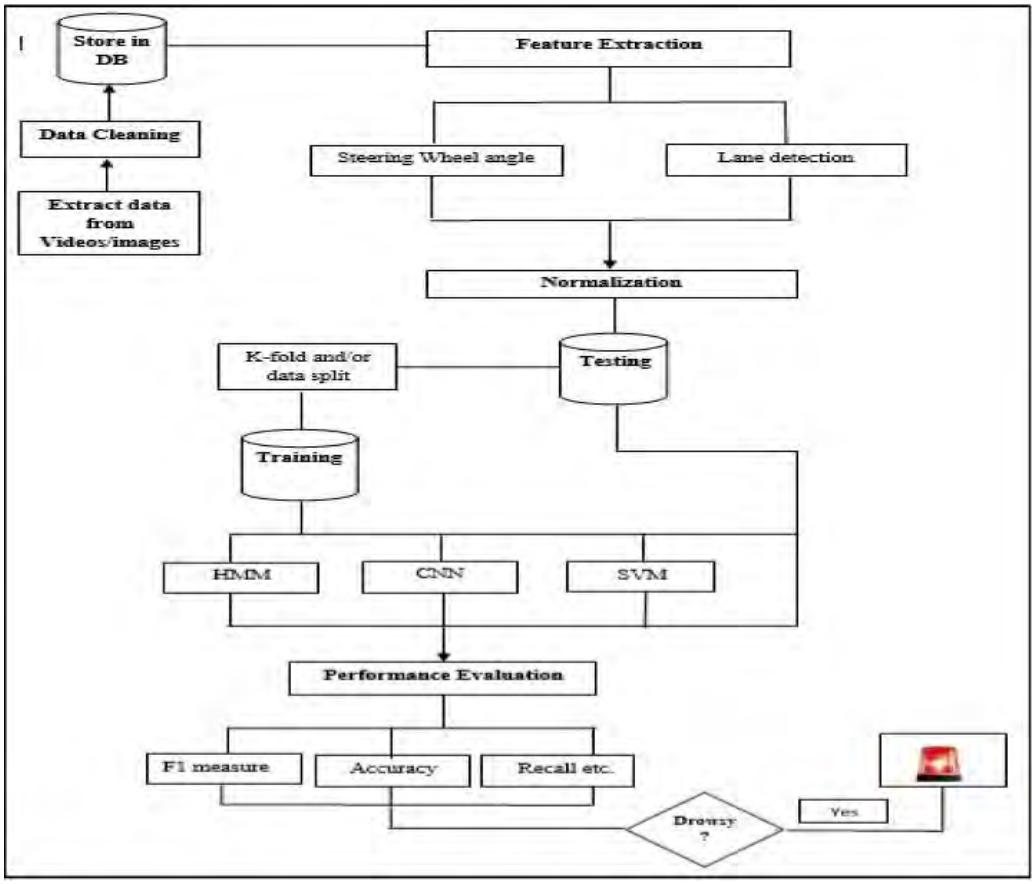


FIGURE 4. A general framework of vehicular pattern-based techniques.

changing patterns, vehicle speed variability, steering wheel angle, steering wheel grip force, etc. These measures require sensors on vehicle parts like steering wheel, accelerator or brake pedal etc. The signals generated by these sensors are used to analyze the drowsiness of drivers. The main goal of these techniques is to observe driving patterns and detect a decline in driving performance due to fatigue and tiredness. A general framework of a drowsiness detection system based on vehicular measures is represented in *Figure 4*.

A list of vehicular measures-based driver drowsiness detection systems is listed in Table 3. It is widely used in the vehicle-based measure by using the steering angle sensor for detecting the drowsiness of the driver. A single angle sensor is placed under the steering of the car used for detecting the drivers steering behaviors. During the drowsiness the driver made the steering wheel reversal, then the normal drivers. For reducing the effect of the lane change, the researcher considered only the change of little degree (0.5 to 5).

Figure 5 shows the SWM base detection technique. Normally, the behaviors of the drowsy driver also create an effect on the behavior of the driving tasks (like speed, acceleration, driver states, lane wide).

However, measuring driver fatigue, according to the vehicle movement is limited because the measurement values

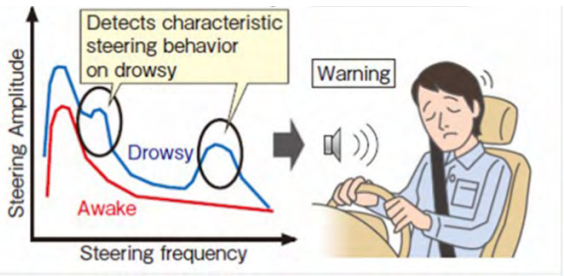


FIGURE 5. Steering movement-based detection.

can be easily affected by external factors such as the geometric characteristics of roads and weather conditions [14]. Steering wheel grip force measures seem fancy in drowsiness detection. But the problem with steering wheel grip force measures is that they are closely related with diver mood swings and road conditions as on an empty road driver may not grip the steering with that pressure with which he grips the steering on a busy road as well as grip force on a straight road is very much different with that on a dangerous mountain road. So the steering wheel grip force measures can be used for other category approaches such as eye movement for better results [15]. Sometimes drowsiness does not change

the vehicular interaction, thus vehicular parameters-based techniques became unreliable in such cases.

1) REAL TIME LANE DETECTION SYSTEM

Road accidents have become common in the present era, causing the severe damage to the property and also to the lives of people travelling. There are many reasons of road accidents like: rash driving, inexperience, ignoring signboards, jumping signal etc. To address the issues, Katyal *et al.* [16] proposed the Drivers' Drowsiness Detection system. The system works in two phases: firstly, detects lane based on hough transform. Secondly, detects the drivers' eyes to detect the drowsiness. For eye detection, firstly use viola jones method to detect face, then do the image segmentation, after that otsu thresholding is done and canny edge detection is applied. The obtained results is integrated with the circle detection hough transform method to detect eyes to detect the fatigue level. It will also work in low lightning conditions. Result shows that the proposed system is useful for the drivers travelling on lengthy routes, driving late night, drivers who drink and drive.

2) TIME SERIES ANALYSIS OF STEERING WHEEL ANGULAR VELOCITY

To avoid the road accidents, Zhenhai *et al.* [17] proposed the Driver Drowsiness Detection method using time series analysis of steering wheel angular velocity. The method firstly analyzes the behavior of steering below the fatigue, then temporal detection window is used as the detection feature to determine the angular velocity of steering wheel during time-series. In the temporal window, if the detection feature satisfies the variability constraints and extent constraints, then the state of drowsiness is detected accordingly. The experiment based on real testers is performed, and results shows that the proposed method outperforms the previous methods and useful in the real world.

3) STEERING WHEEL ANGLE FOR REAL DRIVING CONDITIONS FOR DDT

To avoid road accidents, Li *et al.* [18] proposed the online detection of Drowsiness Detection System to monitor the fatigue level of drivers under real conditions using Steering Wheel Angles (SWA). The data of SWA is collected from the sensors attached on the steering lever. The system firstly extracts the features of Approximate Entropy (ApEn) from fixed sliding windows on time series of real time steering wheel angles, then the system linearizes the features of ApEn using the deviation of adaptive linear piecewise fitting method. After that the system calculates the warping distance between the series of linear features of sample data. Finally, system determine the drowsiness state of drivers using warping distance according to the designed decision classifier. The empirical analysis uses the data collected in 14.68 hrs. driving under real road conditions and evaluated on two fatigue levels: drowsy and awake. Results show that the proposed system is capable for working online with an accuracy of 78.01 % and useful for the prevention of road accidents caused by drivers' fatigue.

4) AUTOMATIC DETECTION OF DRIVER FATIGUE

To address the issue of drivers' fatigue, an online detection of drivers' fatigue using the Steering Wheel Angles (SWA) and Yaw Angles (YA) information in the real driving conditions [19] is proposed. The system firstly investigates the operation features of SWA and YA in the different states of fatigue, after that calculates the ApEn features on time series of shot sliding window, then using the dynamic time series of non-linear feature construction theory and taking features of fatigue as input, designs a 2-6-6-3 multi-level Back Propagation (BP) neural network classifier to determine the fatigue detection. For empirical analysis, 15 hours long experiment is performed in real road conditions. The experts evaluated the retrieved data and categorized in three levels of fatigue: drowsy, very drowsy, and awake. And the experiment achieves the average accuracy of 88.02 % in fatigue detection and valuable for the engineering applications.

C. PHYSIOLOGICAL PARAMETER-BASED TECHNIQUES

The Physiological parameters-based techniques detect drowsiness based on drivers' physical conditions such as heart rate, pulse rate, breathing rate, respiratory rate and body temperature, etc. These biological parameters are more reliable and accurate in drowsiness detection as they are concerned with what is happening with driver physically. Fatigue or drowsiness, change the physiological parameters such as a decrease in blood pressure, heart rate and body temperature, etc. Physiological parameters-based drowsiness detection systems detect these changes and alert the driver when he is in the state, near to sleep. The advantage of this approach is that it alerts the driver to take some rest, before the physical symptoms of drowsiness appear. A general framework of a drowsiness detection system based on physiological parameters is presented in *Figure 6*.

A list of physiological condition-based drowsiness detection system is listed in Table 4. These measures are invasive, so require electrodes to be directly placed on the drivers' body. This method sometimes gets irritating for a driver so becomes difficult to implement.

1) EEG-BASED DRIVER FATIGUE DETECTION

The drivers' fatigue detection system using Electroencephalogram (EEG) signals [20] is proposed to avoid the road accidents usually caused due to drivers' fatigue. The proposed method firstly finds the index related to different drowsiness levels. The system takes EEG signal as input which is calculated by a cheap single electrode neuro signal acquisition device. To evaluate the proposed method, data set for simulated car driver under the different levels of drowsiness is collected locally. And result shows that the proposed system can detect all subjects of tiredness.

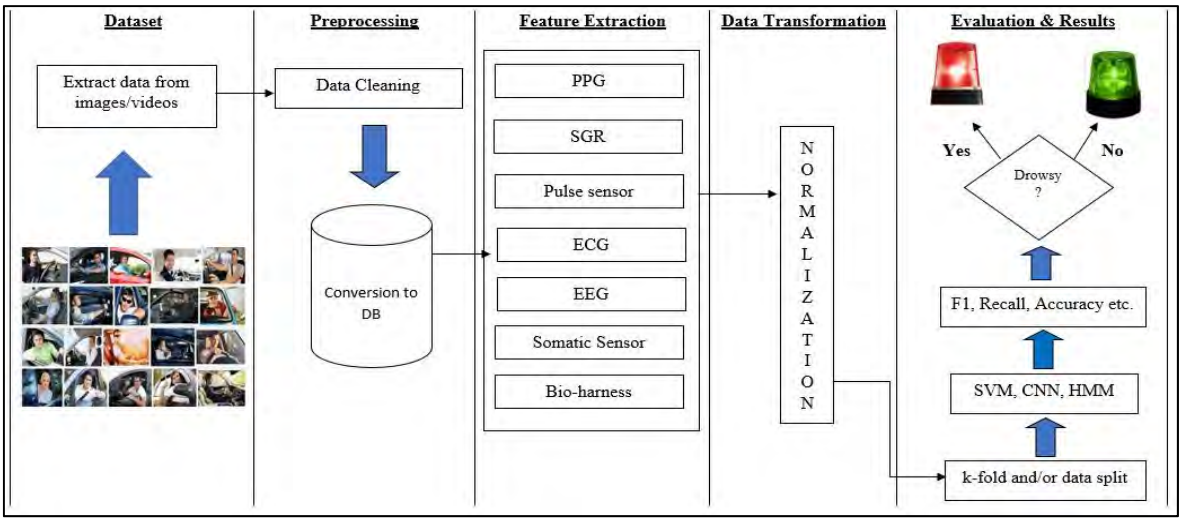


FIGURE 6. A general framework of Physiological parameters-based techniques.

TABLE 4. Physiological parameters-based drowsiness detection systems.

Ref	Biological Parameter	Sensor	Method & Classifier	Description	Efficiency
[20]	Brain physiological changes	EEG	IIR Band-pass Filter, PSD, Features Fusion	Based on low-cost Mind Wave Neuro-signal acquisition device to detect drowsiness using EEG signals.	Nil
[21]	Heart rate analysis	PPG	SVM classification, PERCLOS, Karolinska sleepiness scale (KSS), ROC analysis, wavelet transform method	Detects the driver drowsiness using heart rate analysis and wavelet method	95%
[22]	Heart pulse wave	Pulse sensor	HRV frequency domain, LF/HF ratio	Used non-intrusive sensor to sense heart pulse wave from driver finger or hand to calculate the drowsiness level.	Nil
[23]	Heart rate, respiratory rate, pulse rate, stress level	PPG, SGR	SVM, adaptive threshold method	Proposed wearable driver drowsiness detection mechanism in real driving conditions and does not require installation of components in the vehicle.	98.3%
[24]	Heart rate, breathing rate, RR interval	Bio Harness sensor	FFT, PSD, neural network classification	Works in two phases: design wearable Bio-harness sensor to detect biological parameters of driver and mobile based drowsiness detection system is designed.	Nil
[25]	Core body temperature, pulse rate	Somatic sensor, Temperature sensor LM-35, PPG	Viola- Jones algorithm, Haar Cascade classifier,	Used combination of behavioral and biological parameters to detect driver drowsiness.	80.55%
[26]	Heart rate, time domain measures, frequency domain measures	ECG, EEG	SVM, KSS	Used features extracted from EEG and ECG to detect drowsiness. Indicates that both physiological signals give better performance together.	80%

2) WAVELET ANALYSIS OF HEART RATE VARIABILITY & SVM CLASSIFIER

Li and Chung [21] proposed the driver drowsiness detection that uses wavelet analysis of Heart Rate Variability (HRV) and Support Vector Machine (SVM) classifier. The basic purpose is to categorize the alert and drowsy drivers using the wavelet transform of HRV signals over short durations. The system firstly takes Photo Plethysmo Graphy (PPG) signal as input and divide it into 1-minute intervals and then verify two driving events using average percentage of eyelid closure over pupil over time (PERCLOS) measurement over the interval. Secondly, the system performs the feature extraction of HRV time series based on Fast Fourier Transform (FFT) and wavelet. A Receiver Operation Curve (ROC) and SVM classifier is used for feature extraction and classification respectively. The analysis of ROC shows that the wavelet-based method gives improved results than the FFT-based method. Finally, the real time requirements for drowsiness detection, FFT and wavelet features are used to train the SVM classifier extracted from the HRV signals. The performance of classification using the wavelet-based features achieve the accuracy of 95%, sensitivity to 95% and

specificity to 95%. The FFT-based results achieve the accuracy of 68.85. The results show that wavelet-based methods perform better than the FFT-based methods.

3) PULSE SENSOR METHOD

Mostly, previous studies focus on the physical conditions of drivers to detect drowsiness. That's why Rahim *et al.* [22] detects the drowsy drivers using infrared heart-rate sensors or pulse sensors. The pulse sensor measures the heart pulse rate from drivers' finger or hand. The sensor is attached with the finger or hand, detects the amount of blood flowing through the finger. Then amount of the blood's oxygen is shown in the finger, which causes the infrared light to reflect off and to the transmitter. The sensor picks up the fluctuation of oxygen that are connected to the Arduino as microcontroller. Then, the heart pulse rate is visualizing by the software processing of HRV frequency domain. Experimental results show that LF/HF (Low to high frequency) ratio decreases as drivers go from the state of being awake to the drowsy and many road accidents can be avoided if an alert is sent on time.

4) WEARABLE DRIVER DROWSINESS DETECTION SYSTEM

Mobile based applications have been developed to detect the drowsiness of drivers. But mobile phones distract the drivers' attention and may cause accident. To address the issue, Leng *et al.* [23] proposed the wearable-type drowsiness detection system. The system uses self-designed wrist band consists of PPG signal and galvanic skin response sensor. The data collected from the sensors are delivered to the mobile device which acts as the main evaluating unit. The collected data are examined with the motion sensors that are built-in in the mobiles. Then five features are extracted from the data: heart rate, respiratory rate, stress level, pulse rate variability, and adjustment counter. The features are moreover used as the computation parameters to the SVM classifier to determine the drowsiness state. The experimental results show that the accuracy of the proposed system reaches up to 98.02 %. Mobile phone generates graphical and vibrational alarm to alert the driver.

5) WIRELESS WEARABLES METHOD

To avoid the disastrous road accidents, Warwick *et al.* [24] proposed the idea for drowsiness detection system using wearable Bio sensor called Bio-harness. The system has two phases. In the first phase, the physiological data of driver is collected using bio-harness and then analyzes the data to find the key parameters like ECG, heart rate, posture and others related to the drowsiness. In the second phase, drowsiness detection algorithm will be designed and develop a mobile app to alert the drowsy drivers.

6) DRIVER FATIGUE DETECTION SYSTEM

Chellappa *et al.* [25] presents the Driver fatigue detection system. The basic of the system is to detect the drowsiness when the vehicle is in the motion. The system has three components: external hardware (sensors and camera), data processing module and alert unit. Hardware unit communicates over the USB port with the rest of the system. Physiological and physical factors like pulse rate, yawning, closed eyes, blink duration and others are continuously monitored using somatic sensor. The processing module uses the combination of the factors to detect drowsiness. In the end, alert unit alerts the driver at multiple stages according to the severity of the symptoms.

7) HYBRID APPROACH UTILIZING PHYSIOLOGICAL FEATURES

To improve the performance of detection, Awais *et al.* [26] proposed the hybrid method which integrates the features of ECG and EEG. The method firstly extracts the time and frequency domain features like time domain statistical descriptors, complexity measures and power spectral measures from EEG. Then using ECG, features like heart rate, HRV, low frequency, high frequency and LF/HF ratio. After that, subjective sleepiness is measured to study its relationship with drowsiness. To select only statistically significant features, t-tests is used that can differentiate between the drowsy and alert. The features extracted from ECG and EEG are integrated to study the improvements in the performance using SVM. The other main contribution is to study the channel reduction and its impact on the performance of detection. The method measures the differences between the drowsy and alert state from physiological data collected from the driving simulated-based study. Monotonous driving environment is used to induce the drowsiness in the participants. The proposed method demonstrated that combining ECG and EEG improves the performance of system in differentiating the drowsy and alert states, instead of using them alone. The analysis of channel reduction confirms that the accuracy level reaches to 80% by just combining the ECG and EEG. The performance of the system indicates that the proposed system is feasible for practical drowsiness detection system.

IV. A COMPARATIVE STUDY OF DDT

Each drowsiness detection technique has its own pros and cons. Table 5 presents the advantages and disadvantage of each drowsiness detection technique discussed above. Non-intrusive drowsiness detection techniques are easy to use but do not provide accurate results. Behavioral parameters-based techniques are affected by illumination effects while vehicular parameters-based techniques are dependent upon the weather condition and geometric conditions of roads so sometimes become unreliable. Physiological parameters-based techniques give accurate results, but their intrusive nature needs to be resolved. Researchers are working to use physiological measures in a less- intrusive way with wireless devices. Sensors are placed on the drivers' body and signals are obtained wirelessly. The electrodes are also embedded on driving seat, seat belt, seat cover or steering wheel now a day to reduce the intrusiveness [27], [28]. Less-intrusive system decreases in accuracy as compared to intrusive physiological measures because of lesser direct

TABLE 5. Pros and cons of each technique.

Technique	Parameters	Pros	Cons
Behavioral parameters-based DDT	Eye blinking, Eye closeness ratio, Head movement, Yawning	Non-intrusive Easy to use	Effected by illumination, Lightening conditions
Vehicle parameters-based DDT	Steering wheel behavior, yaw angle, lane changing pattern	Non-intrusive	Effected by geometric characteristics of roads, Unreliable
Physiological parameters-based DDT	Heart rate, pulse rate, brain activity, respiratory rate, body temperature	Efficient, reliable	Intrusive

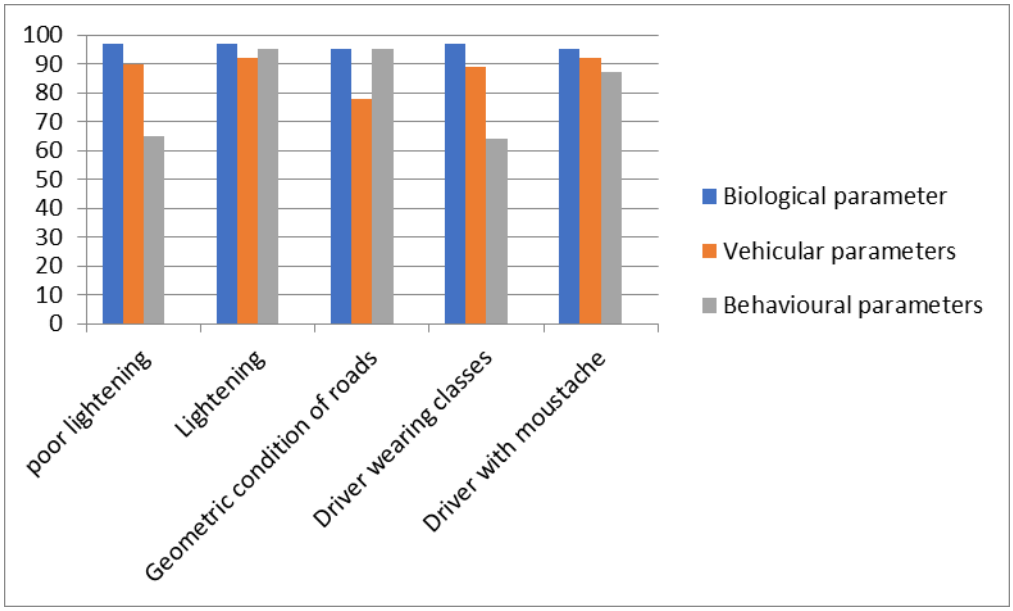


FIGURE 7. Comparative graph of DDT.

contact with the driver and errors in proper contact with electrodes in wireless communication. Still, there is a need to overcome the problems associated with physiological parameters-based detection techniques.

The detailed study of drowsiness detection systems based on behavior, vehicle and biological parameters, a comparative graph is generated in Figure 7. The efficiency of each drowsiness detection technique under certain circumstances is illustrated. The circumstances discussed here is poor lightening, proper lightening, and Geometric condition of roads, drivers with sun glasses and drivers with mustache. This analysis shows that biological parameters give stable and good results in such conditions, while the performance of behavioral parameters-based techniques reduce in poor lightening, driving with glasses and mustache. Biological parameters-based techniques are also reduced in efficiency in the bad geometric condition of roads.

V. HYBRID APPROACHES OF DDT

Physiological, behavioral and vehicular parameters for drowsiness detection can be used in combination with each other in such a way that one technique can reduce the limitations of others to increase overall accuracy of the system. A list of drowsiness detection system using hybrid parameters is listed in Table 6.

VI. CLASSIFICATION METHODS USED FOR DDT

Detail description of classification methods and their impact in drowsiness detection systems are discussed in detail. Pros and cons of each classification technique and comparative analysis of error rate in each of them is illustrated in this section of the systematic review paper. A detailed review of classification techniques, their pros and cons and later their graph-based comparison are discussed. Various types of machine learning classifiers are used for training data in the drowsiness detection system. Selection of appropriate classifier has strong impact on the efficiency of the system. Generally, SVM, HMM and CNN classifiers are used due to their high accuracy in drowsiness systems are compared to another set of classifiers such as KNN, HOG etc. Details of each of these three classifiers are discussed.

A. SUPPORT VECTOR MACHINE (SVM)

For classification and regression, SVMs are the supervised learning approaches. SVMs are firstly used for the selection of the training data set in a pre-defined form of data. In the

TABLE 6. Hybrid approaches of drowsiness detection.

Ref	Hybrid approach	Hybrid parameters	Method & Classifiers	Accuracy
[29]	Physiological, Behavioral and vehicular	Heart rate, BP, temperature, speed eyelid closure ratio	HRV color model, Fuzzy Bayesian network	94%-99%
[30]	Physiological and behavioral	Heart rate, Eye blinking duration	PSG, NASA lead method	Nil
[31]	Behavioral and vehicular	Eye status, lateral position, steering wheel angle	KSS, Neutral network, BPL algorithm.	87.78%-94.63%
[26]	Physiological and behavioral	Head movement, heart rate	Frame difference algorithm, R-peak detection algorithm	Nil
[32]	Physiological and behavioral	Heart rate, Eyelid closure ratio	PERCLOS, Burg method, Kruskal-Wallis test,	86%-96%

TABLE 7. Drowsiness detection systems using SVM.

Ref	Measure	Classifier	Description	Accuracy
[33]	Eye state	SVM	Based on the supervised learning approach.	98.4%
[34]	Eye state	Binary	Used binary method to detect the eye state	93.5%
[35]	Eye Closure	Harr Feature with SVMs	Uses the feature of the Haar algorithm with SVM's for the eye closure detection.	99.74%
[36]	Eye state	HOG and SVM	HOG is a feature extraction algorithm. SVM is used to detect the Current state of the eye.	91.6%
[37]	Eye Closure and Yawning	Binary SVM with linear kernel	Uses Binary SVM with linear kernel to detect the Eye closure and yawning.	94.5%

TABLE 8. Drowsiness detection systems using HMM.

Ref	Measures	Classifier	Description	Accuracy
[38]	EYE state	HMM	Used to detect the eye state information	99.7%
[39]	Eye blink	HMM	Used HMM to detect eye blinking rate	95.7&
[40]	Eye blinks	SVM and HMM	Detects drowsiness using eye blinking rate.	90.99%
[41]	Eye closure and other features	HMM and SVM	Detect the closures of the eye and some other features.	97%
[42]	Eye State	HMM	Used for the detection of the eye states.	95.9%
[43]	Eye state and head position	HMM	Eye state and head position are used for drowsiness detection	Nil

drowsiness detection, SVMs learns from the categorized data into the classified form of data. A number of measures are used in the detection of the driver drowsiness and the level of the driver drowsiness. A fully automatic system for the detection of the driver drowsiness is presented [33], Haar feature algorithm is used for the detection of the Eyes and face detection, SVMs is trained in the states like close, open eyes and to trigger the alarm. This framework is resulting in the accuracy of 100%, but its result is achievable in the lower frame rate, which leads us to the missed facial expression. A list of systems using SVMs classifiers are presented in Table 7.

B. HIDDEN MARKOV MODEL (HMM)

HMM is the statistical model used to predict the hidden state on the basis of observed state. In many applications, HMM is used like facial expression detection, gene annotation, modeling DNA, computer virus classification, and sequence errors. Table 8 lists the various features and approaches used by the HMM-based Downiness Detector. They proposed a new facial feature by detecting the change in the wrinkle by calculating the face intensity. IR (Infra-Red) webcam is used to eliminate changes for both day and night conditions. This system leads to the drawback because older people have a deeper wrinkle. Also, HMM technique is implemented for the tracking of eye based on color and its geometrical features, but the system fails to detect the face it driver is not looking forward and designed for the indoor conditions.

C. CONVOLUTIONAL NEURAL NETWORKS (CNN)

CNN's are like a standard neural network that is additionally made from the neurons that incorporate with the learnable

TABLE 9. Drowsiness detection systems using CNN.

Ref	Measure	Classifier	Description	Efficiency
[44]	Visual Feature	CNN with SoftMax layer	Used voila and Jones algorithm with the CCN and SoftMax layer of the visual features.	78%
[45]	Eye gaze	CNN	Used voila and Jones algorithm with the CCN for the eye gaze.	98%
[46]	Eye State	MTCNN and DDDN	Used MTCNN and DDDN for the eye state detection	91.6%
[47]	Eye State	CNN	Used the Ada-boost and LBF and PERCLOS with CNN for the Eye State Detection	95.18%

TABLE 10. Pros and cons of classifiers.

Technique	Parameters	Pros	Cons
HMM	Probabilities, hidden states, eye state, head position, eye blinks, eye closure	Capture dependencies between the measurement, able to represent the variance of the appliance's power	Limited by the intrinsic structure
CNN	Eye state, visual features, eye gaze,	Accuracy in the image recognition	Slow to train, need lots of training data, high computational cost.
SVM	Eye Closure, Eye State, Eye Closure and yawning	Guarantees optimal, abundance of implementation, implicitly, good out-of-sample generalization.	Not suitable for large data set, less effective on nosier dataset

weights. CNN uses the layers of the spatial convolution that is considered best for the image, which show the strong correlations. CNN is used in many applications and proven itself successful such as image recognition, classification, and video analysis. CNN is firstly applied to Computer Visions by CUN and Yoshua but the best results are generated in 2012 for the object recognition shows the excellent results in Deep CNN. Representation learning is used in the proposed algorithm for the detection of driver drowsiness, here the Viola and Jones algorithm is used to detect the face. Firstly, the images are cropped in 48∗48 size image and then fed up to the outmost layer of the network contained the 20 filters, output is delivered to the SoftMax layer, but the system leads to failure because it does not consider the head pose. But another author achieved the more accurate result by using the 3D deep neutral network in which face is tackle by the combination of two more filters, the system works well even when the driver is changing the head. A list of CNN classifier-based drowsiness detection systems is listed in Table 9.

VII. COMPARATIVE STUDY OF CLASSIFICATION METHODS

Each of classifier has its own advantages and disadvantages. It is not necessary that each classifier will be suitable in every situation. Selection of appropriate classifier according to the system requirements is very significant for better efficiency and accuracy. Type of parameters for these classifiers and their pros and cons are discussed in Table 10.

A comparative analysis of test error rates of SVM, HMM, and CNN classification methods are illustrated. Lesser error rate leads to greater efficiency. Comparative analysis is presented in Figure 8. Comparative analysis of our study shows that HMM is more accurate as it has a lesser error rate as compared to the other two. But SVM is easy to use, so it is most widely used classification method.

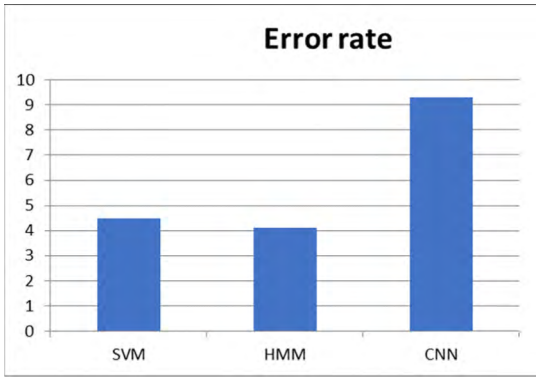


FIGURE 8. Comparative graph of classifiers.

VIII. CONCLUSION

The main idea of this systematic review is to discover the state-of-the-art research in the drowsiness detection system. The systematic review provides details of behavioral, vehicular and physiological parameters-based drowsiness detection techniques. These techniques are elaborated in detail and their pros and cons are also discussed. The comparative analysis showed that none of these techniques provide full accuracy, but physiological parameters-based techniques give more accurate results than others. Their non-intrusiveness can be reduced using wireless sensors on the drivers' body, driving seat, seat cover, steering wheel, etc. Hybrid of these techniques such as physiological measures combined with vehicular or behavioral measures, helps in overcoming the problem associated with individual technique thus results in

improved drowsiness detection results like the combination of ECG and EEG features achieves the high-performance results emphasizing the fact that combining the physiological signals improves the performance instead of using them alone.

The top supervised learning techniques have been presented. The pros and cons and comparative study of such techniques are discussed in detail. A comparative study of classifier shows different accuracy in various situations. However, SVM is the mostly commonly used classifiers which gives better accuracy and speed in most of the situations, but not suitable for large datasets. HMM shows a less error rate, but both CNN and HMM are slow in training and expensive as compared to SVM classifier.

REFERENCES

[1] D. Budgen and P. Brereton, "Performing systematic literature reviews in software engineering," in *Proc. 28th Int. Conf. Softw. Eng.*, May 2006, pp. 1051–1052.
[2] L. Barr, S. Popkin, and H. Howarth, "An evaluation of emerging driver fatigue detection measures and technologies," Federal Motor Carrier Saf. Admin., Washington, DC, USA, Tech. Rep. FMCSA-RRR-09-005, 2009.
[3] W.-B. Horng, C.-Y. Chen, Y. Chang, and C.-H. Fan, "Driver fatigue detection based on eye tracking and dynamic template matching," in *Proc. IEEE Int. Conf. Netw., Sens. Control*, vol. 1, Mar. 2004, pp. 7–12.
[4] M. Saradadevi and P. Bajaj, "Driver fatigue detection using mouth and yawning analysis," *Int. J. Comput. Sci. Netw. Secur.*, vol. 8, no. 6, pp. 183–188, Jun. 2008.
[5] M. A. Assari and M. Rahmati, "Driver drowsiness detection using face expression recognition," in *Proc. IEEE Int. Conf. Signal Image Process. Appl. (ICSIPA)*, Nov. 2011, pp. 337–341.
[6] C. Yan *et al.*, "Video-based classification of driving behavior using a hierarchical classification system with multiple features," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 30, no. 5, 2016, Art. no. 1650010.
[7] I. Teyeb, O. Jemai, M. Zaied, and C. B. Amar, "A novel approach for drowsy driver detection using Head posture estimation and eyes recognition system based on wavelet network," in *Proc. 5th Int. Conf. Inf., Intell., Syst. Appl. (IISA)*, Jul. 2014, pp. 379–384.
[8] N. Kuamr, N. C. Barwar, and N. Kuamr, "Analysis of real time driver fatigue detection based on eye and yawning," *Int. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 6, pp. 7821–7826, 2014.
[9] R. Ahmad and J. N. Borole, "Drowsy driver identification using eye blink detection," *Int. J. Comput. Sci. Inf. Technol.*, vol. 6, no. 1, pp. 270–274, Jan. 2015.
[10] T. P. Nguyen, M. T. Chew, and S. Demidenko, "Eye tracking system to detect driver drowsiness," in *Proc. 6th Int. Conf. Automat., Robot. Appl. (ICARA)*, Feb. 2015, pp. 472–477.
[11] T. Liu, J. Xie, W. Yan, and P. Li, "Driver's face Detection using space-time restrained Adaboost method," *KSII Trans. Internet Inf. Syst.*, vol. 6, no. 9, pp. 2341–2350, Sep. 2012.
[12] A. Rahman, M. Sirshar, and A. Khan, "Real time drowsiness detection using eye blink monitoring," in *Proc. Nat. Softw. Eng. Conf. (NSEC)*, Dec. 2015, pp. 1–7.
[13] O. Khunpisuth, T. Chotchinasri, V. Koschakosai, and N. Hnoohom, "Driver drowsiness detection using eye-closeness detection," in *Proc. 12th Int. Conf. Signal-Image Technol. Internet-Based Syst. (SITIS)*, Nov./Dec. 2016, pp. 661–668.
[14] M. Ingre, T. Åkerstedt, B. Peters, A. Anund, and G. Kecklund, "Subjective sleepiness, simulated driving performance and blink duration: Examining individual differences," *J. Sleep Res.*, vol. 15, no. 1, pp. 47–53, Mar. 2006.
[15] A. Polychronopoulos, A. Amditis, and E. Bekiaris, "Information data flow in AWAKE multi-sensor driver monitoring system," in *Proc. IEEE Intell. Vehicles Symp.*,, Jun. 2004, pp. 902–906.
[16] Y. Katyal, S. Alur, and S. Dwivedi, "Safe driving by detecting lane discipline and driver drowsiness," in *Proc. IEEE Int. Conf. Adv. Commun., Control Comput. Technol.*, May 2014, pp. 1008–1012.
[17] G. Zhenhai, L. DinhDat, H. Hongyu, Y. Ziwen, and W. Xinyu, "Driver drowsiness detection based on time series analysis of steering wheel angular velocity," in *Proc. 9th Int. Conf. Measuring Technol. Mechatron. Automat. (ICMTMA)*, Jan. 2017, pp. 99–101.
[18] Z. Li, S. E. Li, R. Li, B. Cheng, and J. Shi, "Online detection of driver fatigue using steering wheel angles for real driving conditions," *Sensors*, vol. 17, no. 3, p. 495, Mar. 2017.
[19] Z. Li, L. Chen, J. Peng, and Y. Wu, "Automatic detection of driver fatigue using driving operation information for transportation safety," *Sensors*, vol. 17, no. 6, p. 1212, May 2017.
[20] H. S. AlZu'bi, W. Al-Nuaimy, and N. S. Al-Zubi, "EEG-based driver fatigue detection," in *Proc. 6th Int. Conf. Develop. Syst. Eng. (DESE)*, Dec. 2013, pp. 111–114.
[21] G. Li and W.-Y. Chung, "Detection of driver drowsiness using wavelet analysis of heart rate variability and a support vector machine classifier," *Sensors*, vol. 13, no. 12, pp. 16494–16511, Dec. 2013.
[22] H. A. Rahim, A. Dalimi, and H. Jaafar, "Detecting drowsy driver using pulse sensor," *J. Technol.*, vol. 73, no. 3, pp. 5–8, Mar. 2015.
[23] L. B. Leng, L. B. Giin, and W.-Y. Chung, "Wearable driver drowsiness detection system based on biomedical and motion sensors," in *Proc. IEEE Sensors*, Nov. 2015, pp. 1–4.
[24] B. Warwick, N. Symons, X. Chen, and K. Xiong, "Detecting driver drowsiness using wireless wearables," in *Proc. 12th Int. Conf. Mobile Ad Hoc Sensor Syst. (MASS)*, Oct. 2015, pp. 585–588.
[25] Y. Chellappa, N. N. Joshi, and V. Bharadwaj, "Driver fatigue detection system," in *Proc. IEEE Int. Conf. Signal Image Process. (ICSIP)*, Aug. 2016, pp. 655–660.
[26] M. Awais, N. Badruddin, and M. Drieberg, "A hybrid approach to detect driver drowsiness utilizing physiological signals to improve system performance and wearability," *Sensors*, vol. 17, no. 9, p. 1991, Aug. 2017.
[27] I. Teyeb, O. Jemai, M. Zaied, and C. B. Amar, "Towards a Smart Car Seat Design for Drowsiness Detection Based on Pressure Distribution of the Driver's Body," in *Proc. ICSEA*, Aug. 2016, p. 230.
[28] P. Sandela, "A smart seat belt,"
[29] B.-G. Lee and W.-Y. Chung, "A smartphone-based driver safety monitoring system using data fusion," *Sensors*, vol. 12, no. 12, pp. 17536–17552, Dec. 2012.
[30] M. Miyaji, "Method of drowsy state detection for driver monitoring function," *Int. J. Inf. Electron. Eng.*, vol. 4, no. 4, p. 264, Jul. 2014.
[31] S. Samiee, S. Azadi, R. Kazemi, A. Nahvi, and A. Eichberger, "Data fusion to develop a driver drowsiness detection system with robustness to signal loss," *Sensors*, vol. 14, no. 9, pp. 17832–17847, Sep. 2014.
[32] A. Lemkaddem *et al.*, "Multi-modal driver drowsiness detection: A feasibility study," in *Proc. IEEE EMBS Int. Conf. Biomed. Health Inform. (BHI)*, Mar. 2018, pp. 9–12.
[33] M. Sabet, R. A. Zoroofi, K. Sadeghniiat-Haghighi, and M. Sabbaghian, "A new system for driver drowsiness and distraction detection," in *Proc. 20th Iranian Conf. Elect. Eng. (ICEE)*, May 2012, pp. 1247–1251.
[34] A. Punitha, M. K. Geetha, and A. Sivaprakash, "Driver fatigue monitoring system based on eye state analysis," in *Proc. Int. Conf. Circuits, Power Comput. Technol. (ICCPCT)*, Mar. 2014, pp. 1405–1408.
[35] G. J. AL-Anizy, M. J. Nordin, and M. M. Razooq, "Automatic driver drowsiness detection using haar algorithm and support vector machine techniques," *Asian J. Appl. Sci.*, vol. 8, no. 2, pp. 149–157, 2015.
[36] L. Pauly and D. Sankar, "Detection of drowsiness based on HOG features and SVM classifiers," in *Proc. IEEE Int. Conf. Res. Comput. Intell. Commun. Netw. (ICRCICN)*, Nov. 2015, pp. 181–186.
[37] B. N. Manu, "Facial features monitoring for real time drowsiness detection," in *Proc. 12th Int. Conf. Innov. Inf. Technol. (IIT)*, Nov. 2016, pp. 1–4.
[38] A. M. Bagci, R. Ansari, A. Khokhar, and E. Cetin, "Eye tracking using Markov models," in *Proc. 17th Int. Conf. Pattern Recognit. (ICPR)*, vol. 3, Aug. 2004, pp. 818–821.
[39] G. Pan, L. Sun, Z. Wu, and S. Lao, "Eyeblink-based anti-spoofing in face recognition from a generic webcamera," in *Proc. 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.
[40] Y. Sun, S. Zafeiriou, and M. Pantic, "A hybrid system for on-line blink detection," in *Proc. Hawaii Int. Conf. Syst. Sci.*, Jan. 2013, pp. 1–5.
[41] E. Tadesse, W. Sheng, and M. Liu, "Driver drowsiness detection through HMM based dynamic modeling," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May/Jun. 2014, pp. 4003–4008.
[42] B. Zhang, W. Wang, and B. Cheng, "Driver eye state classification based on cooccurrence matrix of oriented gradients," *Adv. Mech. Eng.*, vol. 7, no. 2, Nov. 2015, Art. no. 707106.
[43] I.-H. Choi, C.-H. Jeong, and Y.-G. Kim, "Tracking a driver's face against extreme head poses and inference of drowsiness using a hidden Markov model," *Appl. Sci.*, vol. 6, no. 5, p. 137, May 2016.

[44] K. Dwivedi, K. Biswaranjan, and A. Sethi, "Drowsy driver detection using representation learning," in *Proc. IEEE Int. Advance Comput. Conf. (IACC)*, Feb. 2014, pp. 995–999.
[45] A. George and A. Routray, "Real-time eye gaze direction classification using convolutional neural network," in *Proc. Int. Conf. Signal Process. Commun. (SPCOM)*, Jun. 2016, pp. 1–5.
[46] B. Reddy, Y.-H. Kim, S. Yun, C. Seo, and J. Jang, "Real-time driver drowsiness detection for embedded system using model compression of deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 438–445.
[47] F. Zhang, J. Su, L. Geng, and Z. Xiao, "Driver fatigue detection based on eye state recognition," in *Proc. Int. Conf. Mach. Vis. Inf. Technol. (CMVIT)*, Feb. 2017, pp. 105–110.

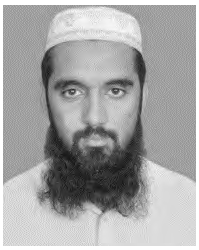
MUHAMMAD RAMZAN is currently pursuing the Ph.D. degree with the University of Management and Technology, Lahore, Pakistan. He is currently a Lecturer with the University of Sargodha, Pakistan. He has authored several research articles published in reputed peer-reviewed journals. His areas of research include algorithms, machine learning, software engineering, and computer vision.

HIKMAT ULLAH KHAN received the master's degree in computer science and the Ph.D. degree in computer science from International Islamic University, Islamabad. He has been an Active Researcher for the last ten years. He is currently an Assistant Professor with the Department of Computer Science, COMSATS University Islamabad, Wah Cantt, Pakistan. He has authored a number of research articles in top peer-reviewed journals and international conferences. His research interests include social Web mining, semantic Web, data science, information retrieval, and scientometrics. He is an Editorial Board Member of a number of prestigious impact factor journals.

SHAHID MAHMOOD AWAN is currently a Lecturer with the School of Systems and Technology, University of Management and Technology, Lahore, Pakistan. His research interests include artificial intelligence, machine learning, load forecasting, and optimization techniques.

AMINA ISMAIL is currently pursuing the M.S. degree in computer science from COMSATS University Islamabad at Wah Campus, Wah Cantt, Pakistan. Her research interests include data mining, machine learning, information retrieval, and scientometrics.

MAHWISH ILYAS is currently pursuing the Ph.D. degree with the Computer Science and Information Technology Department, University of Sargodha, Sargodha, Pakistan. She is the Visiting Lecturer with the Computer Science Department, University of Sargodha. Her areas of research include machine learning, digital image processing, data mining, software engineering, and artificial intelligence.



AHSAN MAHMOOD received the master's degree in computer science from COMSATS University at Attock Campus, Attock, Pakistan. His research interests include data mining, social media analysis, sentiment analysis, and machine learning.

Computers and Electronics in Agriculture 177 (2020) 105709

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Crop yield prediction using machine learning: A systematic literature review

Thomas van Klompenburg[a], Ayalew Kassahun[a], Cagatay Catal[b,*]

[a] Information Technology Group, Wageningen University & Research, Wageningen, the Netherlands
[b] Department of Computer Engineering, Bahcesehir University, Istanbul, Turkey

ARTICLE INFO

Keywords:
Crop yield prediction
Decision support system
Systematic literature review
Machine learning
Deep learning

ABSTRACT

Machine learning is an important decision support tool for crop yield prediction, including supporting decisions on what crops to grow and what to do during the growing season of the crops. Several machine learning algorithms have been applied to support crop yield prediction research. In this study, we performed a Systematic Literature Review (SLR) to extract and synthesize the algorithms and features that have been used in crop yield prediction studies. Based on our search criteria, we retrieved 567 relevant studies from six electronic databases, of which we have selected 50 studies for further analysis using inclusion and exclusion criteria. We investigated these selected studies carefully, analyzed the methods and features used, and provided suggestions for further research. According to our analysis, the most used features are temperature, rainfall, and soil type, and the most applied algorithm is Artificial Neural Networks in these models. After this observation based on the analysis of machine learning-based 50 papers, we performed an additional search in electronic databases to identify deep learning-based studies, reached 30 deep learning-based papers, and extracted the applied deep learning algorithms. According to this additional analysis, Convolutional Neural Networks (CNN) is the most widely used deep learning algorithm in these studies, and the other widely used deep learning algorithms are Long-Short Term Memory (LSTM) and Deep Neural Networks (DNN).

1. Introduction

Machine learning (ML) approaches are used in many fields, ranging from supermarkets to evaluate the behavior of customers (Ayodele, 2010) to the prediction of customers' phone use (Witten et al., 2016). Machine learning is also being used in agriculture for several years (McQueen et al., 1995). Crop yield prediction is one of the challenging problems in precision agriculture, and many models have been proposed and validated so far. This problem requires the use of several datasets since crop yield depends on many different factors such as climate, weather, soil, use of fertilizer, and seed variety (Xu et al., 2019). This indicates that crop yield prediction is not a trivial task; instead, it consists of several complicated steps. Nowadays, crop yield prediction models can estimate the actual yield reasonably, but a better performance in yield prediction is still desirable (Filippi et al., 2019a).

Machine learning, which is a branch of Artificial Intelligence (AI) focusing on learning, is a practical approach that can provide better yield prediction based on several features. Machine learning (ML) can determine patterns and correlations and discover knowledge from datasets. The models need to be trained using datasets, where the outcomes are represented based on past experience. The predictive model is built using several features, and as such, parameters of the models are determined using historical data during the training phase. For the testing phase, part of the historical data that has not been used for training is used for the performance evaluation purpose.

An ML model can be descriptive or predictive, depending on the research problem and research questions. While descriptive models are used to gain knowledge from the collected data and explain what has happened, predictive models are used to make predictions in the future (Alpaydin, 2010). ML studies consist of different challenges when aiming to build a high-performance predictive model. It is crucial to select the right algorithms to solve the problem at hand, and in addition, the algorithms and the underlying platforms need to be capable of handling the volume of data.

To get an overview of what has been done on the application of ML in crop yield prediction, we performed a systematic literature review (SLR). A Systematic Literature Review (SLR) shows the potential gaps in research on a particular area of problem and guides both practitioners and researchers who wish to do a new research study on that problem area. By following a methodology in SLR, all relevant studies are accessed from electronic databases, synthesized, and presented to respond to research questions defined in the study. An SLR study leads to new perspectives and helps new researchers in the field to understand the state-of-the-art.

* Corresponding author.
E-mail address: cagatay.catal@eng.bau.edu.tr (C. Catal).

https://doi.org/10.1016/j.compag.2020.105709
Received 29 January 2020; Received in revised form 21 July 2020; Accepted 9 August 2020
Available online 18 August 2020
0168-1699/

An SLR study is expected to be replicable, which means that all the steps taken need to be explained clearly, and the results should be transparent for other researchers. The critical factors for a successful SLR study are objectivity and transparency (Kitchenham et al., 2007). As its name indicates, an SLR needs to be systematic and cover all the literature published so far. This study presents all the available literature published so far on the application of machine learning in crop yield prediction problem. In this study, we present our empirical results and responses to the research questions defined as part of this review article.

The remainder of this paper is organized as follows: Section 2 explains the background. Section 3 discusses the methodology. Section 4 presents the results of the SLR. Section 5 explains the deep learning-based crop yield prediction research. Section 5 presents the discussion, and Section 7 concludes this paper.

2. Related work

Crop yield prediction is an essential task for the decision-makers at national and regional levels (e.g., the EU level) for rapid decision-making. An accurate crop yield prediction model can help farmers to decide on what to grow and when to grow. There are different approaches to crop yield prediction. This review article has investigated what has been done on the use of machine learning in crop yield prediction in the literature.

During our analysis of the retrieved publications, one of the exclusion criteria is that the publication is a survey or traditional review paper. Those excluded publications are, in fact, related work and are discussed in this section. Chlingaryan and Sukkarieh performed a review study on nitrogen status estimation using machine learning (Chlingaryan et al., 2018). The paper concludes that quick developments in sensing technologies and ML techniques will result in cost-effective solutions in the agricultural sector. Elavarasan et al. performed a survey of publications on machine learning models associated with crop yield prediction based on climatic parameters. The paper advises looking broad to find more parameters that account for crop yield (Elavarasan et al., 2018). Liakos et al. (2018) published a review paper on the application of machine learning in the agricultural sector. The analysis was performed with publications focusing on crop management, livestock management, water management, and soil management. Li, Lecourt, and Bishop performed a review study on determining the ripeness of fruits to decide the optimal harvest time and yield prediction (Li et al., 2018). Mayuri and Priya addressed the challenges and methodologies that are encountered in the field of image processing and machine learning in the agricultural sector and especially in the detection of diseases (Mayuri and Priya, xxxx). Somvanshi and Mishra presented several machine learning approaches and their application in plant biology (Somvanshi and Mishra, 2015). Gandhi and Armstrong published a review paper on the application of data mining in the agricultural sector in general, dealing with decision making. They concluded that further research needs to be done to see how the implementation of data mining into complex agricultural datasets could be realized (Gandhi and Armstrong, 2016). Beulah performed a survey on the various data mining techniques that are used for crop yield prediction and concluded that the crop yield prediction could be solved by employing data mining techniques (Beulah, 2019).

According to our survey of review articles, the significant ones of which are presented in this section, this paper is the first SLR that focuses on the application of machine learning in the crop yield prediction problem. The existing survey studies did not systematically review the literature, and most of them reviewed studies on a specific aspect of crop yield prediction. Also, we presented 30 deep learning-based studies in this article and discussed which deep learning algorithms have been used in these studies.

3. Methodology

3.1. Review protocol

Before conducting the systematic review, a review protocol is defined. The review has been done using the well-known review guidelines provided by Kitchenham et al. (2007). Firstly, the research questions are defined. When research questions are ready, databases are used to select the relevant studies. The databases that were used in this study are Science Direct, Scopus, Web of Science, Springer Link, Wiley, and Google Scholar. After the selection of relevant studies, they were filtered and assessed using a set of exclusion and quality criteria. All the relevant data from the selected studies are extracted, and eventually, the extracted data were synthesized in response to the research questions. The approach we followed can be split up into three parts: plan review, conduct review, and report review.

The first stage is planning the review. In this stage, research questions are identified, a protocol is developed, and eventually, the protocol is validated to see if the approach is feasible. In addition to the research questions, publication venues, initial search strings, and publication selection criteria are also defined. When all of this information is defined, the protocol is revised one more time to see if it represents a proper review protocol. In Fig. 1, the internal steps of the Plan Review stage are represented.

The second stage is conducting the review, which is represented in Fig. 2. When conducting the review, the publications were selected by going through all the databases. The data was extracted, which means that their information regarding authors, year of publication, type of publication, and more information regarding the research questions were stored. After all the necessary data was extracted correctly, the data was synthesized in order to provide an overview of the relevant papers published so far.

In the final stage, a.k.a., Reporting the Review, the review was concluded by documenting the results and addressing the research questions, as shown in Fig. 3.

3.2. Research questions

This SLR aims to get insight into what studies have been published in the domain of ML and crop yield prediction. To get insight, studies have been analyzed from several dimensions. For this SLR study, the following four research questions(RQs) have been defined.

- RQ1- Which machine learning algorithms have been used in the literature for crop yield prediction?
- RQ2- Which features have been used in literature for crop yield prediction using machine learning?
- RQ3- Which evaluation parameters and evaluation approaches have been used in literature for crop yield prediction?
- RQ4- What are challenges in the field of crop yield prediction using machine learning?

3.3. Search strategy

The searching is done by narrowing down to the basic concepts that are relevant for the scope of this review. Machine learning has many

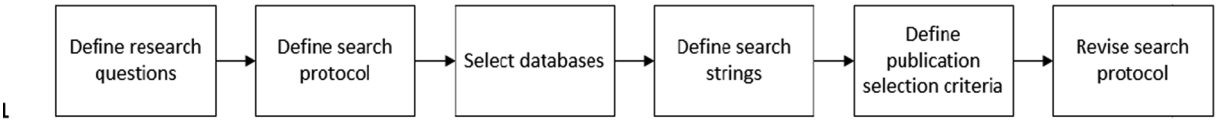


Fig. 1. Details of the Plan Review Step.

Fig. 2. Details of the Conducting Review Step.

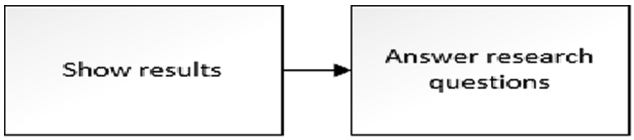


Fig. 3. Details of the Reporting Review Step.

application fields, which means that there are a lot of published studies that are probably not in the scope of this review article. The basic searching is done by an automated search. The starting input for the search was "machine learning" AND "yield prediction". Articles were retrieved, and abstracts were read to find the synonyms of the keywords. The search was performed in six databases. The search input "machine learning" AND "yield prediction" was used to get a broad view of the studies. After the exclusion criteria were applied, and all the results were processed, and a more complex search string was built in order to avoid missing relevant studies. This final search string is as follows: (("machine learning" OR "artificial intelligence") AND "data mining" AND ("yield prediction" OR "yield forecasting" OR "yield estimation")). After executing this search string, 567 studies were retrieved.

A specific description of the search strings per database are provided as follows:

Science direct: The search string is ["machine learning" AND "yield prediction"] (Title, abstract, keywords) and [(("machine learning" OR "artificial intelligence") AND "data mining" AND ("yield prediction" OR "yield forecasting" OR "yield estimation"))](Title, abstract, keywords).

Scopus: The search string is ["machine learning" AND "yield prediction"](Title, abstract, keywords) and [(("machine learning" OR "artificial intelligence") AND "data mining" AND ("yield prediction" OR "yield forecasting" OR "yield estimation"))] (Title, abstract, keywords).

Web of Science: The search string is ["machine learning" AND "yield prediction"] (title, abstract, author keywords, and Keywords Plus).

Springer Link: The search string is ["machine learning" AND "yield prediction"](anywhere) and [(("machine learning" OR "artificial intelligence") AND "data mining" AND ("yield prediction" OR "yield forecasting" OR "yield estimation"))] (anywhere)

Wiley: The search string is ["machine learning" AND "yield prediction"] (anywhere).

Google Scholar: The search string is ["machine learning" AND "yield prediction"] (anywhere) and [(("machine learning" OR "artificial intelligence") AND "data mining" AND ("yield prediction" OR "yield forecasting" OR "yield estimation"))] (anywhere).

For Web of Science and Wiley, the search string [(("machine learning" OR "artificial intelligence") AND "data mining" AND ("yield prediction" OR "yield forecasting" OR "yield estimation"))] did not result in any publications.

3.4. Exclusion criteria

To exclude irrelevant studies, the studies were analyzed and graded based on exclusion criteria to set the boundaries for the systematic review. The exclusion criteria (EC) are shown as follows:

Exclusion criteria 1 - Publication is not related to the agricultural sector and yield prediction combined with machine learning

Table 1
Distribution of papers based on the databases.

Database	# of initially retrieved papers	# of papers after exclusion criteria	Percentage of Papers (%)
Science Direct	17	4	8
Scopus	68	11	22
Web of Science	32	0	0
Springer Link	132	10	20
Wiley	20	1	2
Google Scholar	298	24	48
Total	567	50	100

Exclusion criteria 2 – Publication is not written in English
Exclusion criteria 3 – Publication that is a duplicate or already retrieved from another database
Exclusion criteria 4 – Full text of the publication is not available
Exclusion criteria 5 – Publication is a review/survey paper
Exclusion criteria 6 – Publication has been published before 2008

After the first three exclusion criteria were applied, only 77 studies remained for further analysis. After applying all the six exclusion criteria, 50 studies were selected for further analysis. In Table 1, we show the number of initially retrieved papers and the number of papers after selection criteria were applied. Fig. 4 shows the distribution of selected publications based on the databases we searched. As shown in Table 1, most of the papers were retrieved from Google Scholar, Scopus, and Springer databases.

To answer the four research questions, data from the selected studies have been extracted and synthesized. The information retrieved was focused on checking whether or not the studies meet the requirements stated in the exclusion criteria and on responding to the research questions. The selected studies that passed the exclusion criteria are presented in Appendix A. During the data synthesis, all the extracted data have been combined and synthesized, and the research questions were answered accordingly. The results are presented in Section 4.

4. Results

The selected publications are shown in Table 2. The table shows the publication year, title, and algorithms used in these papers.

Fig. 4 shows the number of publications per year published in the last ten years. This figure indicates that recently the number of papers

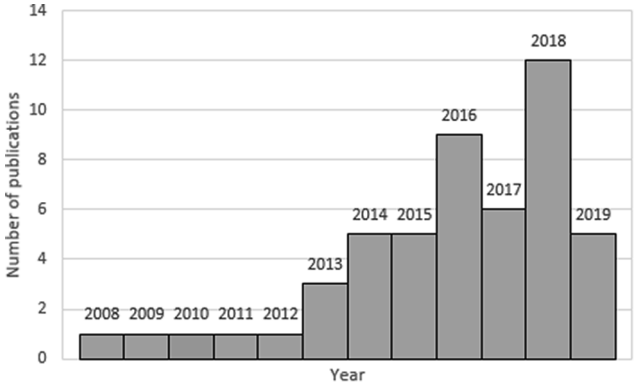


Fig. 4. Distribution of the selected publications per year.

Table 2
Selected publications.

Retrieved From	Reference	Title	Algorithm used	Year
Scopus	Ruß et al. (2008)	Data Mining with Neural Networks for Wheat Yield Prediction	Neural networks	2008
Science Direct	Everingham et al. (2009)	Ensemble data mining approaches to forecast regional sugarcane crop production	Forward stagewise algorithm	2009
Springer Link	Ruß & Kruse (2010)	Regression Models for Spatial Data: An Example from Precision Agriculture	Clustering, random forest, support vector machine	2010
Springer Link	Baral et al. (2011)	Yield Prediction Using Artificial Neural Networks	Neural networks	2011
Springer Link	Črtomir et al. (2012)	Application of Neural Networks and Image Visualization for Early Forecast of Apple Yield	Neural networks	2012
Google Scholar	Johnson (2013)	Crop yield forecasting on the Canadian Prairies by remotely sensed vegetation indices and machine learning methods	Multiple linear regression, neural networks	2013
Google Scholar	Romero et al. (2013)	Using classification algorithms for predicting durum wheat yield in the province of Buenos Aires	K-nearest neighbor, decision tree	2013
Google Scholar	Ananthara et al. (2013)	CRY - an improved crop yield prediction model using bee hive clustering approach for agricultural data sets	Clustering	2013
Scopus	Shekoofa et al. (2014)	Determining the most important physiological and agronomic traits contributing to maize grain yield through machine learning algorithms: A new avenue in intelligent agriculture	Decision tree, clustering	2014
Scopus	Gonzalez-Sanchez et al. (2014)	Predictive ability of machine learning methods for massive crop yield prediction	M5-prime regression tree, k-nearest neighbor, support vector machine	2014
Scopus	Pantazi et al. (2014)	Application of supervised self-organizing models for wheat yield prediction	Neural networks	2014
Google Scholar	Cakir et al. (2014)	Yield prediction of wheat in south-east region of Turkey by using artificial neural networks	Neural networks, multivariate polynomial regression	2014
Google Scholar	Rahman & Haq (2014)	Machine learning facilitated rice prediction in Bangladesh	Decision tree, neural networks, linear regression	2014
Scopus	Kunapuli et al. (2015)	Yield prediction for precision territorial management in maize using spectral data	Polynomial regression, logistic regression	2015
Google Scholar	Matsumura et al. (2015)	Maize yield forecasting by linear regression and artificial neural networks in Jilin, China	Neural networks, multiple linear regression	2015
Google Scholar	Ahamed et al. (2015)	Applying data mining techniques to predict annual yield of major crops and recommend planting different crops in different districts in Bangladesh	Linear regression, neural networks, clustering, k-nearest neighbor	2015
Google Scholar	Paul et al. (2015)	Analysis of soil behavior and prediction of crop yield using data mining approach	Naïve Bayes, k-nearest neighbor	2015
Science Direct	Pantazi et al. (2016)	Wheat yield prediction using machine learning and advanced sensing techniques	Neural networks	2016
Scopus	Jeong et al. (2016)	Random forests for global and regional crop yield predictions	Random forest, linear regression	2016
Wiley	Mola-Yudego et al. (2016)	Spatial yield estimates of fast-growing willow plantations for energy based on climatic variables in northern Europe	Gradient boosting tree	2016
Google Scholar	Everingham et al. (2016)	Accurate prediction of sugarcane yield using a random forest algorithm	Random forest	2016
Scopus	Gandhi et al. (2016)	Rice crop yield prediction in India using support vector machines	Support vector machine	2016
Google Scholar	Bose et al. (2016)	Spiking neural networks for crop yield estimation based on spatiotemporal analysis of image time series	Neural networks	2016
Google Scholar	Gandhi et al. (2016)	Rice crop yield prediction using artificial neural networks	Neural networks	2016
Google Scholar	Gandhi and Armstrong (2016)	Applying data mining techniques to predict yield of rice in Humid Subtropical Climatic Zone of India	Decision tree, logistic regression, k-nearest neighbor	2016
Google Scholar	Sujatha and Isakki (2016)	A study on crop yield forecasting using classification techniques	Naïve Bayes, J48, random forest, neural networks, decision tree, support vector machines *(No experimental results reported)*	2016
Google Scholar	Ying-xue et al. (2017)	Support vector machine-based open crop model (SBOCM): Case of rice production in China	Support vector machine	2017
Google Scholar	Cheng et al. (2017)	Early yield prediction using image analysis of apple fruit and tree canopy features with neural networks	Neural networks	2017
Google Scholar	Bargoti and Underwood (2017)	Image segmentation for fruit detection and yield estimation in apple orchards	Neural networks	2017
Google Scholar	Fernandes et al. (2017)	Sugarcane yield prediction in Brazil using NDVI time series and neural networks ensemble	Neural networks	2017
Google Scholar	You et al. (2017)	Deep Gaussian process for crop yield prediction based on remote sensing data	Neural networks and gaussian process, neural networks	2017
Springer Link	Osman et al. (2017)	Predicting Early Crop Production by Analysing Prior Environment Factors	Neural networks, linear regression	2017
Google Scholar	Ali et al. (2017)	Modeling managed grassland biomass estimation by using multitemporal remote sensing data machine learning approach	ANFIS, neural networks, multiple linear regression	2017
Science Direct	Kouadio et al. (2018)	Artificial intelligence approach for the prediction of Robusta coffee yield using soil fertility properties	Extreme learning machine, multiple linear regression, random forest	2018
Springer Link	Goldstein et al. (2018)	Applying machine learning on sensor data for irrigation recommendations: revealing the agronomists tacit knowledge	Gradient boosting tree, linear regression	2018
Scopus	Zhong et al. (2018)	Hierarchical modeling of seed variety yields and decision making for future planting plan	Random forest, linear regression	2018
Scopus	Crane-Droesch (2018)	Machine learning methods for crop yield prediction and climate change impact assessment in agriculture	Neural networks	2018
Scopus	Villanueva et al. (2018)	Bitter melon crop yield prediction using Machine Learning Algorithm	Neural networks	2018
Google Scholar	Girish et al. (2018)	Crop Yield and Rainfall Prediction in Tumakuru District using Machine Learning	Support vector machine, linear regression, k-nearest neighbor	2018
Google Scholar	Khanal et al. (2018)	Integration of high resolution remotely sensed data and machine learning techniques for spatial prediction of soil properties and corn yield	Neural networks, support vector machine, random forest	2018
Google Scholar	Taherei Ghazvinei et al. (2018)	Sugarcane growth prediction based on meteorological parameters using extreme learning machine and artificial neural network	Neural networks	2018
Springer Link	Ahmad et al. (2018)	Yield Forecasting of Spring Maize Using Remote Sensing and Crop Modeling in Faisalabad-Punjab Pakistan	Support vector machine, random forest, decision tree	2018

(*continued on next page*)

Table 2 (*continued*)

Retrieved From	Reference	Title	Algorithm used	Year
Springer Link	Shah et al. (2018)	Smart Farming System: Crop Yield Prediction Using Regression Techniques	Support vector machine, random forest, multivariate polynomial regression	2018
Springer Link	Monga (2018)	Estimating Vineyard Grape Yield from Images	Neural networks	2018
Google Scholar	Wang et al. (2018)	Deep transfer learning for crop yield prediction with remote sensing data	Neural networks	2018
Science Direct	Xu et al. (2019)	Design of an integrated climatic assessment indicator (ICAI) for wheat production: A case study in Jiangsu Province, China	Random forest, support vector machine	2019
Scopus	Filippi et al. (2019b)	An approach to forecast grain crop yield using multi-layered, multi-farm data sets and machine learning	Random forest	2019
Google Scholar	Rao & Manasa (2019)	Artificial Neural Networks for Soil Quality and Crop Yield Prediction using Machine Learning	Neural networks	2019
Springer Link	Ranjan & Parida (2019)	Paddy acreage mapping and yield prediction using sentinel-based optical and SAR data in Sahibganj district, Jharkhand (India)	Linear regression	2019
Springer Link	Charoen-Ung & Mittrapiyanuruk (2019)	Sugarcane Yield Grade Prediction Using Random Forest with Forward Feature Selection and Hyper-parameter Tuning	Random forest	2019

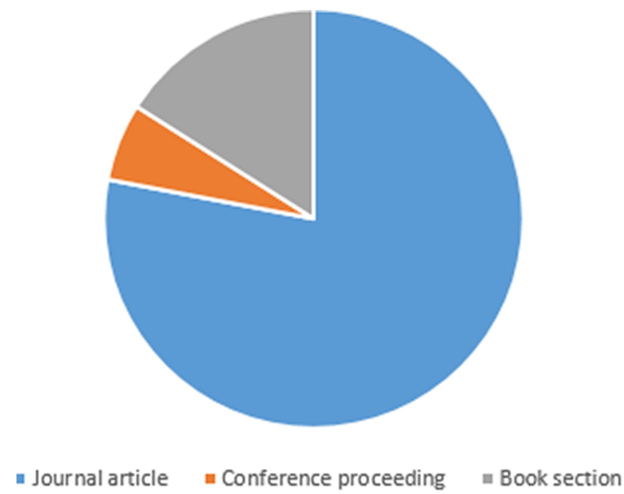


Fig. 5. Distribution of the type of 50 primary publications.

on crop yield prediction is increasing.

There were no exclusion criteria based on the type of publication; therefore, conference papers were also included. The pie chart in Fig. 5 shows the distribution of types of publications. The figure shows that most of the articles we accessed are journal articles; conference papers and book chapters constitute less than 25% of the total number of papers.

To address research question two (RQ2), features used in the machine learning algorithms applied in the papers were investigated and summarized. All features we were able to extract are shown in Table 3.

As shown in Table 3, the most used features are related to temperature, rainfall, and soil type. Crop yield is the dependent variable. To get a better overview of the independent variables (features), the features were grouped. The independent features can be grouped into soil and crop information, humidity, nutrients, and field management. The number of times these groups are used is presented in Table 4. As shown in this table, the feature groups that are most used are related to the soil, solar, and humidity information.

The feature group "soil information" consists of the following variables: soil maps, soil type, pH value, cation exchange capacity, and area of production. Whether or not soil maps were used and the information content of the maps differs among the different publications. In the soil maps, general information about the nutrients in the soil, type of the soil, and location can be found. Crop information refers to information about the crop itself, such as weight, growth during the growth-process, variety of plants, and crop density. Other measurements that indicate growth is also included in this group, for example, the leaf area index. Humidity stands for the water in the field. The features that fall under the humidity group include rainfall, humidity, forecasted rainfall, and precipitation. Nutrients can be nutrients that are already in the soil, but the nutrients can also be applied nutrients. These features measure the level of saturation. The measured nutrients are nitrogen, magnesium, potassium, sulphur, zinc, boron, calcium, manganese, and phosphorus. With field management, decisions of farmers to adjust their field are grouped. These features are irrigation and fertilization, and thus field management could also refer to the management of nutrients. The solar information contains features related to radiation or temperature. These are gamma radiometric, temperature, photoperiod, shortwave radiation, degree-days, and solar radiation. The feature group labeled as 'Other' contains the features that cannot be put in any of the groups mentioned above. Most of these features are used only once or are calculated features (Measuring Vegetation (NDVI & EVI), 2000). These features are used less and include features such as wind speed, pressure, and images. The calculated features are MODIS Enhanced Vegetation Index (MODIS-EVI), Normalized Vegetation Index (NDVI), and Enhanced Vegetation Index

Table 3
All features used.

Feature	# of times used
Temperature	24
Soil type	17
Rainfall	17
Crop information	13
Soil maps	12
Humidity	11
pH-value	11
Solar radiation	10
Precipitation	9
Images	8
Area of production	8
Fertilization	7
NDVI	6
Cation exchange capacity	6
Nitrogen	6
Irrigation	5
Potassium	5
Wind speed	5
Zinc	3
Magnesium	3
Shortwave radiation	2
Sulphur	2
Boron	2
Calcium	2
Organic carbon	2
EVI	2
Phosphorus	2
Gamma radiametrics	1
MODIS-EVI	1
Forecasted rainfall	1
Photoperiod	1
Climate	1
Degree-days	1
Time	1
Pressure	1
Leaf area index	1
Manganese	1

Table 4
Grouped features.

Group	# of times used
Soil information	54
Solar information	39
Humidity	38
Nutrients	28
Other	24
Crop information	14
Field management	12

(EVI) (Filippi et al., 2019a).

To represent all the features gathered through this SLR study, we drew a feature map depicted in Fig. 6 shows the significant features and sub-features.

To address the first research question (RQ1), machine learning algorithms were investigated and summarized. The algorithms used more than once are listed in Table 5. As shown in the table, Neural Networks (NN) and Linear Regression algorithms are the two algorithms used mostly. Also, Random Forest (RF) and Support Vector Machines (SVM) are widely used, according to Table 5.

To address research question three (RQ3), evaluation parameters were identified. All the evaluation parameters that were used and the number of times they were used are shown in Table 6. As the table shows, Root Mean Square Error (RMSE) is the most used parameter in the studies.

Apart from the evaluation parameters, several validation approaches were used as well. Most of the time, cross-validation is used. The most used evaluation method was 10-fold cross-validation.

To address research question four (RQ4), the publications were read to see if they stated any problems or improvements for future models. In several studies, insufficient availability of data (too few data) was mentioned as a problem. The studies stated that their systems worked for the limited data that they had at hand, and indicated data with more variety should be used for further testing. This means data with different climatic circumstances, different vegetation, and longer time-series of yield data. Another suggested improvement is that more data sources should be integrated. Finally, the publication indicated that the use of machine learning in farm management systems should be explored. If the models work as requested, software applications must be created that allow the farmer to make decisions based on the models.

5. Deep learning-based crop yield prediction

In the first part of our research (i.e., Systematic Literature Review), we observed that Artificial Neural Networks (ANN) is the most used algorithm for crop yield prediction. Recently, deep learning, which is a sub-branch of machine learning, has provided state-of-the-art results in many different domains, such as face recognition and image classification. These Deep Neural Networks (DNN) algorithms use similar concepts of ANN algorithms; however, they include different hidden layer types such as convolutional layer and pooling layer and consist of many hidden layers instead of a single hidden layer.

As such, in the second part of our research, we aimed to investigate to what extent deep learning algorithms have been applied in crop yield prediction. To broaden our analysis and reach recent applications of deep learning algorithms in yield prediction, we designed a new search criterion (i.e., “deep learning” AND “yield prediction”) and performed a new search in the same electronic databases that were used during the SLR study. We reached the following 30 papers shown in Table 7. We investigated these articles in detail, extracted, and synthesized the deep learning algorithms applied by researchers.

Fig. 7 shows the yearly distribution of deep learning-based papers. Although we are in the half of the year 2020, the number of papers that belong to the year 2020 is now equal to the number of papers published in 2019. This shows that the number of papers is increasing every year.

In Table 8, we show the distribution of deep learning-based papers per database. Most of the papers were retrieved from Google Scholar, and the second top database was Scopus. Science Direct and Springer Link returned a similar number of deep learning-based papers.

In Table 9, we show the distribution of applied deep learning algorithms in the identified papers list. The most applied deep learning algorithm is Convolutional Neural Networks (CNN), and the other widely used algorithms are Long-Short Term Memory (LSTM) and Deep Neural Networks (DNN) algorithms. Since some papers applied more than one deep learning algorithm, the total number of usages shown in the second column is larger than the total number of papers.

These deep learning algorithms are shortly described as follows:

- *Deep Neural Networks (DNN):* These DNN algorithms are very similar to the traditional Artificial Neural Networks (ANN) algorithms except the number of hidden layers. In DNN networks, there are many hidden layers that are mostly fully connected, as in the case of ANN algorithms. However, for other kinds of deep learning algorithms such as CNN, there are also different types of layers, such as the convolutional layer and the pooling layer.
- *Convolutional Neural Networks (CNN):* Compared to a fully connected network, CNN has fewer parameters to learn. There are three types of layers in a CNN model, namely convolutional layers, pooling layers, and fully-connected layers. Convolutional layers consist of filters and feature maps. Filters are the neurons of the layer, have weighted inputs, and create an output value (Brownlee, 2016). A feature map can be considered as the output of one filter. Pooling layers are applied to down-sample the feature map of the previous layers, generalize feature representations, and reduce the

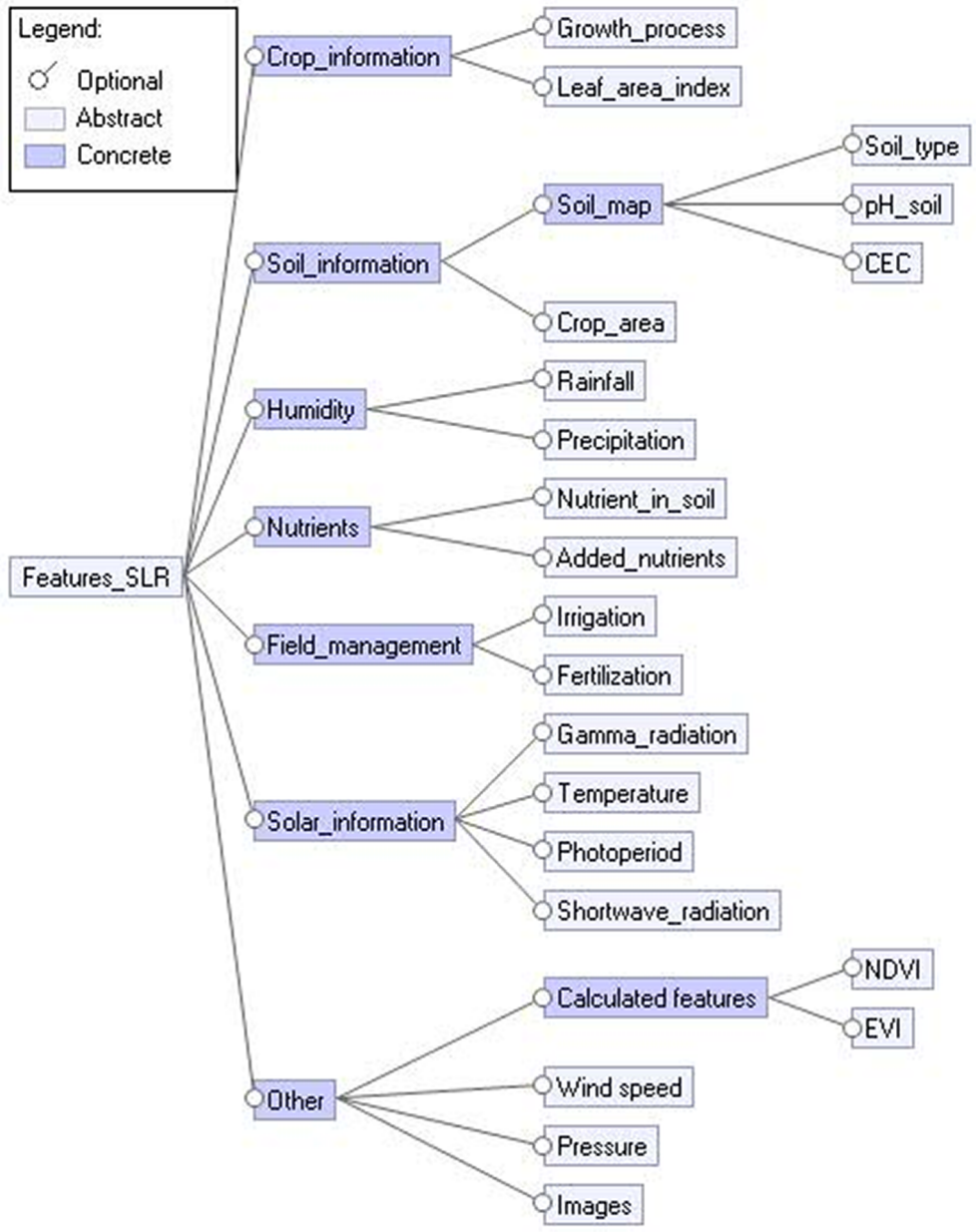


Fig. 6. Feature diagram.

Table 5
Most used machine learning algorithms.

Most used machine learning algorithms	# of times used
Neural Networks	27
Linear Regression	14
Random Forest	12
Support Vector Machine	10
Gradient Boosting Tree	4

Table 6
All evaluation parameters used.

Key	Evaluation parameter	# of times used
RMSE	Root mean square error	29
R^2	R-squared	19
MAE	Mean absolute error	8
MSE	Mean square error	5
MAPE	Mean absolute percentage error	3
RSAE	Reduced simple average ensemble	3
LCCC	Lin's concordance correlation coefficient	1
MFE	Multi factored evaluation	1
SAE	Simple average ensemble	1
rcv	Reference change values	1
MCC	Matthew's correlation coefficient	1

overfitting (Brownlee, 2019). Fully-connected layers are mostly used at the end of the network for predictions. The general pattern for CNN models is that one or more convolutional layers are followed by a pooling layer, and this structure is repeated several times, and finally, fully connected layers are applied (Brownlee, 2016, 2019).

- *Long-Short Term Memory (LSTM):* LSTM networks were designed specifically for sequence prediction problems. There are several LSTM architectures (Brownlee, 2017), namely vanilla LSTM, stacked LSTM, CNN-LSTM, Encoder-Decoder LSTM, Bidirectional LSTM, and Generative LSTM. There are several limitations of Multi-Layer

Table 7
Deep learning-based publications.

Retrieved From	Reference	Title	Deep Learning Algorithm(s) used	Year
Science Direct	Schwalbert et al. (2020)	Satellite-based soybean yield forecast: Integrating machine learning and weather data for improving crop yield prediction in southern Brazil	Long-Short Term Memory (LSTM)	2020
Science Direct	Chu and Yu (2020)	An end-to-end model for rice yield prediction using deep learning fusion	The combination of Back-Propagation Neural Networks (BPNNs) and Independently Recurrent Neural Network (IndRNN)	2020
Science Direct	Tedesco-Oliveira et al. (2020)	Convolutional neural networks in predicting cotton yield from images of commercial fields	Convolutional Neural Networks (CNN)	2020
Science Direct	Nevavuori et al. (2019)	Crop yield prediction with deep convolutional neural networks	Convolutional Neural Networks (CNN)	2019
Science Direct	Maimaitijiang et al. (2020)	Soybean yield prediction from UAV using multimodal data fusion and deep learning	Deep Neural Networks (DNN)	2020
Science Direct	Yang et al. (2019)	Deep convolutional neural networks for rice grain yield estimation at the ripening stage using UAV-based remotely sensed images	Convolutional Neural Networks (CNN)	2019
Google Scholar	Khaki and Wang (2019)	Crop Yield Prediction Using Deep Neural Networks	Deep Neural Networks (DNN)	2019
Google Scholar	Rahnemoonfar and Sheppard (2017)	Real-time yield estimation based on deep learning	Convolutional Neural Networks (CNN)	2017
Google Scholar	Chen et al. (2019)	Strawberry Yield Prediction Based on a Deep Neural Network Using High-Resolution Aerial Orthoimages	Faster Region-based Convolutional Neural Networks (Faster R-CNN)	2019
Google Scholar	Sun et al. (2019)	County-Level Soybean Yield Prediction Using Deep CNN-LSTM Model	The combination of Convolutional Neural Networks and Long-Short Term Memory Networks (CNN-LSTM)	2019
Google Scholar	Khaki et al. (2020)	A CNN-RNN Framework for Crop Yield Prediction	The combination of Convolutional Neural Networks and Recurrent Neural Networks (CNN-RNN)	2020
Google Scholar	Terliksiz and Altýlar (2019)	Use Of Deep Neural Networks For Crop Yield Prediction: A Case Study Of Soybean Yield in Lauderdale County, Alabama, USA	3D Convolutional Neural Networks (3D CNN)	2019
Google Scholar	Lee et al. (2019)	A Self-Predictable Crop Yield Platform (SCYP) Based On Crop Diseases Using Deep Learning	Convolutional Neural Networks (CNN)	2019
Google Scholar	Elavarasan and Vincent (2020)	Crop Yield Prediction Using Deep Reinforcement Learning Model for Sustainable Agrarian Applications	Deep Recurrent Q-Network	2020
Google Scholar	Wang et al. (2020)	Winter Wheat Yield Prediction at County Level and Uncertainty Analysis in Main Wheat-Producing Regions of China with Deep Learning Approaches	The combination of Convolutional Neural Networks and Long-Short Term Memory (CNN-LSTM)	2020
Google Scholar	Wolanin et al. (2020)	Estimating and understanding crop yields with explainable deeplearning in the Indian Wheat Belt	Convolutional Neural Networks (CNN)	2020
Springer Link	Bhojani and Bhatt (2020)	Wheat crop yield prediction using new activation functions in neuralnetwork	Deep Neural Networks (DNN)	2020
Springer Link	Fathi et al. (2019)	Crop Yield Prediction Using Deep Learning in Mediterranean Region	Deep Neural Networks (DNN)	2019
Springer Link	Shidnal et al. (2019)	Crop yield prediction: two-tiered machine learning model approach	Convolutional Neural Networks (CNN)	2019
Springer Link	Khaki and Wang (2019)	Crop Yield Prediction Using Deep Neural Networks	Deep Neural Networks (DNN)	2019
Springer Link	Nguyen et al. (2019)	Spatial-Temporal Multi-Task Learningfor Within-Field Cotton Yield Prediction	Spatial-Temporal Multi-Task Learning	2019
Springer Link	De Alwis et al. (2019)	Duo Attention with Deep Learning on Tomato Yield Prediction and Factor Interpretation	Duo Attention Long-Short Term Memory	2019
Wiley	Jiang et al. (2020)	A deep learning approach to conflating heterogeneous geospatial data for corn yield estimation: A case study of the US Corn Belt at the county level	Long-Short Term Memory (LSTM)	2020
Scopus	Saravi et al. (2019)	Quantitative model of irrigation effect on maize yield by deep neural network	Deep Neural Networks (DNN)	2019
Scopus	Zhang et al. (2020)	Combining Optical, Fluorescence, Thermal Satellite, and Environmental Data to Predict County-Level Maize Yield in China Using Machine Learning Approaches	Long-Short Term Memory (LSTM)	2020
Scopus	Kang et al. (2020)	Comparative assessment of environmental variables and machine learning algorithms for maize yield prediction in the US Midwest	Long-Short Term Memory (LSTM) and Convolutional Neural Networks (CNN)	2020
Scopus	Wang et al. (2020)	Combining Multi-Source Data and Machine Learning Approaches to Predict Winter Wheat Yield in the Conterminous United States	Deep Neural Networks (DNN)	2020
Scopus	Ju et al. (2020)	Machine learning approaches for crop yield prediction with MODIS and weather data	Long-Short Term Memory (LSTM) Convolutional Neural Networks (CNN), Stacked-Sparse AutoEncoder (SSAE)	2020
Scopus	Yalcin (2019)	An Approximation for A Relative Crop Yield Estimate from Field Images Using Deep Learning	Convolutional Neural Networks (CNN)	2019
Scopus	Wang et al. (2018)	Deep Transfer Learning for Crop Yield Prediction with Remote Sensing Data	Long-Short Term Memory (LSTM) for Transfer Learning	2018

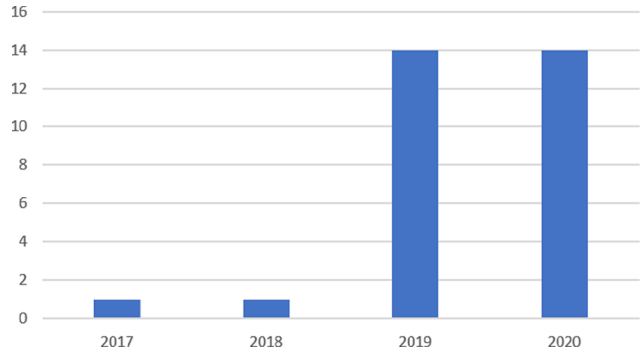


Fig. 7. Yearly distribution of deep learning-based papers.

Table 8
Distribution of deep learning-based papers per database.

Database	# of papers	Percentage of Papers (%)
Science Direct	6	20
Scopus	7	23,33
Web of Science	0	0
Springer Link	6	20
Wiley	1	3,33
Google Scholar	10	33,33
Total	30	100

Table 9
Distribution of deep learning algorithms.

Algorithms used	# of usages	Percentage (%)
CNN	10	30,30
LSTM	7	21,21
DNN	7	21,21
Hybrid	4	12,12
Autoencoder	1	3,03
Multi-Task Learning (MTL)	1	3,03
Deep Recurrent Q-Network (DQN)	1	3,03
3D CNN	1	3,03
Faster R-CNN	1	3,03
Total	33	100

Perceptron (MLP) feedforward ANN algorithms, such as being stateless, unaware of temporal structure, messy scaling, fixed sized inputs, and fixed-sized outputs (Brownlee, 2017). Compared to the MLP network, LSTM can be considered as the addition of loops to the network. Also, LSTM is a special type of Recurrent Neural Network (RNN) algorithm. Since LSTM has an internal state, is aware of the temporal structure in the inputs, can model parallel input series, can process variable-length input to generate variable-length output, they are very different than the MLP networks. The memory cell is the computational unit of the LSTM (Brownlee, 2017). These cells consist of weights (i.e., input weights, output weights, and internal state) and gates (i.e., forget gate, input gate, and output gate).

- *3D CNN:* This network is a special type of CNN model in which the kernels move through height, length, and depth. As such, it produces 3D activation maps. This type of model was developed to improve the identification of moving, as in the case of security cameras and medical scans. 3D convolutions are performed in the convolutional layers of CNN (Ji et al., 2012).
- *Faster R-CNN:* The Region-Based Convolutional Neural Network (R-CNN) is a family of CNN models that were designed specifically for object detection (Brownlee, 2019). There are four variations of R-CNN, namely R-CNN, Fast R-CNN, Faster R-CNN, and Mask R-CNN. In Faster R-CNN, a Region Proposal Network is added to interpret features extracted from CNN (Ren et al., 2015).
- *Autoencoder:* Autoencoders are unsupervised learning approaches that consist of the following four main parts: encoder, bottleneck, decoder, and reconstruction loss. The architecture of autoencoders can be designed based on simple feedforward neural networks, CNN, or LSTM networks (Baldi, 2012; Vincent et al., 2008).
- *Hybrid networks:* It is possible to combine the power of different deep learning algorithms. As such, researchers combine different algorithms in a different way. Chu and Yu (2020) combined Back-Propagation Neural Networks (BPNNs) and Independently Recurrent Neural Network (IndRNN) and applied this model for crop yield prediction. Sun et al. (2019) combined Convolutional Neural Networks and Long-Short Term Memory Networks (CNN-LSTM) for soybean yield prediction. Khaki et al. (2020) combined Convolutional Neural Networks and Recurrent Neural Networks (CNN-RNN) for yield prediction. Wang et al. (2020) combined CNN and LSTM (CNN-LSTM) networks for the wheat yield prediction problem.
- *Multi-Task Learning (MTL):* In multi-task learning, we share representations between tasks to improve the performance of our models developed for these tasks (Ruder, 2017). It has been applied in many different domains, such as drug discovery, speech recognition, and natural language processing. The aim is to improve the performance of all the tasks involved instead of improving the performance of a single task. Zhang and Yang (2017) reviewed several multi-task learning approaches for supervised learning tasks and also explained how to combine multi-task learning with other learning categories, such as semi-supervised learning and reinforcement learning. They divided supervised MTL approaches into the following categories: feature learning approach, low-rank approach, task clustering approach, task relation learning approach, and decomposition approach.
- *Deep Recurrent Q-Network (DQN):* In reinforcement learning, agents observe the environment and act based on some rules and the available data. Agents get rewards based on their actions (i.e., positive or negative reward) and try to maximize this reward. The environment and agents interact with each other continuously. DQN algorithm was developed in 2015 by the researchers of DeepMind acquired by Google in 2014. This DQN algorithm that combines the power of reinforcement learning and deep neural networks solved several Atari games in 2015. The classical Q-learning algorithm was enhanced with deep neural networks, and also, the experience replay technique was integrated (Mnih et al., 2015). Elavarasan and Vincent (2020) applied this algorithm for crop yield prediction.

The number of papers that apply deep learning for crop yield prediction is increasing. As such, we expect to see more research in this direction.

6. Discussion

- **General discussion:** Such research is susceptible to threats to validity, and potential threats to validity can be external, construct validity, and reliability (Šmite et al., 2010). The external validity and construct validity are addressed for this SLR study since the initial search string was broad, and the query returned a substantial number of studies: 567 publications in total. The search string covered the whole scope of the SLR. For reliability of the SLR, the validity can be considered well-addressed since the process of the SLR has been described clearly and is replicable. If this SLR is replicated, it could return slightly different selected publications, but the differences would be a result of different personal judgments. However, it is highly unlikely that the overall findings would change.
- **Search-related discussion:** There is a possibility that valuable publications might have been missed. More synonyms could have been used, and a broader search could have returned new studies. However, the search string resulted in a high number of publications

indicating a broad enough search.

- **Analysis-related discussion:** Another issue that could be a threat to validity the way the analysis is conducted. For example, not all publications stated what kind of evaluation parameters were used, and sometimes just a few examples of features were explained. Thus, sometimes this information that is required to address the research questions could not be found in the paper. This way, the data that was used to answer the research questions were derived from a few numbers of publications than a total of 50 selected publications. To get more information about the publications, the authors could potentially have been contacted, but this line of action was not feasible within the context of this research, and that might also not solve all the issues.
- **RQ1-Related (algorithms) discussion:** Linear Regression is the second most used algorithms, according to Table 5. Linear Regression is used as a benchmarking algorithm in most cases to check whether the proposed algorithm is better than Linear Regression or not. Therefore, although it is shown in many articles, it does not mean that it is the best performing algorithm. Table 5 should be interpreted carefully because "most used" does not mean the best-performing ones. In fact, Deep Learning (DL), which is a sub-branch of Machine Learning, has been used for the crop yield prediction problem recently and is believed to be very promising. In this study, we also identified several deep learning-based studies. There are several additional promising aspects of DL methods, such as automatic feature extraction and superior performance. We expect that more research will be conducted on the use of DL approaches in crop yield prediction in the near future due to the superior performance of DL algorithms in other problem domains.

Among the selected publications, both classifiers and clustering algorithms are used. Since pictures are used for clustering in those publications, the publication is in connection with the machine vision instead of ML using a numerical dataset. The use of clustering algorithms for this problem can be investigated in detail to find different research perspectives in this problem.

- **RQ2-related (features) discussion:** Groups are created for features and algorithms to visualize the main features and algorithms. Due to this decision, detailed information is lost, but clarity has been maintained. The most used features are soil type, rainfall, and temperature. Apart from those features that are used in several studies, there are also features that were used in specific studies. Those features are gamma radiation, MODIS-EVI, forecast rainfall, humidity, photoperiod, pH-value, irrigation, leaf area, NDVI, EVI, and crop information. There are also studies that use different nutrients as features, which are magnesium, potassium, sulphur, zinc, nitrogen, boron, and calcium. The most used features are not always the same kind of data. Temperature, for example, is measured as average temperature, but more features like maximum temperature and minimum temperature are also applied.
- **RQ3-related (evaluation parameters and approaches) discussion:** There are not many evaluation parameters reported in the selected papers. Almost every study used RMSE as the measurement of the quality of the model. Other evaluation parameters are MSE, R^2, and MAE. Some parameters were used in specific studies, most of these parameters look like some of the previously mentioned parameters, with a small difference. These are MAPE, LCCC, MFE, SAE, rcv, RSAE, and MCC. Most of the models had outcomes with high accuracy values for their evaluation parameters, which means that the model made correct predictions. As the evaluation approach, the 10-fold cross-validation approach was preferred by researchers.
- **RQ4-related (challenges) discussion:** Challenges were reported based on the explicit statements in the articles. However, there might be additional challenges that were not stated in the identified papers. The challenges are mainly in the field of improvement of a working model. When more data is gathered to train and test, much more can be said about the precision of the model. Another challenge is the implementation of the models into the farm management systems. When applications are made that the farmer can use, then only can the models be useful to make decisions, also during the growing season. When specific parameters for that specific place are measured and added, predictions will have higher precision.

7. Conclusion

This study showed that the selected publications use a variety of features, depending on the scope of the research and the availability of data. Every paper investigates yield prediction with machine learning but differs from the features. The studies also differ in scale, geological position, and crop. The choice of features is dependent on the availability of the dataset and the aim of the research. Studies also stated that models with more features did not always provide the best performance for the yield prediction. To find the best performing model, models with more and fewer features should be tested. Many algorithms have been used in different studies. The results show that no specific conclusion can be drawn as to what the best model is, but they clearly show that some machine learning models are used more than the others. The most used models are the random forest, neural networks, linear regression, and gradient boosting tree. Most of the studies used a variety of machine learning models to test which model had the best prediction.

Since Neural Networks is the most applied algorithm, we also aimed to investigate to what extent deep learning algorithms were used for crop yield prediction. After the identification of 30 papers that applied deep learning, we extracted and synthesized the applied algorithms. We observed that CNN, LSTM, and DNN algorithms are the most preferred deep learning algorithms. However, there are also other kinds of algorithms applied to this problem. We consider that this article will pave the way for further research on the development of crop yield prediction problem.

In our future work, we aim to build on the outcomes of this study and focus on the development of a DL-based crop yield prediction model.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

In Table A1, features used per publications are shown. If there is a '1' in the box, it means that that specific feature was used.
In Table A2, the evaluation parameters used per publication are presented.

Table A1
Features used per selected publication.

Paper	Soil type	Gamma radia-metrics	Soil maps	MODIS-EVI	Rainfall	Forecas-ted rainfall	Precipi-tation	Temper-ature	Humidi-ty	Photop-eriod	Fertiliz-ation	Climate	pH-value	Irrigati-on	Cation ex-change capacity	Magnes-ium	Potassi-um	Area of produc-tion	Wind Speed
Filippi et al., 2019	1	1	1	1	1	1													
Jeong et al., 2016	1		1		1				1	1	1	1							
Zhong et al., 2018	1				1			1					1						
Villanueva and Salenga, 2018																			
Crane-Droesch, 2018	1		1		1			1	1					1					1
Gonzalez-Sanchez et al., 2014	1		1		1			1						1					
Xu et al., 2019							1	1											1
Pantazi et al., 2016	1		1										1						
Kouadio et al., 2018	1		1										1			1	1		
Kunapuli et al., 2015											1							1	
Shekoofa et al., 2014	1										1								
Pantazi et al., 2014			1										1		1	1			
Goldstein et al., 2018	1							1	1										
Mola-Yudego et al., 2016							1	1											
Girish et al., 2018					1														
Rao and Manasa, 2019	1		1					1			1		1				1		
Khanal et al., 2018	1		1										1		1	1	1		
Cheng et al., 2017																			
Everingham et al., 2009					1			1											
Everingham et al., 2016					1			1											
Bargoti and Underwood, 2017																			
Fernandes and Ebecken, 2017								1											
Johnson et al., 2013																			
Matsumura et al., 2015							1	1			1								
Taherei Ghazvinei, 2018					1			1	1					1	1				

(*continued on next page*)

Table A1 (*continued*)

Paper	Soil type	Gamma radiametrics	Soil maps	MODIS-EVI	Rainfall	Forecasted rainfall	Precipitation	Temperature	Humidity	Photoperiod	Fertilization	Climate	pH-value	Irrigation	Cation exchange capacity	Magnesium	Potassium	Area of production	Wind Speed
Romero et al., 2013																			
Su et al., 2017	1		1				1	1	1				1				1		1
You et al., 2017																			
Ahmad et al., 2018	1														1				
Črtomir et al., 2012																			
Osman et al., 2017					1			1	1									1	1
Ranjan and Parida, 2019			1															1	
Shah et al., 2018					1			1	1										
Russ et al., 2008											1				1				
Monga, 2018																			
Russ and Kruse, 2010			1								1				1				
Baral et al., 2011					1		1	1											
Ahamed et al., 2015					1			1	1				1	1				1	
Ali et al., 2017							1	1											
Cakir et al., 2014					1			1	1										1
Gandhi et al., 2016							1	1										1	
Wang et al., 2018																			
Charoen-Ung and Mittrapiyanuruk, 2019	1				1									1				1	
Ananthara et al., 2013	1				1				1				1						
Bose et al., 2016								1											
Gandhi et al., 2016							1	1										1	
Gandhi and Armstrong, 2016	1						1	1										1	
Paul et al., 2015													1				1		
Rahman and Haq, 2014					1			1	1										
Sujatha and Isakki, 2016	1												1						

Paper	Shortwave radiation	Degree-days	Time	Solar radiation	Pressure	Sulphur	Zinc	Nitrogen	Boron	Calcium	Crop Information	Leaf Area Index	Phosphorus	Manganese	Organic carbon	Images	NDVI	EVI
Filippi et al., 2019																		
Jeong et al., 2016																		
Zhong et al., 2018				1														
Villanueva and Salenga, 2018																		
Crane-Droesch, 2018	1	1	1															

(*continued on next page*)

Table A1 (*continued*)

Paper	Shortw-ave radia-tion	Degree-days	Time	Solar radia-tion	Pressure	Sulphur	Zinc	Nitroge-n	Boron	Calcium	Crop Inform-ation	Leaf Area Index	Phosph-orus	Manga-nese	Organic carbon	Images	NDVI	EVI
Gonzalez-Sanchez et al., 2014				1														
Xu et al., 2019				1														
Pantazi et al., 2016																		
Kouadio et al., 2018						1	1	1	1	1								
Kunapuli et al., 2015											1	1						
Shekoofa et al., 2014											1							
Pantazi et al., 2014								1		1					1			
Goldstein et al., 2018				1							1							
Mola-Yudego et al., 2016																		
Girish et al., 2018																		
Rao and Manasa, 2019							1	1	1		1		1	1				
Khanal et al., 2018																		
Cheng et al., 2017											1							
Everingham et al., 2009				1														
Everingham et al., 2016				1	1													
Bargoti and Underwood, 2017																1		
Fernandes and Ebecken, 2017																	1	
Johnson et al., 2013																	1	1
Matsumura et al., 2015																		
Taherei Ghazvinei, 2018	1																	
Romero et al., 2013											1							
Su et al., 2017													1					

(*continued on next page*)

Table A1 (*continued*)

Paper	Shortw-ave radia-tion	Degree-days	Time	Solar radia-tion	Pressure	Sulphur	Zinc	Nitroge-n	Boron	Calcium	Crop Inform-ation	Leaf Area Index	Phosph-orus	Manga-nese	Organic carbon	Images	NDVI	EVI
You et al., 2017																1		
Ahmad et al., 2018								1			1				1	1	1	
Črtomir et al., 2012																1		
Osman et al., 2017											1							
Ranjan and Parida, 2019											1							
Shah et al., 2018																		
Russ et al., 2008								1										
Monga, 2018																1		
Russ and Kruse, 2010																		
Baral et al., 2011																		
Ahamed et al., 2015				1														
Ali et al., 2017																1	1	1
Cakir et al., 2014				1														
Gandhi et al., 2016											1							
Wang et al., 2018																1	1	
Charoen-Ung and Mittrapiyanur-uk, 2019																		
Ananthara et al., 2013				1							1							
Bose et al., 2016																1	1	
Gandhi et al., 2016											1							
Gandhi and Armstrong, 2016																		
Paul et al., 2015						1	1	1										
Rahman and Haq, 2014				1														
Sujatha and Isakki, 2016											1							

Table A2
Evaluation parameters used per publication.

Paper	Root mean square error	Lin's concordance correlation coefficient	Mean square error	R-squared	Mean absolute error	Mean absolute percentage error	Multi factored evaluation	Simple average ensemble	Reference change values	Reduced simple average ensemble	Matthew's correlation coefficient
Filippi et al., 2019	1	1	1								
Jeong et al., 2016	1										
Zhong et al., 2018	1				1						
Villanueva and Salenga, 2018											
Crane-Droesch, 2018			1								
Gonzalez-Sanchez et al., 2014	1			1	1						
Xu et al., 2019	1			1							
Pantazi et al., 2016	1										
Kouadio et al., 2018	1				1						
Kunapuli et al., 2015				1							
Shekoofa et al., 2014											
Pantazi et al., 2014											
Goldstein et al., 2018	1										
Mola-Yudego et al., 2016	1			1							
Girish et al., 2018											
Rao and Manasa, 2019											
Khanal et al., 2018	1			1							
Cheng et al., 2017	1			1		1	1				
Everingham et al., 2009								1	1	1	
Everingham et al., 2016				1							
Bargoti and Underwood, 2017				1							
Fernandes and Ebecken, 2017	1			1							
Johnson et al., 2013											
Matsumura et al., 2015	1		1								
Taherei Ghazvinei, 2018	1			1							
Romero et al., 2013											
Su et al., 2017	1										
You et al., 2017	1										
Ahmad et al., 2018	1			1	1	1					
Črtomir et al., 2012				1							

(*continued on next page*)

Table A2 (*continued*)

Paper	Root mean square error	Lin's concordance correlation coefficient	Mean square error	R-squared	Mean absolute error	Mean absolute percentage error	Multi factored evaluation	Simple average ensemble	Reference change values	Reduced simple average ensemble	Matthew's correlation coefficient
Osman et al., 2017	1										
Ranjan and Parida, 2019											
Shah et al., 2018	1			1	1						
Russ et al., 2008			1	1							
Monga, 2018				1	1						
Russ and Kruse, 2010	1										
Baral et al., 2011											
Ahamed et al., 2015	1										
Ali et al., 2017	1			1							
Cakir et al., 2014	1										
Gandhi et al., 2016	1			1	1					1	
Wang et al., 2018	1			1							
Charoen-Ung and Mittrapiyanuruk, 2019											
Ananthara et al., 2013											
Bose et al., 2016	1		1			1					
Gandhi et al., 2016	1			1							1
Gandhi and Armstrong, 2016	1				1					1	
Paul et al., 2015											
Rahman and Haq, 2014	1										
Sujatha and Isakki, 2016											

Appendix B. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.compag.2020.105709.

References

Ahamed, A.T.M.S., Mahmood, N.T., Hossain, N., Kabir, M.T., Das, K., Rahman, F., Rahman, R.M., 2015. Applying data mining techniques to predict annual yield of major crops and recommend planting different crops in different districts in Bangladesh. In: 2015 IEEE/ACIS 16th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, SNPD 2015 - Proceedings, https://doi.org/10.1109/SNPD.2015.7176185.

Ahmad, I., Saeed, U., Fahad, M., Ullah, A., Habib-ur-Rahman, M., Ahmad, A., Judge, J., 2018. Yield forecasting of spring maize using remote sensing and crop modeling in Faisalabad-Punjab Pakistan. J. Indian Soc. Remote Sens. 46 (10), 1701–1711. https://doi.org/10.1007/s12524-018-0825-8.

Ali, I., Cawkwell, F., Dwyer, E., Green, S., 2017. Modeling managed grassland biomass estimation by using multitemporal remote sensing data—a machine learning approach. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 10 (7), 3254–3264. https://doi.org/10.1109/JSTARS.2016.2561618.

Alpaydin, E., 2010. Introduction to Machine Learning, 2nd ed. Retrieved from https://books.google.nl/books?hl=nl&lr=&id=TtrxCwAAQBAJ&oi=fnd&pg=PR7&dq=introduction+to+machine+learning&ots=T5ejQG_7pZ&sig=0xC_H0agN7mPhYW7oQsWiMVwRnQ#v=onepage&q=introduction to machine learning&f=false.

Ananthara, M.G., Arunkumar, T., Hemavathy, R., 2013. CRY-An improved crop yield prediction model using bee hive clustering approach for agricultural data sets. In: Proceedings of the 2013 International Conference on Pattern Recognition, Informatics and Mobile Engineering, PRIME 2013, 473–478. https://doi.org/10.1109/ICPRIME.2013.6496717.

Ayodele, T.O., 2010. Introduction to Machine Learning.

Baldi, P., 2012. Autoencoders, unsupervised learning, and deep architectures. In: Proceedings of ICML workshop on unsupervised and transfer learning, pp. 37–49.

Baral, S., Kumar Tripathy, A., Bijayasingh, P., 2011. Yield Prediction Using Artificial Neural Networks, pp. 315–317. https://doi.org/10.1007/978-3-642-19542-6_57.

Bargoti, S., Underwood, J.P., 2017. Image segmentation for fruit detection and yield estimation in apple orchards. J. Field Rob. 34 (6), 1039–1060. https://doi.org/10.1002/rob.21699.

Beulah, R., 2019. A survey on different data mining techniques for crop yield prediction. Int. J. Comput. Sci. Eng. 7 (1), 738–744. https://doi.org/10.26438/ijcse/v7i1.738744.

Bhojani, S.H., Bhatt, N., 2020. Wheat crop yield prediction using new activation functions in neural network. Neural Comput. Appl. 1–11.

Bose, P., Kasabov, N., Bruzzone, L., n.d. Spiking neural networks for crop yield estimation based on spatiotemporal analysis of image time series. Ieeexplore.Ieee.Org. Retrieved from https://ieeexplore.ieee.org/abstract/document/7524771/.

Brownlee, J., 2016. Deep learning with Python: develop deep learning models on Theano and TensorFlow using Keras. Machine Learning Mastery.

Brownlee, J., 2017. Long Short-term Memory Networks with Python: Develop Sequence Prediction Models with Deep Learning. Machine Learning Mastery.

Brownlee, J., 2019. Deep Learning for Computer Vision: Image Classification, Object Detection, and Face Recognition in Python. Machine Learning Mastery.

Cakir, Y., Kirci, M., Gunes, E.O., 2014. Yield prediction of wheat in south-east region of Turkey by using artificial neural networks. In: 2014 The 3rd International Conference on Agro-Geoinformatics, Agro-Geoinformatics 2014. https://doi.org/10.1109/Agro-Geoinformatics.2014.6910609.

Charoen-Ung, P., Mittrapiyanuruk, P., 2019. Sugarcane yield grade prediction using random forest with forward feature selection and hyper-parameter tuning, pp. 33–42. https://doi.org/10.1007/978-3-319-93692-5_4.

Chen, Y., Lee, W.S., Gan, H., Peres, N., Fraisse, C., Zhang, Y., He, Y., 2019. Strawberry yield prediction based on a deep neural network using high-resolution aerial orthoimages. Remote Sens. 11 (13), 1584.

Cheng, H., Damerow, L., Sun, Y., Blanke, M., 2017. Early yield prediction using image analysis of apple fruit and tree canopy features with neural networks. J. Imag. 3 (1), 6. https://doi.org/10.3390/jimaging3010006.

Chlingaryan, A., Sukkarieh, S., Whelan, B., 2018. Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: a review. Comput. Electron. Agric. 151, 61–69. https://doi.org/10.1016/j.compag.2018.05.012.

Chu, Z., Yu, J., 2020. An end-to-end model for rice yield prediction using deep learning fusion. Comput. Electron. Agric. 174.

Crane-Droesch, A., 2018. Machine learning methods for crop yield prediction and climate change impact assessment in agriculture. Environ. Res. Lett. 13 (11), 114003. https://doi.org/10.1088/1748-9326/aae159.

Črtomir, R., Urška, C., Stanislav, T., Denis, S., Karmen, P., Pavlovič, M., Marjan, V., 2012. Application of neural networks and image visualization for early forecast of apple yield. Erwerbs-Obstbau 54 (2), 69–76. https://doi.org/10.1007/s10341-012-0162-y.

De Alwis, S., Zhang, Y., Na, M., Li, G., 2019. Duo attention with deep learning on tomato yield prediction and factor interpretation. In: Pacific Rim International Conference on Artificial Intelligence. Springer, Cham, pp. 704–715.

Elavarasan, D., Vincent, P.D., 2020. Crop yield prediction using deep reinforcement learning model for sustainable agrarian applications. IEEE Access 8, 86886–86901.

Elavarasan, D., Vincent, D.R., Sharma, V., Zomaya, A.Y., Srinivasan, K., 2018. Forecasting yield by integrating agrarian factors and machine learning models: a survey. Comput. Electron. Agric. 155, 257–282. https://doi.org/10.1016/j.compag.2018.10.024.

Everingham, Y., Sexton, J., Skocaj, D., Inman-Bamber, G., 2016. Accurate prediction of sugarcane yield using a random forest algorithm. Agron. Sustainable Dev. 36 (2). https://doi.org/10.1007/s13593-016-0364-z.

Everingham, Y.L., Smyth, C.W., Inman-Bamber, N.G., 2009. Ensemble data mining approaches to forecast regional sugarcane crop production. Agric. For. Meteorol. 149 (3–4), 689–696. https://doi.org/10.1016/J.AGRFORMET.2008.10.018.

Fathi, M.T., Ezziyyani, M., Ezziyyani, M., El Mamoune, S., 2019. Crop yield prediction using deep learning in Mediterranean Region. In: International Conference on Advanced Intelligent Systems for Sustainable Development. Springer, Cham, pp. 106–114.

Fernandes, J.L., Ebecken, N.F.F., Esquerdo, J.C.D.M., 2017. Sugarcane yield prediction in Brazil using NDVI time series and neural networks ensemble. Int. J. Remote Sens. 38 (16), 4631–4644. https://doi.org/10.1080/01431161.2017.1325531.

Filippi, P., Jones, E.J., Wimalathunge, N.S., Somarathna, P.D.S.N., Pozza, L.E., Ugbaje, S.U., Bishop, T.F.A., 2019a. An approach to forecast grain crop yield using multi-layered, multi-farm data sets and machine learning. Precis. Agric. 1–15. https://doi.org/10.1007/s11119-018-09628-4.

Filippi, P., Jones, E.J., Wimalathunge, N.S., Somarathna, P.D.S.N., Pozza, L.E., Ugbaje, S.U., Bishop, T.F.A., 2019b. An approach to forecast grain crop yield using multi-layered, multi-farm data sets and machine learning. Precis. Agric. https://doi.org/10.1007/s11119-018-09628-4.

Gandhi, N., Armstrong, L., 2016. Applying data mining techniques to predict yield of rice in humid subtropical climatic zone of India. In: Proceedings of the 10th INDIACom; 2016 3rd International Conference on Computing for Sustainable Global Development, INDIACom 2016, 1901–1906. Retrieved from https://ieeexplore.ieee.org/abstract/document/7724597/.

Gandhi, N., Armstrong, L.J., 2016b. A review of the application of data mining techniques for decision making in agriculture. In: Proceedings of the 2016 2nd International Conference on Contemporary Computing and Informatics, https://doi.org/10.1109/IC3I.2016.7917925.

Gandhi, N., Petkar, O., Armstrong, L.J., Tripathy, A.K., 2016. Rice crop yield prediction in India using support vector machines. In: 2016 13th International Joint Conference on Computer Science and Software Engineering, JCSSE 2016. https://doi.org/10.1109/JCSSE.2016.7748856.

Girish, L., Gangadhar, S., Bharath, T., Balaji, K., n.d. Crop Yield and Rainfall Prediction in Tumakuru District using Machine Learning. Ijream.Org. Retrieved from https://www.ijream.org/papers/NCTFRD2018015.pdf.

Goldstein, A., Fink, L., Meitin, A., Bohadana, S., Lutenberg, O., Ravid, G., 2018. Applying machine learning on sensor data for irrigation recommendations: revealing the agronomist's tacit knowledge. Precis. Agric. 19 (3), 421–444. https://doi.org/10.1007/s11119-017-9527-4.

Gonzalez-Sanchez, A., Frausto-Solis, J., Ojeda-Bustamante, W., 2014. Predictive ability of machine learning methods for massive crop yield prediction. Spanish J. Agric. Res. 12 (2), 313–328. https://doi.org/10.5424/sjar/2014122-4439.

Jeong, J.H., Resop, J.P., Mueller, N.D., Fleisher, D.H., Yun, K., Butler, E.E., Kim, S.H., 2016. Random forests for global and regional crop yield predictions. PLoS ONE 11 (6). https://doi.org/10.1371/journal.pone.0156571.

Ji, S., Xu, W., Yang, M., Yu, K., 2012. 3D convolutional neural networks for human action recognition. IEEE Trans. Pattern Anal. Mach. Intell. 35 (1), 221–231.

Jiang, H., Hu, H., Zhong, R., Xu, J., Xu, J., Huang, J., Lin, T., 2020. A deep learning approach to conflating heterogeneous geospatial data for corn yield estimation: a case study of the US Corn Belt at the county level. Glob. Change Biol. 26 (3), 1754–1766.

Johnson, M.D., 2013. Crop Yield Forecasting on the Canadian Prairies by Satellite Data and Machine Learning Methods. Master's Thesis, University of British Columbia, Atmospheric Science. Retrieved from https://www.sciencedirect.com/science/article/pii/S0168192315007546.

Ju, S., Lim, H., Heo, J., 2020. Machine learning approaches for crop yield prediction with MODIS and weather data. 40th Asian Conference on Remote Sensing: Progress of Remote Sensing Technology for Smart Future, ACRS 2019.

Kang, Y., Ozdogan, M., Zhu, X., Ye, Z., Hain, C.R., Anderson, M.C., 2020. Comparative assessment of environmental variables and machine learning algorithms for maize yield prediction in the US Midwest. Environ. Res. Lett.

Khaki, S., Wang, L., 2019. Crop yield prediction using deep neural networks. Front. Plant Sci. 10, 621.

Khaki, S., Wang, L., Archontoulis, S.V., 2020. A cnn-rnn framework for crop yield prediction. Front. Plant Sci. 10, 1750.

Khanal, S., Fulton, J., Klopfenstein, A., Douridas, N., Shearer, S., 2018. Integration of high resolution remotely sensed data and machine learning techniques for spatial prediction of soil properties and corn yield. Comput. Electron. Agric. 153, 213–225. https://doi.org/10.1016/J.COMPAG.2018.07.016.

Kitchenham, B., Charters, S., Budgen, D., Brereton, P., Turner, M., Linkman, S., Visaggio, G., 2007. Guidelines for performing Systematic Literature Reviews in Software Engineering. Retrieved from https://userpages.uni-koblenz.de/~laemmel/esecourse/slides/slr.pdf.

Kouadio, L., Deo, R.C., Byrareddy, V., Adamowski, J.F., Mushtaq, S., Phuong Nguyen, V., 2018. Artificial intelligence approach for the prediction of Robusta coffee yield using soil fertility properties. Comput. Electron. Agric. 155, 324–338. https://doi.org/10.1016/J.COMPAG.2018.10.014.

Kunapuli, S.S., Rueda-Ayala, V., Benavidez-Gutierrez, G., Cordova-Cruzatty, A., Cabrera,

A., Fernandez, C., Maiguashca, J., 2015. Yield prediction for precision territorial management in maize using spectral data. In: Precision Agriculture 2015 - Papers Presented at the 10th European Conference on Precision Agriculture, ECPA 2015 (pp. 199–206). Retrieved from https://www.scopus.com/inward/record.uri?eid=2-s2.0-84947244569&partnerID=40&md5=241e9b9de12f2eb0fae3ed0ee2fd22c0.

Lee, S., Jeong, Y., Son, S., Lee, B., 2019. A self-predictable crop yield platform (SCYP) based on crop diseases using deep learning. Sustainability 11 (13), 3637.

Li, B., Lecourt, J., Bishop, G., 2018. Advances in non-destructive early assessment of fruit ripeness towards defining optimal time of harvest and yield prediction—a review. Plants 7 (1). https://doi.org/10.3390/plants7010003.

Liakos, K.G., Busato, P., Moshou, D., Pearson, S., Bochtis, D., 2018. Machine learning in agriculture: a review. Sensors (Switzerland) 18 (8). https://doi.org/10.3390/s18082674.

Maimaitijiang, M., Sagan, V., Sidike, P., Hartling, S., Esposito, F., Fritschi, F.B., 2020. Soybean yield prediction from UAV using multimodal data fusion and deep learning. Remote Sens. Environ. 237.

Matsumura, K., Gaitan, C.F., Sugimoto, K., Cannon, A.J., Hsieh, W.W., 2015. Maize yield forecasting by linear regression and artificial neural networks in Jilin, China. J. Agric. Sci. 153 (3), 399–410. https://doi.org/10.1017/S0021859614000392.

Mayuri, P.K., Priya, V.C., n.d. Role of image processing and machine learning techniques in disease recognition, diagnosis and yield prediction of crops: a review. Int. J. Adv. Res. Comput. Sci., 9(2). https://doi.org/10.26483/ijarcs.v9i2.5793.

McQueen, R.J., Garner, S.R., Nevill-Manning, C.G., Witten, I.H., 1995. Applying machine learning to agricultural data. Comput. Electron. Agric. 12 (4), 275–293. https://doi.org/10.1016/0168-1699(95)98601-9.

Measuring Vegetation (NDVI & EVI), 2000. Retrieved from https://earthobservatory.nasa.gov/features/MeasuringVegetation/measuring_vegetation_2.php.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Petersen, S., 2015. Human-level control through deep reinforcement learning. Nature 518 (7540), 529–533.

Mola-Yudego, B., Rahlf, J., Astrup, R., Dimitriou, I., 2016. Spatial yield estimates of fast-growing willow plantations for energy based on climatic variables in northern Europe. GCB Bioenergy 8 (6), 1093–1105. https://doi.org/10.1111/gcbb.12332.

Monga, T., 2018. Estimating vineyard grape yield from images, pp. 339–343. https://doi.org/10.1007/978-3-319-89656-4_37.

Nevavuori, P., Narra, N., Lipping, T., 2019. Crop yield prediction with deep convolutional neural networks. Comput. Electron. Agric. 163.

Nguyen, L.H., Zhu, J., Lin, Z., Du, H., Yang, Z., Guo, W., Jin, F., 2019. Spatial-temporal multi-task learning for within-field cotton yield prediction. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, Cham, pp. 343–354.

Osman, T., Psyche, S.S., Kamal, M.R., Tamanna, F., Haque, F., Rahman, R.M., 2017. Predicting early crop production by analysing prior environment factors, pp. 470–479. https://doi.org/10.1007/978-3-319-49073-1_51.

Pantazi, X.E., Moshou, D., Mouazen, A.M., Kuang, B., Alexandridis, T., 2014. Application of supervised self organising models for wheat yield prediction, pp. 556–565. https://doi.org/10.1007/978-3-662-44654-6_55.

Pantazi, X.E., Moshou, D., Alexandridis, T., Whetton, R.L., Mouazen, A.M., 2016. Wheat yield prediction using machine learning and advanced sensing techniques. Comput. Electron. Agric. 121, 57–65. https://doi.org/10.1016/j.compag.2015.11.018.

Paul, M., Vishwakarma, S.K., Verma, A., 2015. Analysis of soil behaviour and prediction of crop yield using data mining approach. In: 2015 International Conference on Computational Intelligence and Communication Networks (CICN). IEEE, pp. 766–771. https://doi.org/10.1109/CICN.2015.156.

Rahman, M., Haq, N., n.d. Machine learning facilitated rice prediction in Bangladesh. Ieeexplore.Ieee.Org. Retrieved from https://ieeexplore.ieee.org/abstract/document/7113655/.

Rahnemoonfar, M., Sheppard, C., 2017. Real-time yield estimation based on deep learning. In: Autonomous Air and Ground Sensing Systems for Agricultural Optimization and Phenotyping II Vol. 10218. International Society for Optics and Photonics, pp. 1021809.

Ranjan, A.K., Parida, B.R., 2019. Paddy acreage mapping and yield prediction using sentinel-based optical and SAR data in Sahibganj district, Jharkhand (India). Spatial Inf. Res. https://doi.org/10.1007/s41324-019-00246-4.

Rao, T., Manasa, S., n.d. Artificial Neural networks for soil quality and crop yield prediction using machine learning. Ijfrcsce.Org. Retrieved from http://www.ijfrcsce.org/download/browse/Volume_5/January_19_Volume_5_Issue_1/1547885118_19-01-2019.pdf.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems, pp. 91–99.

Romero, J.R., Roncallo, P.F., Akkiraju, P.C., Ponzoni, I., Echenique, V.C., Carballido, J.A., 2013. Using classification algorithms for predicting durum wheat yield in the province of Buenos Aires. Comput. Electron. Agric. 96, 173–179. https://doi.org/10.1016/j.compag.2013.05.006.

Ruder, S., 2017. An overview of multi-task learning in deep neural networks. arXiv preprint arXiv:1706.05098.

Ruß, G., Kruse, R., 2010. Regression models for spatial data: an example from precision agriculture, pp. 450–463. https://doi.org/10.1007/978-3-642-14400-4_35.

Ruß, G., Kruse, R., Schneider, M., Wagner, P., 2008. Data mining with neural networks for wheat yield prediction. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 5077 LNAI. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 47–56. https://doi.org/10.1007/978-3-540-70720-2_4.

Saravi, B., Nejadhashemi, A.P., Tang, B., 2019. Quantitative model of irrigation effect on maize yield by deep neural network. Neural Comput. Appl. 1–14.

Schwalbert, R.A., Amado, T., Corassa, G., Pott, L.P., Prasad, P.V., Ciampitti, I.A., 2020. Satellite-based soybean yield forecast: Integrating machine learning and weather data for improving crop yield prediction in southern Brazil. Agric. For. Meteorol. 284.

Shah, A., Dubey, A., Hemnani, V., Gala, D., Kalbande, D.R., 2018. Smart Farming System: Crop Yield Prediction Using Regression Techniques. Springer, Singapore, pp. 49–56. https://doi.org/10.1007/978-981-10-8339-6_6.

Shekoofa, A., Emam, Y., Shekoufa, N., Ebrahimi, M., Ebrahimie, E., 2014. Determining the most important physiological and agronomic traits contributing to maize grain yield through machine learning algorithms: a new avenue in intelligent agriculture. PLoS ONE 9 (5), e97288. https://doi.org/10.1371/journal.pone.0097288.

Shidnal, S., Latte, M.V., Kapoor, A., 2019. Crop yield prediction: two-tiered machine learning model approach. Int. J. Inf. Technol. 1–9.

Šmite, D., Wohlin, C., Gorschek, T., Feldt, R., 2010. Empirical evidence in global software engineering: a systematic review. Empirical Softw. Eng. 15 (1), 91–118. https://doi.org/10.1007/s10664-009-9123-y.

Somvanshi, P., Mishra, B.N., 2015. Machine learning techniques in plant biology. In: PlantOmics: The Omics of Plant Science. Springer India, New Delhi, pp. 731–754. https://doi.org/10.1007/978-81-322-2172-2_26.

Su, Y.X., Xu, H., Yan, L.J., 2017. Support vector machine-based open crop model (SBOCM): case of rice production in China. Saudi J. Biol. Sci. 24 (3), 537–547.

Sujatha, R., Isakki, P., 2016. A study on crop yield forecasting using classification techniques. In: 2016 International Conference on Computing Technologies and Intelligent Data Engineering, ICCTIDE 2016. https://doi.org/10.1109/ICCTIDE.2016.7725357.

Sun, J., Di, L., Sun, Z., Shen, Y., Lai, Z., 2019. County-level soybean yield prediction using deep CNN-LSTM model. Sensors 19 (20), 4363.

Taherei-Ghazvinei, P., Hassanpour-Darvishi, H., Mosavi, A., Yusof, K.W., Alizamir, M., Shamshirband, S., Chau, K., 2018. Sugarcane growth prediction based on meteorological parameters using extreme learning machine and artificial neural network. Eng. Appl. Comput. Fluid Mech. 12 (1), 738–749. https://doi.org/10.1080/19942060.2018.1526119.

Tedesco-Oliveira, D., da Silva, R.P., Maldonado Jr, W., Zerbato, C., 2020. Convolutional neural networks in predicting cotton yield from images of commercial fields. Comput. Electron. Agric. 171.

Terliksiz, A.S., Altýlar, D.T., 2019. Use Of deep neural networks for crop yield prediction: a case study Of Soybean Yield in Lauderdale County, Alabama, USA. In: 2019 8th International Conference on Agro-Geoinformatics (Agro-Geoinformatics). IEEE, pp. 1–4.

Villanueva, M.B., Louella, M., Salenga, M., 2018. Bitter Melon Crop Yield Prediction using Machine Learning Algorithm. IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 9. Retrieved from www.ijacsa.thesai.org.

Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A., 2008. Extracting and composing robust features with denoising autoencoders. In: Proceedings of the 25th international conference on Machine learning, pp. 1096–1103.

Wang, A., Tran, C., Desai, N., Lobell, D., n.d. Deep transfer learning for crop yield prediction with remote sensing data. Dl.Acm.Org. Retrieved from https://dl.acm.org/citation.cfm?id=3212707.

Wang, X., Huang, J., Feng, Q., Yin, D., 2020. Winter wheat yield prediction at county level and uncertainty analysis in main wheat-producing regions of china with deep learning approaches. Remote Sens. 12 (11), 1744.

Wang, A.X., Tran, C., Desai, N., Lobell, D., Ermon, S., 2018. Deep transfer learning for crop yield prediction with remote sensing data. In: Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies, pp. 1–5.

Wang, Y., Zhang, Z., Feng, L., Du, Q., Runge, T., 2020. Combining multi-source data and machine learning approaches to predict winter wheat yield in the conterminous United States. Remote Sens. 12 (8), 1232.

Witten, I.H., Frank, E., Hall, M.A., Pal, C.J., 2016. Data Mining: Practical Machine Learning Tools and Techniques. Data Mining: Practical Machine Learning Tools and Techniques. https://doi.org/10.1016/c2009-0-19715-5.

Wolanin, A., Mateo-García, G., Camps-Valls, G., Gómez-Chova, L., Meroni, M., Duveiller, G., Guanter, L., 2020. Estimating and understanding crop yields with explainable deep learning in the Indian Wheat Belt. Environ. Res. Lett. 15 (2).

Xu, X., Gao, P., Zhu, X., Guo, W., Ding, J., Li, C., Wu, X., 2019. Design of an integrated climatic assessment indicator (ICAI) for wheat production: a case study in Jiangsu Province, China. Ecol. Ind. 101, 943–953. https://doi.org/10.1016/j.ecolind.2019.01.059.

Yalcin, H., 2019. An approximation for a relative crop yield estimate from field images using deep learning. In: 2019 8th International Conference on Agro-Geoinformatics (Agro-Geoinformatics). IEEE, pp. 1–6.

Yang, Q., Shi, L., Han, J., Zha, Y., Zhu, P., 2019. Deep convolutional neural networks for rice grain yield estimation at the ripening stage using UAV-based remotely sensed images. Field Crops Res. 235, 142–153.

Ying-xue, S., Huan, X., Li-jiao, Y., 2017. Support vector machine-based open crop model (SBOCM): Case of rice production in China. Saudi J. Biol. Sci. 24 (3), 537–547. https://doi.org/10.1016/j.sjbs.2017.01.024.

You, J., Li, X., Low, M., Lobell, D., Ermon, S., 2017. Deep Gaussian process for crop yield prediction based on remote sensing data. In: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17), 4559–4566. https://doi.org/10.1109/MWSCAS.2006.381794.

Zhang, Y., Yang, Q., 2017. A survey on multi-task learning. arXiv preprint arXiv:1707.08114.

Zhang, L., Zhang, Z., Luo, Y., Cao, J., Tao, F., 2020. Combining optical, fluorescence, thermal satellite, and environmental data to predict county-level maize yield in china using machine learning approaches. Remote Sens. 12 (1), 21.

Zhong, H., Li, Xiaocheng, Lobell, D., Ermon, S., Brandeau, M.L., 2018. Hierarchical modeling of seed variety yields and decision making for future planting plans. Environ. Syst. Decis. 38, 458–470.

Full Name	FirstName	LastName	Nationality	Gender	Dob
Yash Pawar	Yash	Pawar	Indian	Male	2001-Apr-06
Himanshu Sonawane	Himanshu	Sonawane	Indian	Male	2001-Feb-05
Chinmay Ambekar	Chinmay	Ambekar	Indian	Male	2001-Feb-13
Sonal Sonwane	Sonal	Sonwane	Indian	Female	2002-Apr-25
Imon Mandal	Imon	Mandal	Indian	Male	2000-Nov-20
Poonam Londhe	Poonam	Londhe	Indian	Female	2001-Mar-25
Kaustubh Adhe	Kaustubh	Adhe	Indian	Male	2001-Jun-12
Sankit Kotwal	Sankit	Kotwal	Indian	Male	2001-Oct-05
Aovi Anjankar	Aovi	Anjankar	Indian	Female	2001-Aug-22
Shubham More	Shubham	More	Indian	Male	2001-Dec-26
Ritik Jain	Ritik	Jain	Indian	Male	2001-Feb-27
Sana Chaudhari	Sana	Chaudhari	Indian	Female	2001-Sep-24
Vaishnavi Bhujbal	Vaishnavi	Bhujbal	Indian	Female	2002-Jan-18
Lokesh Kasliwal	Lokesh	Kasliwal	Indian	Male	2001-May-22
Dhruv Jain	Dhruv	Jain	Indian	Male	2001-Jul-04
Jaykumar Chandak	Jaykumar	Chandak	Indian	Male	2002-Jul-24
Harshal Jadhav	Harshal	Jadhav	Indian	Male	2000-Jul-24
Atharva Naphade	Atharva	Naphade	Indian	Male	2000-Aug-25
Yashodeep Shelar	Yashodeep	Shelar	Indian	Male	2001-Nov-17
Avantika Patil	Avantika	Patil	Indian	Female	2001-Aug-16
Sakshi More	Sakshi	More	Indian	Female	2001-Jun-13
Abhishek Lande	Abhishek	Lande	Indian	Male	2001-Nov-28
Vaishnavi Gadekar	Vaishnavi	Gadekar	Indian	Female	2001-Jan-08
Rutwik Saraf	Rutwik	Saraf	Indian	Male	2001-May-02
Diptesh Varule	Diptesh	Varule	Indian	Male	2000-Dec-06
Chaitanya Kardile	Chaitanya	Kardile	Indian	Male	2001-Nov-09
Janhavi Nirale	Janhavi	Nirale	Indian	Female	2001-Jul-05
Ashish Patil	Ashish	Patil	Indian	Male	2002-Jan-10
Manasi Hatekar	Manasi	Hatekar	American	Female	2001-Mar-22
Ankita Bharsakle	Ankita	Bharsakle	Indian	Female	2001-Oct-09
Pranav Bhagwat	Pranav	Bhagwat	Indian	Male	2001-Jul-19
Shivendu Shukre	Shivendu	Shukre	Indian	Male	2000-Nov-27
Aditya Shinde	Aditya	Shinde	Indian	Male	2002-Feb-27
Pranav Rathi	Pranav	Rathi	Indian	Male	2001-Jan-28
Hastak Patel	Hastak	Patel	Indian	Male	2000-Aug-27
Mahesh Paliwal	Mahesh	Paliwal	Indian	Male	2001-Aug-12
Shreyash Khandait	Shreyash	Khandait	Indian	Male	2001-Sep-01
Vedant Deshpande	Vedant	Deshpande	Indian	Male	2001-Mar-03

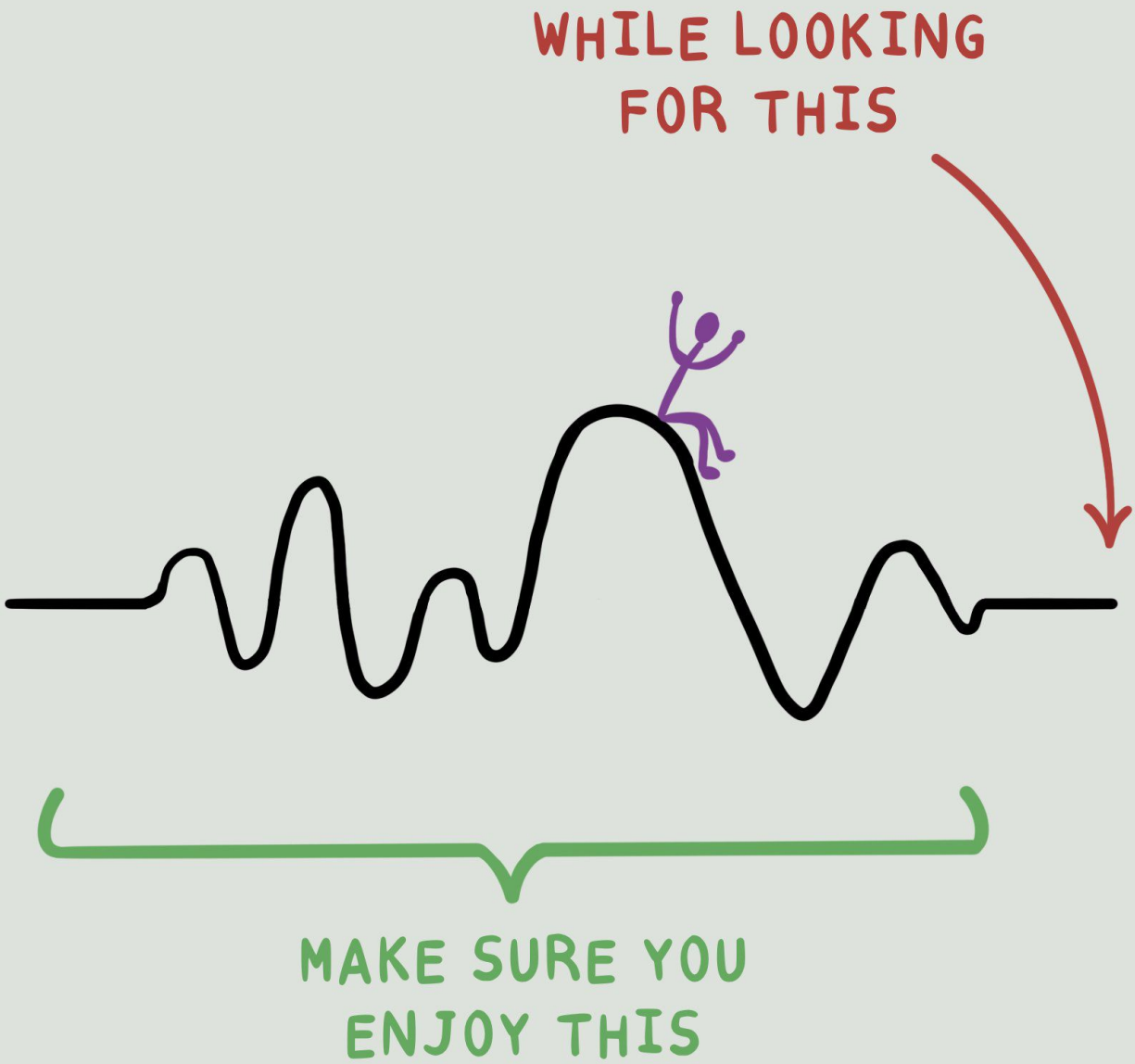
Mobile	EmailId1	Graduation	Graduation Education
91-9049578179	pawaryash1357@gmail.com	Graduation	B.E
91-8793840840	himanshusonawane1345@gmail.com	Graduation	B.E
91-8329870681	cuambekar@gmail.com	Graduation	B.E
91-7397914100	sonalsonwane2812@gmail.com	Graduation	B.E
91-8087565071	imonashokmandal@gmail.com	Graduation	B.E
91-7499910058	poonamlondhe25@gmail.com	Graduation	B.E
91-7021058521	kaustubhadhe@gmail.com	Graduation	B.E
91-9682669726	sankitkotwal0510@gmail.com	Graduation	B.E
91-7028718470	anjankaraovi@gmail.com	Graduation	B.E
91-7666723884	shubhammorewai513@gmail.com	Graduation	B.E
91-7049052727	ritikjain666@gmail.com	Graduation	B.E
91-9867040513	mdsanachaudhari99@gmail.com	Graduation	B.E
91-9307171933	bhujbalvaishu79@gmail.com	Graduation	B.E
91-9404530679	kasliwallokesh@gmail.com	Graduation	B.E
91-9922761114	dhruvjain6467@gmail.com	Graduation	B.Tech
91-7843019382	jaychandak111@gmail.com	Graduation	B.E
91-8975416783	harshaljadhav343@gmail.com	Graduation	B.E
91-9423754562	naphade21@gmail.com	Graduation	B.E
91-9022505679	yashodeepshelar2001@gmail.com	Graduation	B.E
91-8530634832	patilavantika168@gamil.com	Graduation	B.E
91-7620292256	sakshimore1013@gmail.com	Graduation	B.E
91-9146568041	abhisheklande028@gmail.com	Graduation	B.E
91-8605644490	vaishnavigadekar8888@gmail.com	Graduation	B.E
91-7498388349	rutwiksaraf3@gmail.com	Graduation	B.E
91-7769029148	dipteshvarule612@gmail.com	Graduation	B.E
91-8793248454	chaitanyakardile0911@gmail.com	Graduation	B.E
91-8793377096	niralejanhavi@gmail.com	Graduation	B.E
91-8805719908	patilashish2205@gmail.com	Graduation	B.E
91-9960502349	manasihatekar@gmail.com	Graduation	B.E
91-9552044281	bharsakleankita09@gmail.com	Graduation	B.E
91-8554018366	pranav221b@gmail.com	Graduation	B.E
91-7030063914	shivendu.shukre@gmail.com	Graduation	B.E
91-8879097208	adityashinde2722@gmal.com	Graduation	B.E
91-9145196969	impranavrathi@gmail.com	Graduation	B.E
91-9260646064	hastakpatel19@gmail.com	Graduation	B.E
91-9823596668	Paliwalmahesh981@gmail.com	Graduation	B.E
91-9307267342	shreyashkhandait@gmail.com	Graduation	B.E
91-8237827346	vedantd41@gmail.com	Graduation	B.E

Graduation Specialization	Graduation School/College/University
Computer Engineering	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Electronics & Telecommunication Engineering	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Electronics & Telecommunication Engineering	Pune Institute of Computer Technology
Electrical & Electronics Engineering	Pune Institute of Computer Technology
Electronics & Communication Engineering	Pune Institute of Computer Technology
Electronics & Telecommunication Engineering	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Computer Engineering	Pune Institute of Computer Technology
Computer Engineering	Pune Vidhyarthi Griha’s College of Arts & Science
Electronics & Telecommunication Engineering	Pune Institute of Computer Technology
Electronics & Telecommunication Engineering	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology
Information Technology	Pune Institute of Computer Technology

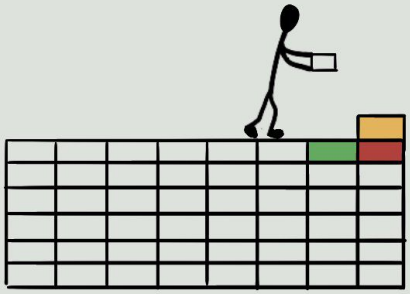
Graduation Year Of Passing	Graduation Percentage
2023	9.51
2023	8.43
2023	9.54
2023	9.5
2023	9.52
2023	9.25
2023	9.48
2023	81.56
2023	9.5
2023	9.49
2023	9.1
2023	9.3
2023	96
2023	94
2023	8.96
2023	9.36
2023	91
2023	8.86
2023	92
2023	8.95
2023	9.1
2023	96
2023	9.09
2023	92
2023	96
2023	88.88
2023	9.04
2023	91
2023	9.41
2023	86.7
2023	92
2023	9.1
2023	94
2023	8.92
2023	90.3
2023	91.8
2023	9.1
2023	98.5

TODAY

1 YEAR LATER


WHILE LOOKING
FOR THIS
MAKE SURE YOU
ENJOY THIS

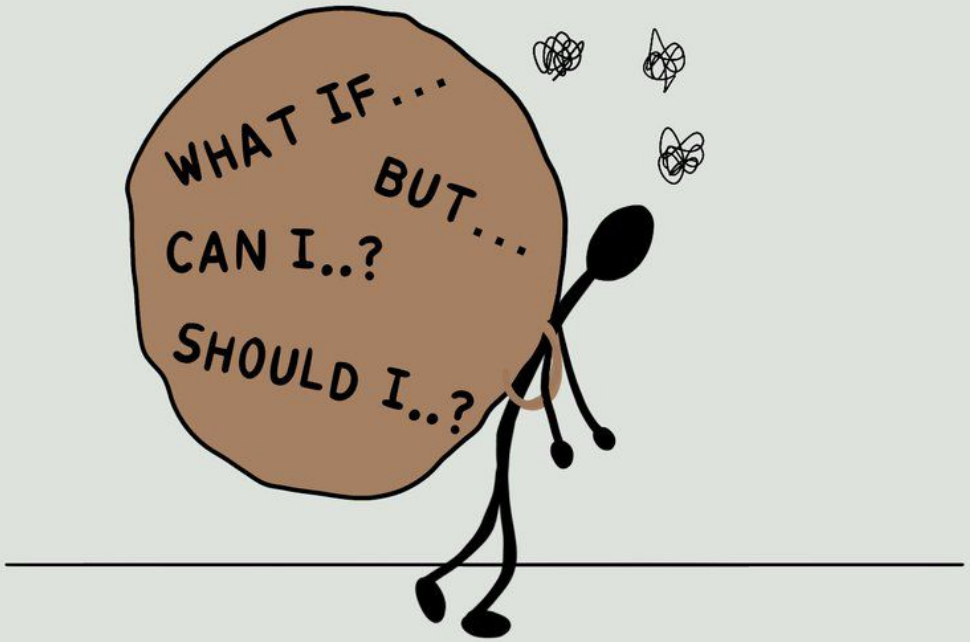
STEP
BY
STEP



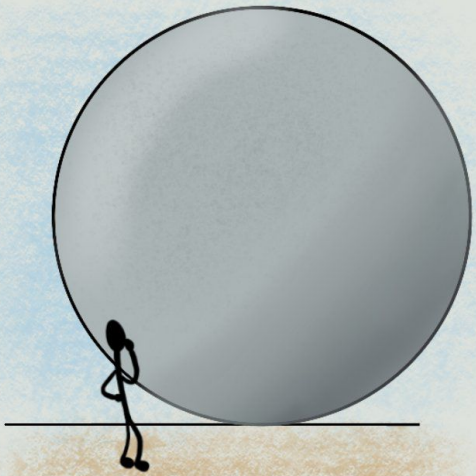
BRICK
BY
BRICK

DROP
BY
DROP

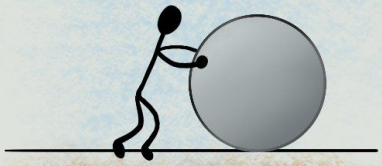

LET'S GO!
THIS IS USELESS
WOW.



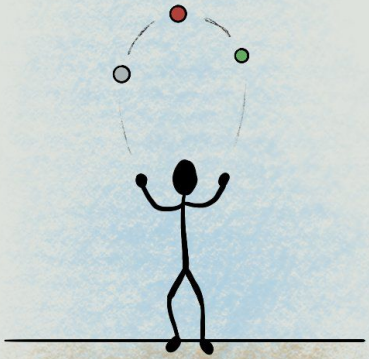
OVERTHINKING ENDS UP
BECOMING A HEAVY BAGGAGE



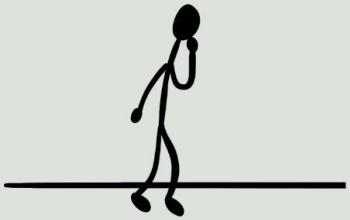
BEFORE YOU START



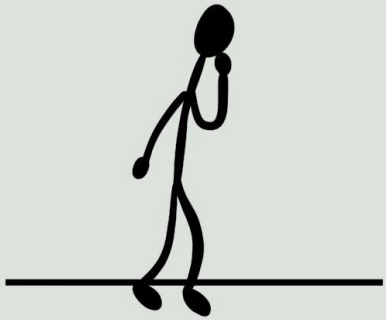
DURING THE TASK



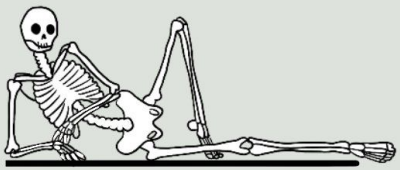
AFTER A WHILE



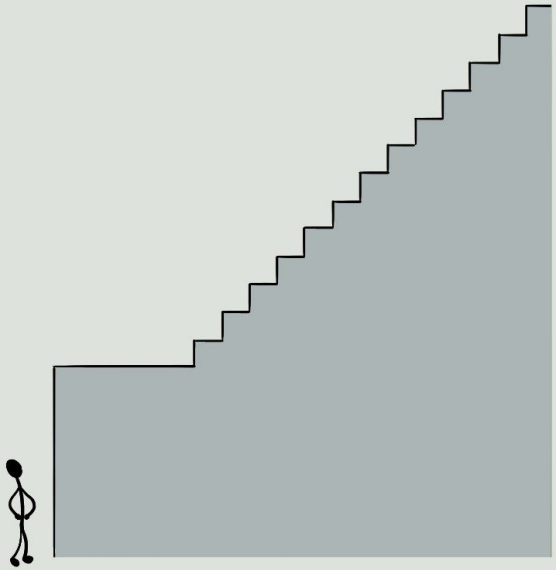
NOT READY YET...



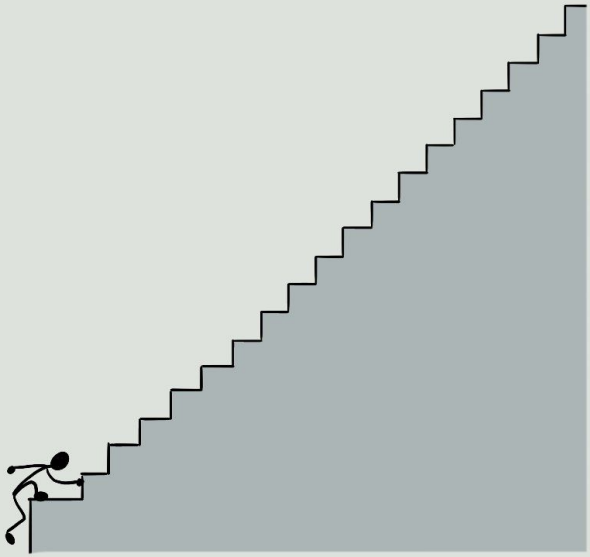
NOT READY YET...



NOT READY YET...

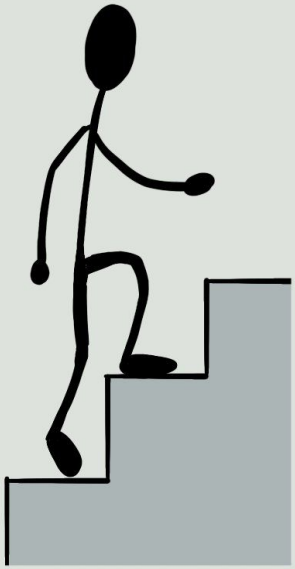


WHAT WE THINK
IT IS LIKE

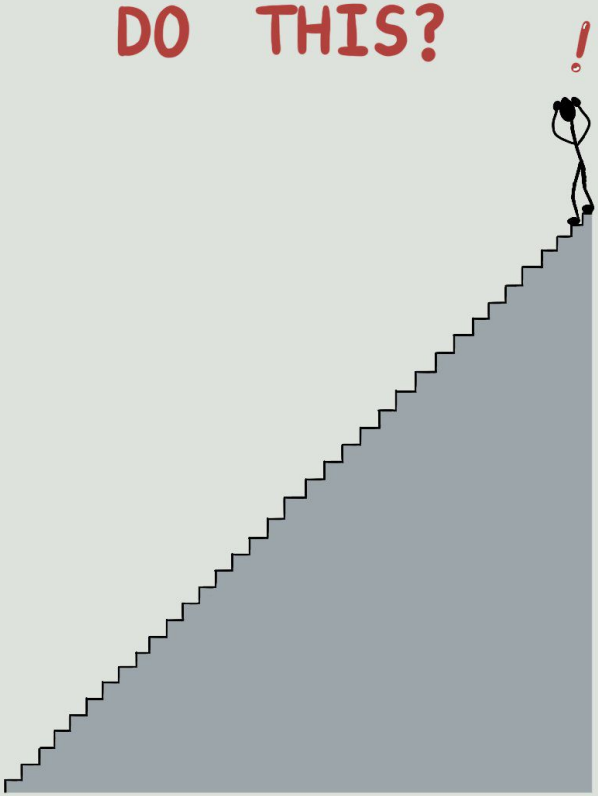


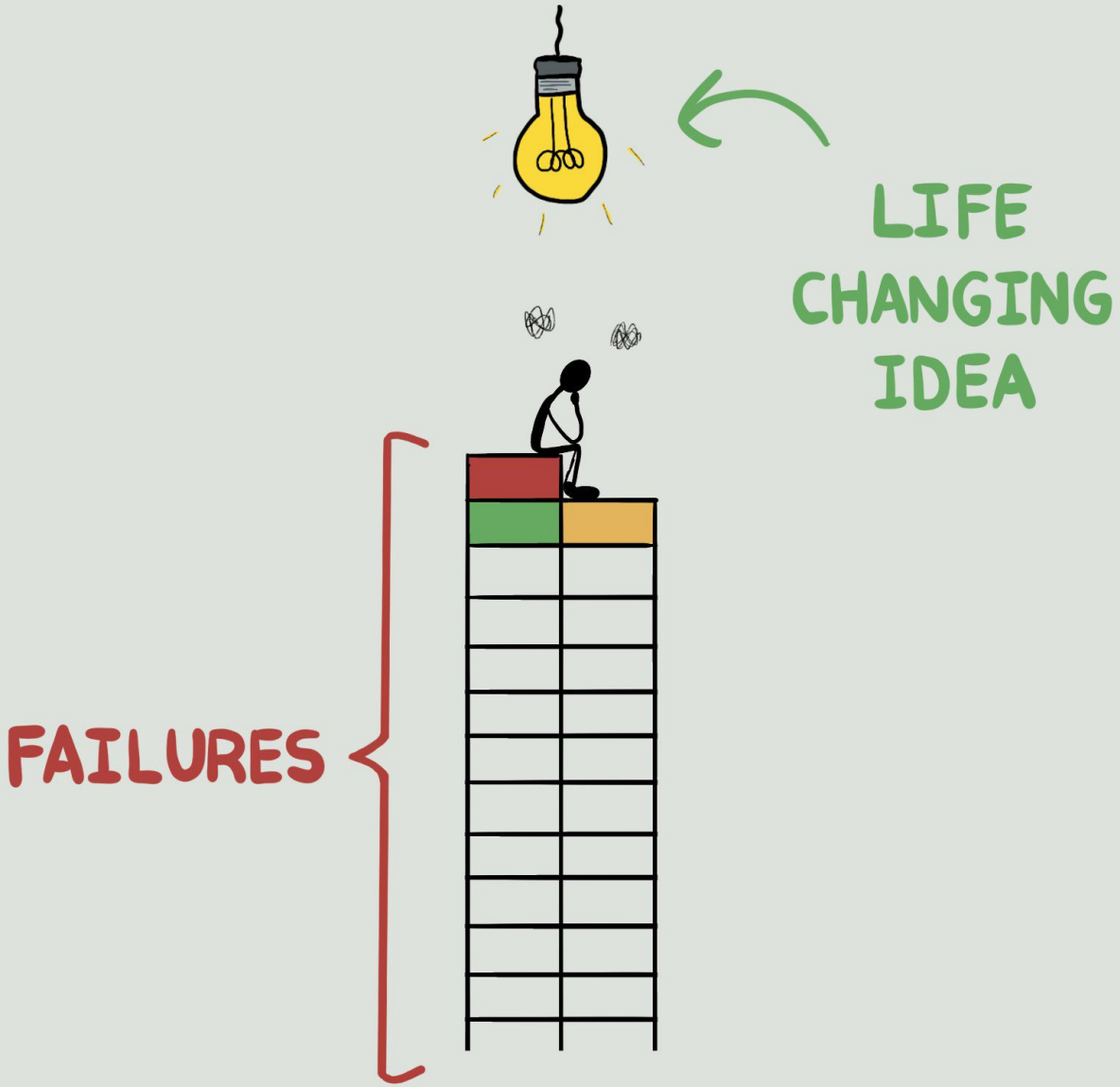
HOW IT
REALLY IS

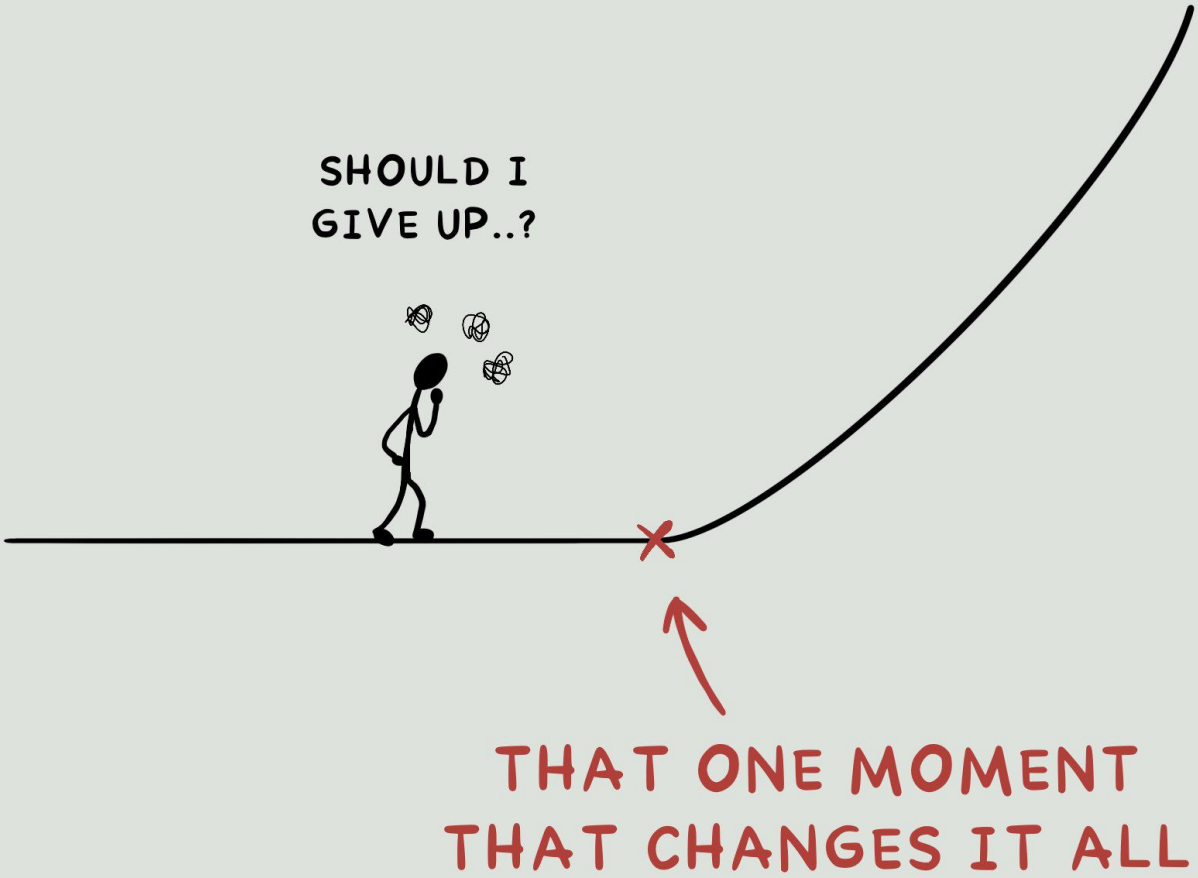
I CAN DO THIS ALL DAY

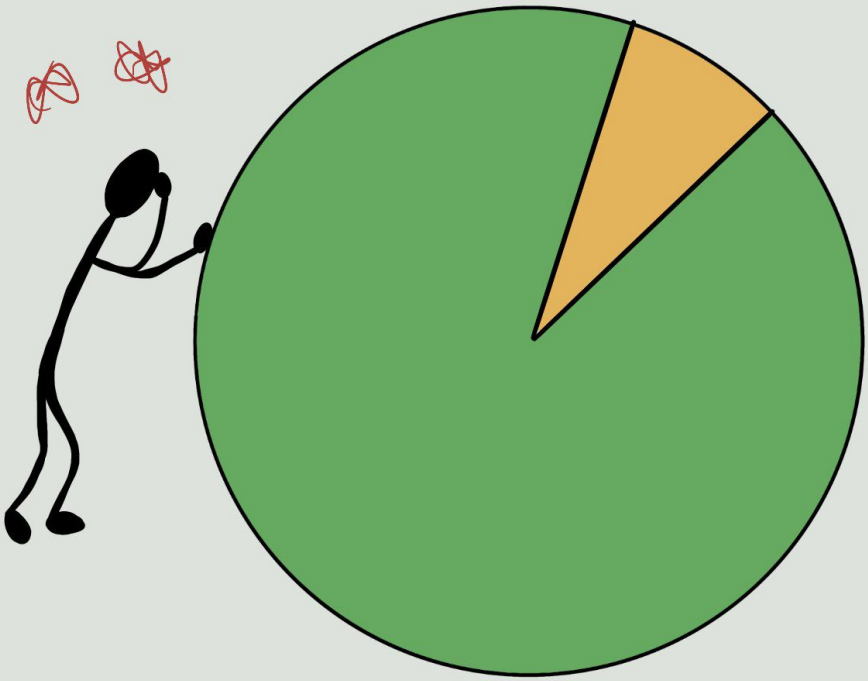


DID I REALLY DO THIS?



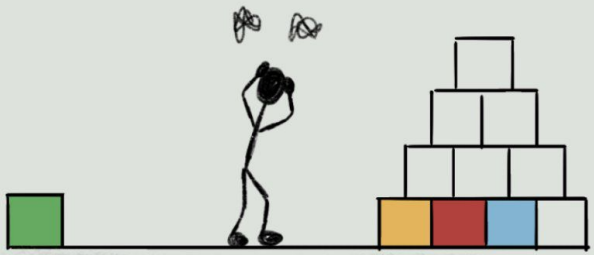

LIFE
CHANGING
IDEA
FAILURES


SHOULD I
GIVE UP..?
THAT ONE MOMENT
THAT CHANGES IT ALL

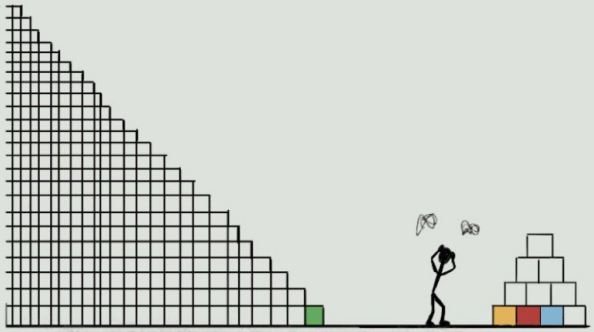


THINKING ABOUT DOING SOMETHING

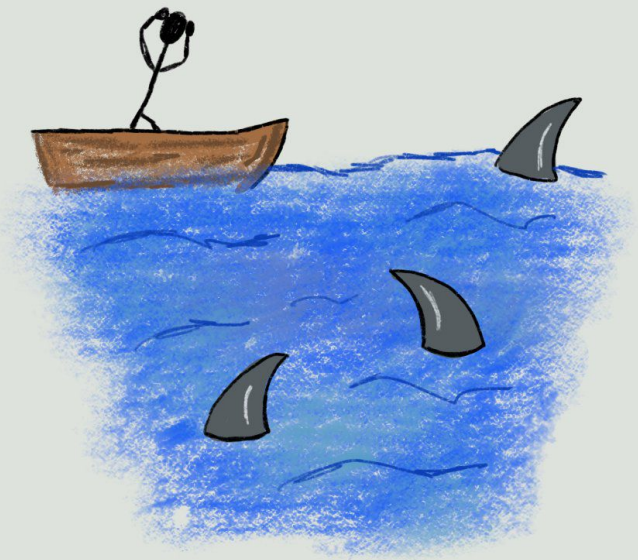
TIME IT REALLY TAKES DOING IT



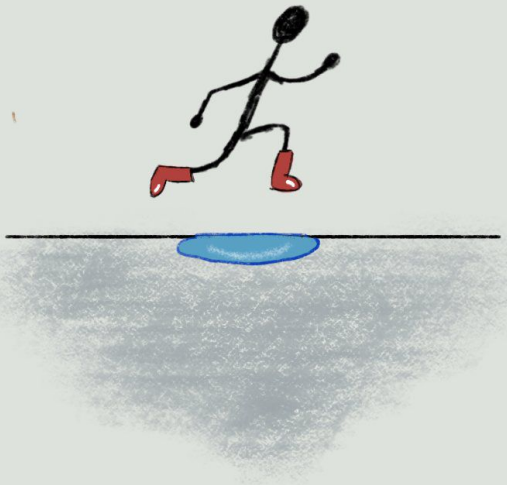
THINGS I HAVE TO DO



THINGS I HAVE SUCCESSFULLY DONE

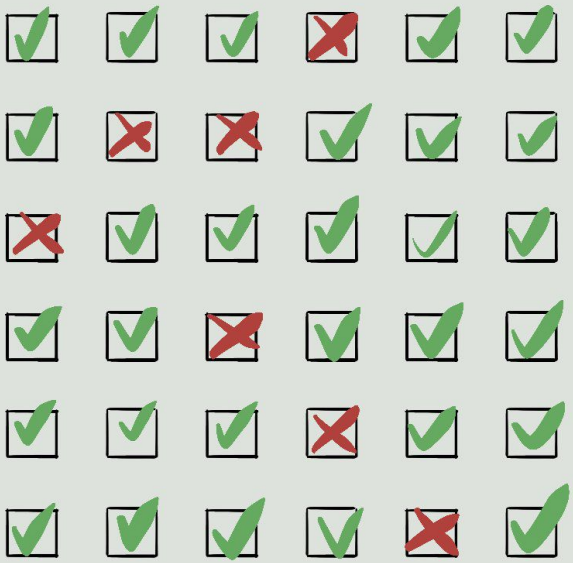


THE PROBLEM
I IMAGINE



THE PROBLEM
IN REALITY

FAILING
TWICE . . .



DOESNT MAKE
YOU A FAILURE

Driver Fatigue Detection Using Mouth and Yawning Analysis

Mandalapu Saradadevi[†] **Dr. Preeti Bajaj**[††]

B.D.College of Engineering, Sevagram, India G.H.Raisoni College of Engineering, India

Summary
Driver fatigue is an important factor in large number of accidents. There has been much work done in driver fatigue detection. This paper presents driver fatigue detection based on tracking the mouth and to study on monitoring and recognizing yawning. The authors proposed a method to locate and track driver's mouth using cascade of classifiers proposed by Viola-Jones for faces. SVM is used to train the mouth and yawning images. During the fatigue detection mouth is detected from face images using cascade of classifiers. Then, SVM is used to classify the mouth and to detect yawning then alert Fatigue.

Key words:
Driver fatigue, Fatigue detection, Driver monitoring system, Yawning, Support Vector Machine (SVM).

1. Introduction

The increasing number of traffic accidents due to a diminished driver's vigilance level has become a serious problem for society. Statistics show that 20% of all the traffic accidents are due to drivers with a diminished vigilance level [1]. Furthermore, accidents related to driver hypo-vigilance are more serious than other types of accidents, since sleepy drivers often do not take correct action prior to a collision. For this reason, developing systems for monitoring driver's level of vigilance and alerting the driver, when he is drowsy and not paying adequate attention to the road, is essential to prevent accidents. The prevention of such accidents is a major focus of effort in the field of active safety research.

In the last decade many researchers have been working on the development of the driver monitoring systems using different techniques. Driver's state of vigilance can be characterized by driver performance with a focus on the vehicle behavior or based on driver's physiological / physical state. The best accurate detection techniques are based on physiological phenomena of drivers like brain waves, heart rate, pulse rate and respiration [2]. These techniques are intrusive, since they need to attach some electrodes on the drivers body and causing annoyance to them. So this paper is focused on non-intrusive methods of fatigue detection mainly on measures of the driver's state.

People in fatigue show some visual behaviors easily observable from changes in their facial features like eyes, head, mouth and face [5]. Computer vision can be a natural and non-intrusive technique to monitor driver's vigilance. Faces as the primary part of human communication have been a research target in computer vision for a long time. The driver fatigue detection is considered as one of the most prospective commercial applications of automatic facial expression recognition [3]. Automatic recognition (or analysis) of facial expression consists of three levels of tasks: face detection, facial expression information extraction, and expression classification. In these tasks, the information extraction is the main issue for the feature based facial expression recognition from an image sequence [4]. It involves detection, identification and tracking facial feature points under different illuminations, face orientations and facial expressions.

Some common assumptions in previous face related works were: frontal facial views, constant illumination, and the fixed lighting source. Unfortunately these assumptions are not realistic. In the application of real world facial expression understanding, one has to consider at least three issues: capturing the full features in a variety of lighting conditions and head motion, multiple and non rigid object tracking, and the self-occlusion of features.

The process of falling asleep at the wheel can be characterized by a gradual decline in alertness from a normal state due to monotonous driving conditions or other environmental factors; this diminished alertness leads to a state of fuzzy consciousness followed by the onset of fatigue [6]. The critical issue that a fatigue detection system must address is the question of how to accurately and early detect fatigue at the initial stage. Possible non intrusive techniques for detecting fatigue in drivers using computer vision are

- Methods based on eye and eyelid movements
- Methods based on head movement
- Methods based on mouth opening

The authors have chosen methods based on mouth opening and yawning. Many of the previous researches deal with yawning detection focuses their methods on geometric features of the mouth. [7]. There are some disadvantages in yawning detection using geometric features of the

Manuscript received June 5, 2008.
Manuscript revised June 20, 2008.

mouth. First, left and right mouth corners are obvious feature points, but the lip positions are difficult to detect precisely. At the same time, lips move more acutely, which makes the lip detection more difficult. Third, geometric features are liable to pose and have more difference for individual. Here a method is proposed to locate and track a driver's mouth using cascade of classifiers proposed by Viola-Jones for faces. SVM is used to train the mouth and yawning images. During the fatigue detection mouth is detected from face images using cascade of classifiers. Then, SVM is used to classify the mouth regions to detect yawning then alert Fatigue.

2. Related Work

Haisong Gu et al. [3] proposes a graph-based reliability propagation to tackle the occlusion problem and verify the tracking results and their experimental results show validity of their active approach to real-life facial tracking under variable lighting conditions, head orientations, and facial expressions.

Xiao Fan et al. [7] gives to locate and track a driver's mouth movement using a CCD camera to study on monitoring and recognizing a driver's yawning and their experiment results show that Gabor coefficients are more powerful than geometric features to detect yawning and the average recognition rate is 95% which has more than 20% improvement.

Paul Viola et al. [8] have described a face detection framework that is capable of processing images extremely rapidly while achieving high detection rates as a process for training an extremely simple and efficient classifier which can be used as a "supervised" focus of attention operator and they present a set of experiments in the domain of face detection.

J. Nesvadba et al. [9] provides the potential of the Cassandra Framework's modular approach – using SUs for individual services - in combination with face-related content analysis algorithms and their framework provides an easy-to-use prototyping environment enabling the real-time execution of efficient and heterogeneous face-related algorithms, such as omni-directional face detection, pose estimation and face tracking in a distributed environment.

Yoav Freund et al. [10] introduces the boosting algorithm AdaBoost, and explains the underlying theory of boosting, including an explanation of why boosting often does not suffer from overfitting as well as boosting's relationship to support-vector machines and also they describe some of the basic underlying theory of boosting, including an explanation of why it often tends not to overfit.

Christopher J.C. Burges [11] give numerous examples and proofs of most of the key theorems and they how Support Vector machines can have very large (even infinite) VC dimension by computing the VC dimension for homogeneous polynomial and Gaussian radial basis function kernels and describes linear Support Vector Machines (SVMs) for separable and non-separable data, working through a non-trivial example in detail.

Chih-Wei Hsu et al. [12] proposes a simple procedure, which usually gives reasonable results and also they do not intend to solve challenging or difficult problems and they briefly introduce SVM basics which are necessary for explaining our procedure.

Theodoros Evgeniou et al. [13] discussed well known as well as emerging learning techniques such as Regularization Networks and Support Vector Machines which can be justified in term of the same induction principle and they overview the main concepts of Statistical Learning Theory, a framework in which learning from examples can be studied in a principled way.

Boser, B. E. et al. [15] developed a training algorithm that automatically tunes the capacity of the classification function by maximizing the margin by training examples and class boundary, optionally after removing some atypical or meaningless examples from the training data.

Cortes, C. et al. [16] construct a new type of learning machine, the so-called support-vector Network and also they compare the performance of the support-vector network to various classical learning algorithms that all took part in a benchmark study of Optical Character Recognition.

Yoshihiro Takei et al. [19] discussed the method to estimate a driver's fatigue through steering motion. They applied the Chaos theory to explain the change of steering wheel motion.

3. Mouth Extraction Algorithm

For Mouth detection the authors have used the detection algorithm proposed by Paul Viola and Micheal.J.Jones [8] used for Face detection by using cascade of classifiers. The overall proposed system is depicted in Figure 1

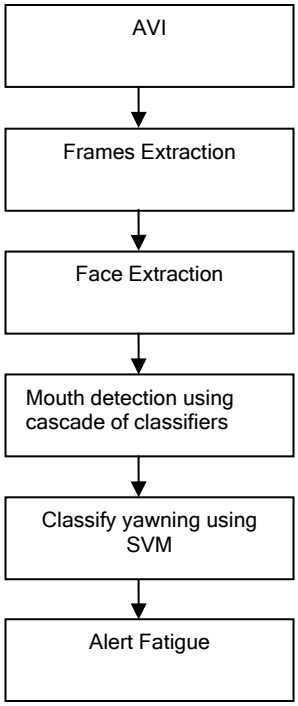


Fig. 1 Proposed System

3.1 Mouth Detection Algorithm

This mouth detection procedure classifies images based on the value of simple features. There are many motivations for using features rather than the pixels directly [8]. This image-based detection algorithm works on uncompressed images and has proven to be robust under various lighting conditions. The method is based on a cascade of boosted classifiers of simple Haar-wavelet like features on different scales and positions. The features are brightness and contrast-invariant and consist of two or more rectangular region pixel-sums that can be efficiently calculated by the canny integral image [9]. The feature set is over complete and an adaptation of the AdaBoost [10] learning algorithm is proposed to select and combine features into a linear classifier. To speed up detection a cascade of classifiers is used such that every classifier can reject an image. All classifiers are trained to reject part of the candidates such that on average only a low amount of features are used per position and scale. After all possible mouth candidates are obtained, a grouping algorithm reduces groups of mouth candidates into single positive detections.

A weak classifier thus consists of a feature (f), a threshold (θ) and a polarity (p) indicating the direction of the inequality:

$$h(x, f, p, \theta) = \begin{cases} 1 & if \ \ pf(x) < p\theta \\ 0 & otherwise \end{cases}$$

Here x is a 24 × 24 pixel sub-window of an image. Figure 2 displays the training of classifiers for mouth detection.

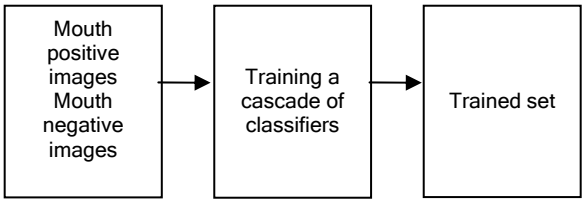
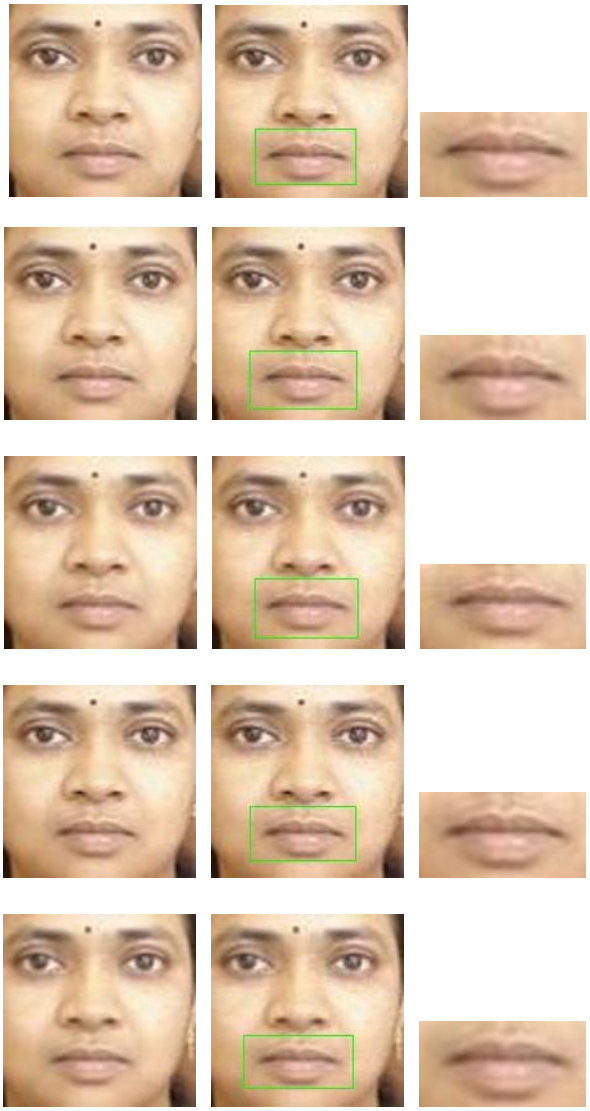


Fig. 2 Training of Classifiers

Figure 3 displays the list of Images used, Mouth Detection and extracted Mouths.



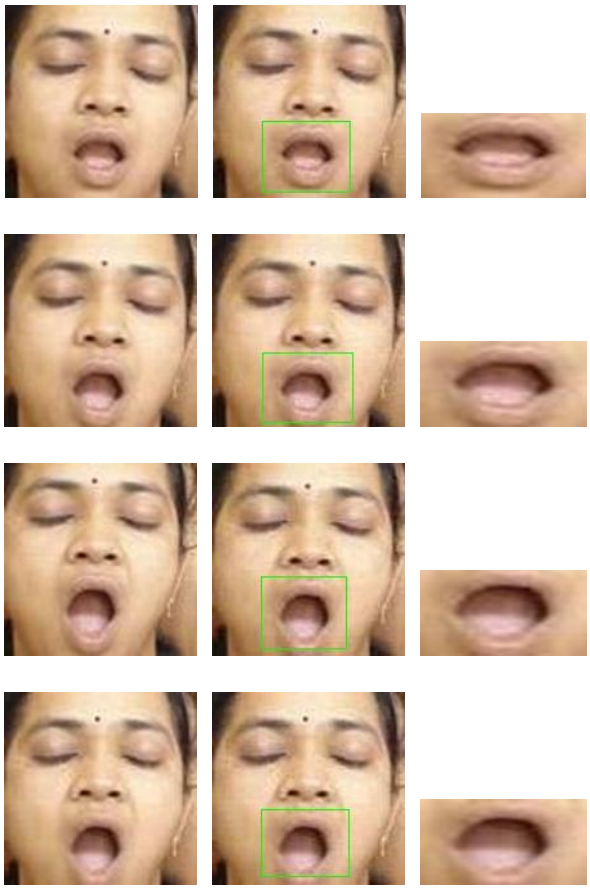


Fig. 3 Mouth Extraction from the Face Images

4. Fatigue Detection

Fatigue Detection has got two phases, one is training phase and the other one is detection phase which discussed below.

4.1 Support Vector Machine

SVM (Support Vector Machine) is a useful technique for data classification [11] - [14]. A classification task usually involves with training and testing data which consist of some data instances. Each instance in the training set contains one “target value" (class labels) and “several attributes" (features). The goal of SVM is to produce a model which predicts target value of data instances in the testing set which are given only the attributes. Figure 4 depicts the training of SVM and Figure 5 and Figure 6 lists the Training set of mouths used for SVM.

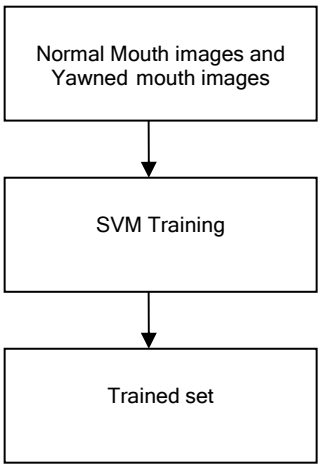


Fig. 4 SVM Training

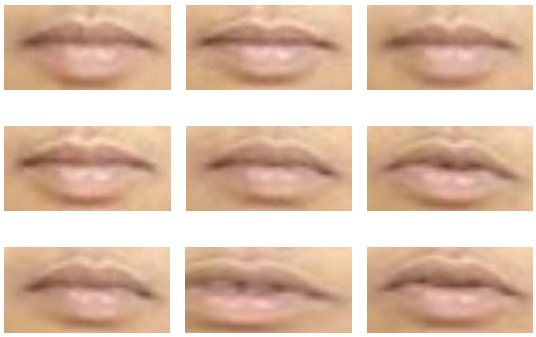


Fig. 5 Normal Mouths for SVM Training

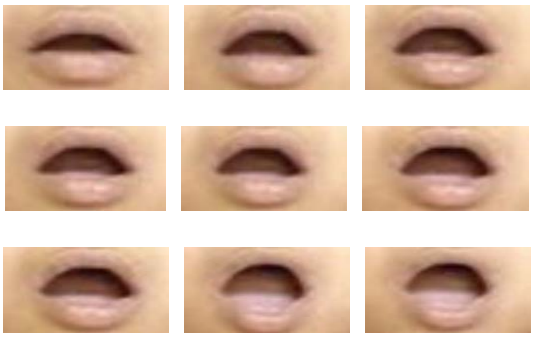


Fig. 6 Yawning Mouths for SVM Training

Given a training set of instance-label pairs (xi, yi), $i = 1,...,l$ where $xi \varepsilon R^n$ and $y \varepsilon \{1,-1\}^l$, the support vector machines (SVM) [15]-[16] require the solution of the following optimization problem:

$$\min_{w,b,\xi} \quad \frac{1}{2} w^T w + C \sum_{i-1}^{l} \xi_i$$

$$subject\ to \quad y_i(w^T\phi(x_i)+b) \geq 1-\xi_i,$$
$$\xi_i \geq 0.$$

Here training vectors x_i are mapped into a higher (maybe infinite) dimensional space by the function. Then SVM finds a linear separating hyperplane with the maximal margin in this higher dimensional space. C > 0 is the penalty parameter of the error term. Furthermore, $K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j)$ is called the kernel function.

Here we have used Radial basis function (RBF) kernel and given by

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0$$

Here γ is kernel parameter.

The following are the steps involved in the training of SVM.

Steps:

- Transform data to the format of an SVM software
- Conduct simple scaling on the data
- Consider the RBF kernel $K(x, y) = e^{-\gamma\|x-y\|^2}$
- Use cross-validation to find the best parameter C and γ
- Use the best parameter C and γ to train the whole training set.

SVM requires that each data instance is represented as a vector of real numbers. Hence, if there are categorical attributes, we first have to convert them into numeric data. Scaling them before applying SVM is very important. [17] explains why the data is scaled while using Neural Networks and most of considerations also apply to SVM. The main advantage is to avoid attributes in greater numeric ranges dominate those in smaller numeric ranges. The RBF kernel nonlinearly maps samples into a higher dimensional space, so it, unlike the linear kernel, can handle the case when the relation between class labels and attributes is nonlinear. The second reason is the number of hyperparameters which influences the complexity of model selection. The polynomial kernel has more hyperparameters than the RBF kernel. Finally, the RBF kernel has less numerical difficulties. One key point is $0 < K_{ij} \leq 1$ in contrast to polynomial kernels of which kernel values may go to infinity $(\gamma x_i^T x_j + r > 1)$ or zero $(\gamma x_i^T x_j + r < 1)$ while the degree is large. Moreover, we must note that the sigmoid kernel is not valid (i.e. not the inner product of two vectors) under some parameters [18].

There are several advantages of SVM's. The most important advantage is that during the training process, only a few vectors out of the training set are selected to become support vectors. This reduces the computational cost and provides a better generalization. Another advantage is that there are no local minima in the quadratic program, so the found solution is always the optimum of the given training set. Finally the main advantage is that the solution is not dependent on start conditions unlike neural networks.

5. Results

The authors collected few videos and selected about 20 yawning images and more than 1000 normal images as the data set. From each video, 10 yawning images and 10 normal images are given to cascade of classifiers for training. Same samples of Video are trained by SVM to classify normal and yawing mouths. Then the proposed approach is tested and the results of classification are presented in table 1.

Table1: Results of classification of Normal and yawning mouths

State	Normal	Yawning	Correct Rate
Normal	260	40	86%
Yawning	7	30	81%

6. Conclusion

In this paper the authors have proposed a method to locate and track a driver's mouth using cascade of classifiers training and mouth detection. They also trained the mouth and yawning images using SVM. Finally, SVM is used to classify the mouth regions to detect yawning then alert Fatigue. The experimental results show that proposed method gives better results than methods using geometric features. The proposed method detects yawning, alert fatigue earlier, and will facilitate to make drive safer. In future, the authors will capture more video clips to train and test the proposed method. The main goal is to develop a system to combine more features including mouth features, eye features and head tracking to monitor driver fatigue.

References

[1] L.M. Bergasa, J. Nuevo, M.A. Sotalo, and M. Vazquez, "Real-time system for monitoring driver Vigilance," in Proc. Intelligent Vehicle Symp., Parma, Italy, pp.78-83, 2004.

[2] J. Healey, and R. Picard, "Smart Car: detecting driver stress," in Proc.15th Int.Conf. Pattern Recognition, Barcelona, Spain, vol. 4, pp. 218-221, 2000.

[3] H. Gu, Q. Ji, and Z.W. Zhu, "Active facial tracking for fatigue detection," in Proc.Workshop on Application of Computer Vision, Orlando, pp. 137-142, 2002.

[4] Quong Wang, Jingyu Yang, Mingwu Ren, Yujie Zheng, "Driver Fatigue Detection: A Survey", in Proc Intelligent

control and Automation, Dalion, China, pp 8587- 8591, 2006.
[5] "AWAKE - System for Effective Assessment of Driver Vigilance and Warning According to Traffic Risk Estimation" from http://www.awake-eu.org/back.html
[6] Ueno, H.. Kaneda, M.. Tsukino, M. "Development of drowsiness detection system" IEEE, Vehicle Navigation and Information Systems Conference, 1994. Proceedings. 1994 Volume, Issue, 31 Aug-2 Sep 1994.
[7] Xiao Fan Bao-Cai Yin Yan-Feng Sun "Yawning Detection for Monitoring Driver Fatigue" IEEE Machine Learning and Cybernetics, 2007 International Conference on 19-22 Aug. 2007.
[8] Viola, P. and Jones, M.J., Robust real-time face detection. International Journal of Computer Vision. v57 i2. 137-154.
[9] J. Nesvadba, A. Hanjalic, P. Fonseca, B. Kroon, H. Celik, E. Hendriks, 'Towards a real-time and distributed system for face detection, pose estimation and face-related features', Invited Paper, Proc. Int. Con. on Methods and Techniques in Behavioral Research, Wageningen, The Netherlands, 2005.
[10] Y. Freud and R. Schapire, A short introduction to Boosting, J. of Japanese Society for AI, 14(5): 71~780, 1999
[11]Christopher J. C. Burges. A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery, 2(2):121{167, 1998.
[12] Chih-Wei Hsu, Chih-Chung Chang, and Chih-Jen Lin. A practical guide to support vector classification. Technical report, Department of Computer Science and Information Engineering National Taiwan University Taipei 106, Taiwan, 2003.
[13] Theodoros Evgeniou, Massimiliano Pontil, and Tomaso Poggio. Statistical learning theory: A primer. International Journal of Computer Vision, 38(1):9-13, 2000.
[14]T. Hastie, R. Tibshirani, and J. Friedman. The Elements of Statistical Learning. Springer, 2001. HAS t 01:1 1.Ex.
[15] Boser, B. E., I. Guyon, and V. Vapnik (1992). A training algorithm for optimal margin classifiers. In Proceedings of the Fifth Annual Workshop on Computational Learning Theory, pp. 144{152. ACM Press.
[16] Cortes, C. and V. Vapnik (1995). Support-vector network. Machine Learning 20,273-297.
[17] Sarle, W. S. (1997). Neural Network FAQ. Periodic posting to the Usenet newsgroup comp.ai.neural-nets.
[18] Vapnik, V. (1995). The Nature of Statistical Learning Theory. New York, NY: Springer-Verlag.
[19] Takei, Y. Furukawa, Y.: Estimate of driver's fatigue through steering motion. In: Man and Cybernetics, 2005 IEEE International Conference. (Volume: 2, On page(s): 1765-1770 Vol. 2)

Mandalapu Saradadevi received B.Tech degree in Electronics & Communication Engineering from V.R.Siddhardha Engineering college, Vijayawada and M.E in Electronics & Communication Engineering from Andhra university ,Visakhapatnam. She is currently pursuing PhD from RTM Nagpur University, Nagpur, India. She has over 3 publications in refereed International Conferences and Journals. She has been with B.D.College of Engineering, Sevagram since 1994. She also served in the faculty of K.L.College of Engineering, Vijayawada. She is member of Board of studies in RTM Nagpur university.

Dr. Preeti Bajaj holds a PhD degree in Electronics Engineering. Currently she is working as Principal, G.H.Raisoni College of Engineering, Nagpur, India. Her research interests include various domains like Soft Computing and optimizations, Hybrid Systems & Applications of Fuzzy logic in areas of Intelligent Transportation Systems such as Driver Monitoring/ Fatigue Systems. Her professional societies affiliation include Member- Institute of Engineers, Senior Member- IEEE, LM-ISTE, ILM-UK, LM-CSI. She has chaired many technical sessions at International Conferences in India and abroad. She is also a recipient of Best Teacher Award in the year 2001-02. She worked as reviewer for many International Conferences and Journals in India and abroad. Currently she is also working as General Chair for First International Conference on Emerging Trends in Engineering & Technology(ICETET-08) at GHRCE Nagpur, India. She has over 30 publications in refereed International Conferences and Journals. She has offered consultancy to National Highways Authority of India for designing and proposing various Intelligent Transportation projects on National Highways.

Design of Real-time Drowsiness Detection System using Dlib

[1]Shruti Mohanty, [1]Shruti V Hegde, [1]Supriya Prasad, [2]J. Manikandan
[1]Dept of ECE, PES University, 100-Feet Ring Road, BSK Stage III, Bengaluru - 85, Karnataka, India
[2]Dept of ECE, Crucible of Research and Innovation (CORI)
PES University, 100-Feet Ring Road, BSK Stage III, Bengaluru - 85, Karnataka, India
{email : shrutimohanty998@gmail.com, shrutihegde98@gmail.com, supriya.sprasad@gmail.com, manikandanj@pes.edu}

***Abstract*—Drowsiness while driving is a highly prevalent problem that leads to thousands of fatal accidents every year. A solution to prevent accidents and fatalities is the need of the hour and while there are complex systems developed that provide solutions for detecting drowsiness in drivers, this paper explores a simpler, yet highly effectual method of doing the same. In this paper, drowsy driver detection system is designed using Python and Dlib model. Dlib's shape detector is used to map the coordinates of the facial landmarks of the input video and drowsiness detected by monitoring aspect ratios of eyes and mouth. Performance evaluation of the proposed system designed is carried out by testing videos from a standard public dataset as well as real-time video captured in our lab. The proposed system gave a maximum recognition accuracy of 96.71% for dataset video input.**

Keywords*— *Python, face detection, drowsiness detection, computer vision, Dlib, OpenCV, HOG, facial landmark estimation

I. INTRODUCTION

Facial expressions have the ability to offer deep insights into many physiological conditions of the body. The display of a myriad of emotions and reactions to stimuli have been a constant area of study and research, and also used in the development of intelligent systems such as facial emotion detection in [1], automatic pain detection in [2] and prediction of personality in [3] to name a few. There are numerous algorithms and methodologies available for face detection which is the fundamental first step in the process.

There has been extensive research and a number of papers that have put forth possible methodologies to detect inattentiveness and drowsiness in a driver in the last two decades. In [4], traditional techniques are elaborated which are based on physiological measurements including brain waves, heart rate, pulse rate and respiration. However, these techniques are intrusive in nature. Reference [5] is based on Rowley's eye detection code from the STASM library. However, the presence of glasses adversely affects the performance of the system. Reference [6] monitors only yawning patterns of the driver using two separate cameras to acquire information of the upper part of the body in order to track the driver's mouth. However, the hardware dependency is higher.

Drowsiness in humans is characterized by a few very specific movements and facial expressions- the eyes begin to close, mouth opens in a yawn, the jaw goes slack and the neck tilts. This paper focuses on tracking the eyes and mouth to detect drowsiness and classify a driver as drowsy. For real-time application of the model, the input video can be acquired by mounting a camera on the dashboard of the car and can accommodate the driver's face, hands, upper body and occlusions such as non-tinted spectacles.

The Dlib model is trained to identify 68 facial landmarks. As shown in Fig 1. the drowsiness features are extracted and the driver is alerted incase of drowsiness being detected. The model does not require prior information on the individual who is testing it. Main software requirements are Python and OpenCV.

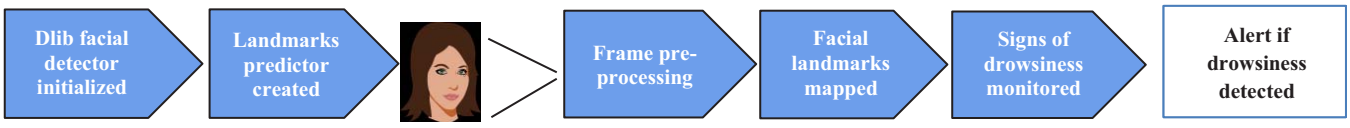


Fig. 1: Block diagram of proposed real-time drowsiness detection system

978-1-7281-4499-3/19/$31.0 ©2019 IEEE

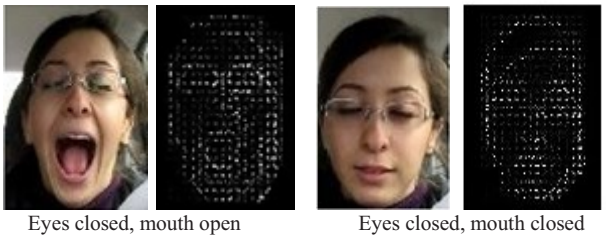
II. IMPLEMENTATION

For this approach, we implement the drowsiness detector using OpenCV and Python. The Dlib library is used to detect and localize facial landmarks using Dlib's pre-trained facial landmark detector. It consists of two shape predictor models [7] trained on the i-Bug 300-W dataset, that each localize 68 and 5 landmark points respectively within a face image. In this approach, 68 facial landmarks have been used (as shown in Fig. 2 below).

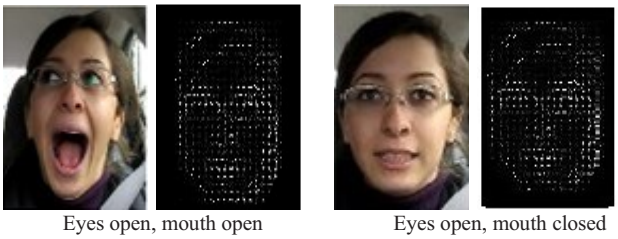
Fig. 2: Manner in which 68 facial landmarks are mapped on a detected face [8]

Histogram of Oriented Gradients (HOG) based face detector is used in Dlib. In this method, frequencies of gradient direction of an image in localized regions are used to form histograms. In many cases, it is more accurate than Haar cascades as the false positive ratio is small. Also, tuning at test time requires less parameters. It is especially suitable for face detection as firstly, it can describe contour and edge features exceptionally in various objects. Secondly, it performs operations on regional cells which allows motion of the subject to be overlooked. Moreover, Dalal and Triggs [9] discovered that HOG descriptor works well for human detection in images, which makes it appropriate for drowsiness detection.

In our model, a HOG based detector is first instantiated to find the location of the face in each individual frame of the input video stream.

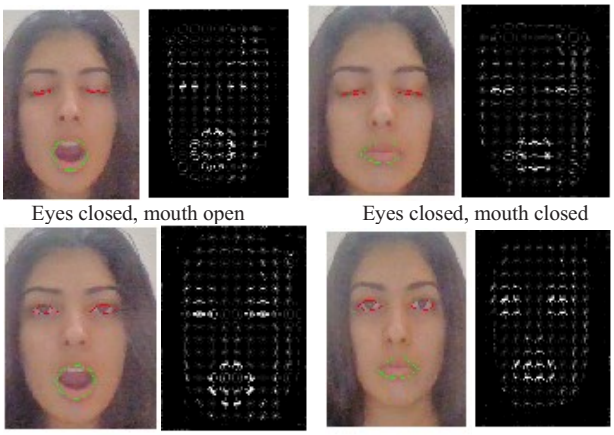


Eyes closed, mouth open | Eyes closed, mouth closed



Eyes open, mouth open | Eyes open, mouth closed

Fig. 3: HOG face features for four different dataset input cases



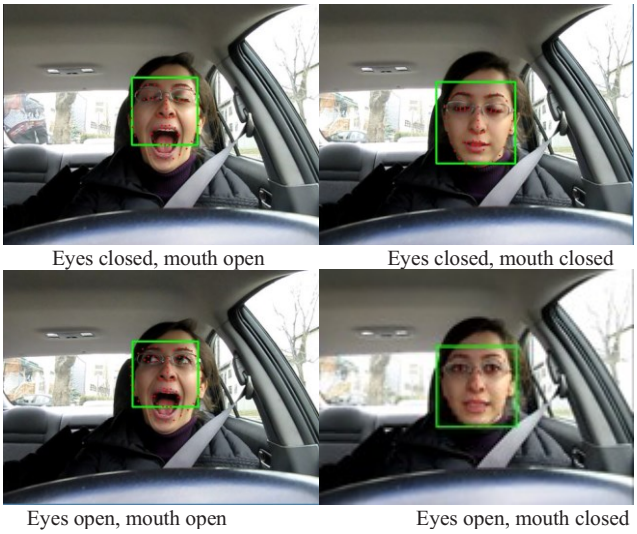
Eyes closed, mouth open | Eyes closed, mouth closed

Eyes open, mouth open | Eyes open, mouth closed

Fig. 4: HOG face features for four different real-time input cases

The outline of the facial features made by the oriented gradients makes it easy to discern the location and even the state of facial features. For example, in Fig. 3 and Fig. 4 we can see the difference in the HOG of an open mouth versus that of a closed mouth.

Upon finding the location of the face, the facial landmarks predictor is called to map the points of interest (eyes and mouth) and extract their coordinates.



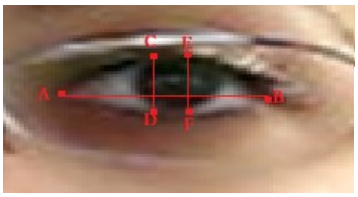
Eyes closed, mouth open | Eyes closed, mouth closed

Eyes open, mouth open | Eyes open, mouth closed

Fig. 5: Facial landmark mapping for four different dataset cases

Fig. 5 shows the facial landmark mapping on particular frames extracted from the video input for different cases such as eyes closed mouth open, eyes open mouth closed, eyes open mouth open and eyes closed mouth closed.

The coordinates of the right eye, left eye and mouth extracted at this stage are used to compute aspect ratio for the right eye, left eye and mouth based on Euclidean distance (as shown in Fig. 6 below).

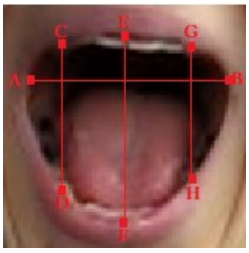


$$EAR = \frac{|CD| + |EF|}{2 * |AB|}$$

Fig. 6: Eye coordinates

From the formula in Fig. 6, the eye coordinates are obtained and eye aspect ratio (EAR) is calculated according to the formula. The aspect ratio of both the eyes is averaged as blinking is performed by both simultaneously.

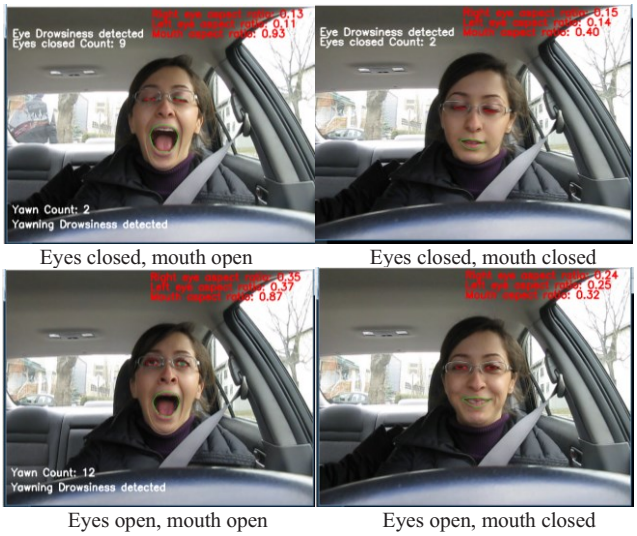
Similarly, mouth aspect ratio is determined to detect yawning from the coordinates of the mouth and the formula shown in Fig. 7 below.



$$MAR = \frac{|CD| + |EF| + |GH|}{3 * |AB|}$$

Fig. 7: Mouth coordinates

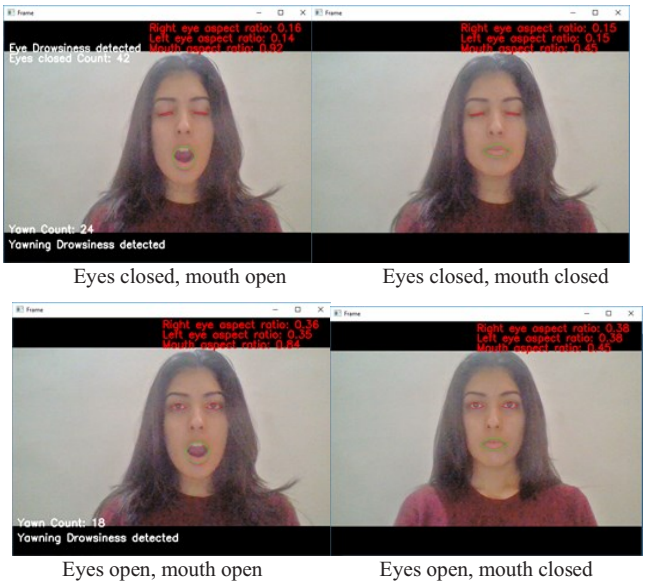
The final display of the drowsiness detection system shows the feed of the video input (from video dataset or real-time capture), along with the computed aspect ratio values and drowsiness detection alerts. If the aspect ratio of the eyes falls below the stipulated threshold, then the message "Eye Drowsiness detected" flashes on the screen along with the count of how many times the eye was noticed to be closed. If the aspect ratio of the mouth falls below the stipulated threshold, then the message "Yawning Drowsiness detected" is displayed on the screen along with the count of how many times yawning was detected.



Eyes closed, mouth open | Eyes closed, mouth closed

Eyes open, mouth open | Eyes open, mouth closed

Fig. 8: Drowsiness detection on a dataset video for 4 different cases (inset text: aspect ratios of both eyes and mouth and drowsiness count)



Eyes closed, mouth open | Eyes closed, mouth closed

Eyes open, mouth open | Eyes open, mouth closed

Fig. 9: Drowsiness detection on a real-time video for 4 different cases (inset text: aspect ratios of both eyes and mouth and drowsiness count)

The following steps are followed for the testing of the model:

Step 1: Input video (pre-recorded or real-time) is fed into the model. Individual frames are resized and converted to grayscale.

Step 2: Dlib's HOG based face detector is initialised. The location of the face is pinpointed.

Step 3: The facial landmarks for the face region are determined by the predictor and mapped onto the face.

Step 4: Left eye, right eye and mouth coordinates are extracted, which are then used to compute aspect ratio for both eyes and mouth based on Euclidean distance respectively.

Step 5: The calculated aspect ratios are compared with fixed threshold values 0.15 and 0.83 for eye and mouth respectively to determine signs of drowsiness. If the average aspect ratio of left and right eye falls below the threshold, it is recognized as a sign of drowsiness. Similarly, if the mouth aspect ratio exceeds the set threshold, there is a possibility for it to be a yawn.

Step 6: When continuous signs of drowsiness is detected over a longer duration, the driver is alerted.

The real time-video is processed at 20 frames per second (fps), so each frame lasts for 0.05 seconds. Drowsy blinks typically last for 20 frames i.e., 1 second. Thus, a normal blink will not be identified as drowsy. Continuous eye blinks also last for lesser number of frames and are hence distinguishable from drowsy blinks.

III. Experimental Results

A. Dataset Description

Table I provides a description of the 2 datasets used for testing purposes.

TABLE I: DATASET DESCRIPTION

Feature	YawDD Dataset[10]	MRL Eye Dataset[11]
Number of videos/images	29 Videos	84,898 Images
Number of males	16	33
Number of females	13	4
Actions performed	Without talking or talking/singing or yawning	Closed or open eyes
Resolution	640x480 RGB (24-bit true colour)	640x480, 1280x1024, 752x480

B. Results Obtained from Comparison

Table II below describes the recognition accuracy obtained from two approaches (eye closure and yawn detection), on using standard datasets and real-time.

Real-time computational results were calculated by taking the average of 5 trials each of 12 subjects (including 5 males and 7 females) recorded at different locations. Average result included cases with and without glasses. Video frames with instances of 2 states (sleepy and non-sleepy) for every trial. Highest percentage accuracy obtained is 96.71 % for yawn detection followed by 93.25% for drowsy blink detection.

TABLE II: RESULTS OBTAINED

Features	Recognition Accuracy	
	Real Time	***Dataset***
Eye	82.02 %	93.25 %
Yawn	85.44 %	96.71 %

IV. Conclusions

In the Dlib approach, the library's pre-trained 68 facial landmark detector is used. The face detector which is based on Histogram of Oriented Gradients (HOG) was implemented. The quantitative metric used in the proposed algorithm was the Eye Aspect Ratio (EAR) to monitor the driver's blinking pattern and Mouth Aspect Ratio (MAR) to determine if the driver yawned in the frames of the continuous video stream. The average real-time test accuracies obtained using Dlib for eyes and yawn were 82.02% and 85.44% respectively and 93.25% and 96.71% respectively for pre-recorded videos.

The results of real-time detection are lower as the model currently works exceedingly well under good to perfect light conditions like those found in the dataset videos, whereas the real-time testing was performed under a variety of lighting conditions.

Future work will focus on enhancing the model to work under poor to mediocre lighting conditions, and including more drowsiness signs such as head nodding for the drowsiness detection model.

References

[1] M. Matsugu., K. Mori., Y. Mitari, and Y. Kaneda, "Subject independent facial expression recognition with robust face detection using a convolutional neural network," *Neural networks: the official journal of the Intern. Neural Network Society*, vol. 16, no. 5-6, pp. 555-559, June 2003.

[2] P. Lucey, J. Cohn, S. Lucey, I. Matthews, S. Sridharan and K. M. Prkachin, "Automatically detecting pain using facial actions," in *3rd Intern. Conf. on Affective Computing and Intelligent Interaction and Workshops*, Amsterdam, 2009, pp. 1-8.

[3] L. Liu, D. Preotiuc-Pietro., Z.R. Saman, M.E. Moghaddam, and L.H. Ungar, "Analyzing Personality through Social Media Profile Picture Choice," *Intern. AAAI Conf. on Web and Social Media*, Cologne, Germany, May 2016, pp. 214.

[4] P. K. Stanley, T. Jaya Prakash, S. Sibin Lal and P. V. Daniel, "Embedded based drowsiness detection using EEG signals," *IEEE Intern. Conf. on Power, Control, Signals and Instrumentation Engineering*, Chennai, India, Sept. 2017, pp. 2596-2600.

[5] T. Danisman, I. M. Bilasco, C. Djeraba and N. Ihaddadene, "Drowsy driver detection system using eye blink patterns," *Intern. Conf. on Machine and Web Intelligence*, Algiers, Algeria, Oct. 2010, pp. 230-233.

[6] L. Li, Y. Chen and Z. Li, "Yawning detection for monitoring driver fatigue based on two cameras," *12th Intern. IEEE Conf. on Intelligent Transportation Systems*, St. Louis, USA, Oct. 2009, pp. 1-6.

[7] Davis E. King, Dlib-models [Online]. Available: https://github.com/davisking/dlib-models

[8] L. Anzalone, *Training Alternative Dlib Shape Predictor models using Python,* Oct. 2018. Accessed on July 16, 2019. [Online]. Available: https://medium.com/datadriveninvestor/training-alternative-dlib-shape-predictor-models-using-python-d1d8f8bd9f5c

[9] N. Dalal, B. Triggs "Histogram of oriented gradients for human detection", *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, San Diego, USA, June 2005, pp. 63-69.

[10] S. Abtahi, M. Omidyeganeh, S. Shirmohammadi, and B. Hariri. 2014. *YawDD: a yawning detection dataset, 5th ACM Multimedia Systems Conf.*, New York, USA, Mar. 2014, pp. 24-28.

[11] R. Fusek, *MRL Eye Dataset*, Accessed on May 28th, 2019. [Online]. Available: http://mrl.cs.vsb.cz/eyedataset

Join the Team Making Possibilities Happen

Digital Payments are revolutionizing our way of transacting and doing business, and ACI Worldwide is a leader driving the transformation of Real-Time Digital payments. If you've ever used an ATM, paid a bill through your phone, sent money to a friend, or shopped online, chances are your transaction was safeguarded and processed using our software. Now it's your turn to serve the payment needs of organizations and people the world over.

As an Associate Software Engineer in Pune, you will join a diverse, passionate team, dedicated to making possibilities happen in the payments industry!

Job Summary:
Participates in planning, designing, developing and testing software systems or applications for software enhancements and new products

Job Responsibilities:
Participates in the design, development, testing and delivery of high-quality software/solutions.
Creates and reviews unit, integration, system, and/or QA Test Plans/Scripts, executes test, and analyzes results.
Participates in the researches and resolution of customer reported problems.
Perform other duties as assigned
Understand and adhere to all corporate policies to include but not limited to the ACI Code of Business Conduct and Ethics.
Working in an Agile environment with working CI/CD model.
Helps with requirements analysis, functional and technical design, application build, product configuration, unit testing, and production deployment.
Provides technical expertise and ownership in the diagnosis and resolution of an issue, including the determination and provision of workaround solution or escalation to service owners.

Knowledge, Skills and Experience required for the job:
Bachelor's degree in computer science or related field or equivalent experience.
Specific knowledge of software development technology and methodology.

Preferred Knowledge, Skills and Experience needed for the job:
Gradle, Maven, Jenkins, Git, Gerrit, Docker

Java EE: the servlet specification, Spring, Hibernate, Hystrix, Java Script, JQuery
Oracle and PGSQL SQL knowledge
.NET , C# , SQL Server , Selenium , Angular 8 and above
Cloud Technology

Sr. No.	Full Name	Email-ID
1	Yash Pawar	pawaryash1357@gmail.com
2	Aovi Anjankar	anjankaraovi@gmail.com
3	Sana Chaudhari	mdsanachaudhari99@gmail.com
4	Atharva Naphade	naphade21@gmail.com
5	Yashodeep Shelar	yashodeepshelar2001@gmail.com
6	Abhishek Lande	abhisheklande028@gmail.com
7	Vaishnavi Gadekar	vaishnavigadekar8888@gmail.com
8	Diptesh Varule	dipteshvarule612@gmail.com
9	Hastak Patel	hastakpatel19@gmail.com
10	Tanaya Chaudhari	tanayachaudhari08@gmail.com
11	Vedant Deshpande	vedantd41@gmail.com
12	Sonal Sonwane	sonalsonwane2812@gmail.com
13	Imon Mandal	imonashokmandal@gmail.com
14	Dhruv Jain	dhruvjain6467@gmail.com
15	Shivendu Shukre	shivendu.shukre@gmail.com

Name	Email-ID	Slot Alloted
Yash Pawar	pawaryash1357@gmail.com	9.30 AM
Himanshu Sonawane	himanshusonawane1345@gmail.com	
Chinmay Ambekar	cuambekar@gmail.com	
Poonam Londhe	poonamlondhe25@gmail.com	
Kaustubh Adhe	kaustubhadhe@gmail.com	
Sankit Kotwal	sankitkotwal0510@gmail.com	
Aovi Anjankar	anjankaraovi@gmail.com	
Shubham More	shubhammorewai513@gmail.com	
Sana Chaudhari	mdsanachaudhari99@gmail.com	
Vaishnavi Bhujbal	bhujbalvaishu79@gmail.com	

Lokesh Kasliwal	kasliwallokesh@gmail.com	11.45 AM
Jaykumar Chandak	jaychandak111@gmail.com	
Harshal Jadhav	harshaljadhav343@gmail.com	
Atharva Naphade	naphade21@gmail.com	
Yashodeep Shelar	yashodeepshelar2001@gmail.com	
Avantika Patil	patilavantika168@gamil.com	
Abhishek Lande	abhisheklande028@gmail.com	
Vaishnavi Gadekar	vaishnavigadekar8888@gmail.com	
Rutwik Saraf	rutwiksaraf3@gmail.com	
Diptesh Varule	dipteshvarule612@gmail.com	

Janhavi Nirale	niralejanhavi@gmail.com	1.30 AM
Ashish Patil	patilashish2205@gmail.com	
Manasi Hatekar	manasihatekar@gmail.com	
Pranav Bhagwat	pranav221b@gmail.com	
Aditya Shinde	adityashinde2722@gmal.com	
Hastak Patel	hastakpatel19@gmail.com	
Mahesh Paliwal	Paliwalmahesh981@gmail.com	
Shreyash Khandait	shreyashkhandait@gmail.com	
Tanaya Chaudhari	tanayachaudhari08@gmail.com	
Vedant Deshpande	vedantd41@gmail.com	

Sonal Sonwane	sonalsonwane2812@gmail.com	2:45 PM
Imon Mandal	imonashokmandal@gmail.com	
Ritik Jain	ritikjain666@gmail.com	
Dhruv Jain	dhruvjain6467@gmail.com	
Sakshi More	sakshimore1013@gmail.com	
Chaitanya Kardile	chaitanyakardile0911@gmail.com	
Ankita Bharsakle	bharsakleankita09@gmail.com	
Shivendu Shukre	shivendu.shukre@gmail.com	
Pranav Rathi	impranavrathi@gmail.com	

SHIVAM BUNGE

Aspiring SDE| Tech Enthusiast| Fullstack Developer

Email: shivambunge42@gmail.com
LinkedIn: linkedin.com/in/shivam-bunge-9526141a0
Portfolio: shivambunge.github.io/

EDUCATION

- **Pune Institute of Computer Technology (B.E.)**
  *B.E. Electronics and Telecommunications -**9.30 CGPA*** *July 2019- July 2023*
- **Army Public School Kirkee**
  *CBSE Class 12 - **94.8%*** *March 2018- March 2019*
- **Army Public School Kirkee**
  *CBSE Class 10 - **98.2%*** *March 2016- March 2017*

SKILLS

- **Programming Languages :** Java, Python, JavaScript
- **Coursework :** Data Structures & Algorithms, Computer Networks, DBMS, OOP
- **Web Technologies :** HTML, CSS, JavaScript, Bootstrap, React.js, Django
- **Databases :** MYSQL, Firebase
- **Others :** Git, Github, Problem Solving, Leadership, Management

WORK EXPERIENCE

- **Associate (Tech and Product)** *June 2022 -Present*
  Global Governance Initiative
  - Worked under the co-founder directly, managed technical aspects, content and operations.
  - Products **Schrodinger** and **Learning Management System.** https://www.schrodingerbyggi.org/
  - Enhanced user experience by building a Spotify like personalization feature and redesigning frontend.
  - Improved Dashboard (LMS) built using *MERN* stack.
  - Added new Business Case Studies using Frontend techs and JavaScript using Twine.

- **Team Lead** *Feb 2022- May 2022*
  Brahmastra
  - Devised the product roadmap and features for an Online Consulting App- **MentorNow.** https://mentornow.me/
  - Managed and led a team of 4 interns.
  - Used *React.js* for creating various components and pages in the frontend.
  - Applied UseEffect, UseState hooks of React for page re-rendering and rendering.
  - Integrated firebase authentication, performed CRUD operations on the data.

- **Web Development Intern**
  Vibrancy *Dec 2021- March 2022*
  - Built the entire frontend of the website of a social media application- **Vibrancy**
  - Hosted the website on AWS Amplify with AWS S3 as the database.
  - Techs used: HTML, CSS, JavaScript, Bootstrap. website: https://www.wevibrancy.com

PROJECTS

- **MentorNow** *May 2022*
  Built an Online Consulting Application where users can get consultation for any service from health, fitness, Legal, career counseling, etc. https://mentornow.me/
- **Finlit** (Hackathon Project-1st rank) *Nov 2021*
  Financial Literacy application for people between ages 14-25 years. Techs: HTML, CSS, JavaScript, Bootstrap, Django. https://finlit.pythonanywhere.com/
- **Parking Application** *Oct 2021*
  Built a parking application that shows the nearest available parking slots using heap and generates receipt on booking. https:/shivamb.pythonanywhere.com/
- **Minor Projects:** React To-do app, React Hooks Project, Django Blog Application, Sentiment Analysis of Stocks

ACHIEVEMENTS

- Ranked 1st in PICT vs VIIT Development Battle (Inter College Hackathon) *Nov 2021*
- Among top 2 percentile in MHCET *June 2019*
- Among top 15 candidates in class 10th in entire Mumbai Zone (CBSE) *May 2017*

COMPETITIVE PROFILES

- **Github:** 650+ contributions on various projects. github.com/ShivamBunge
- **Leetcode**: https:/leetcode.com/ShivamBunge/
- **Hackerrank:** https://www.hackerrank.com/etc_32263_Shivam

ORGANIZATIONS

- **PICT, IEEE Branch** (Backend Web Developer) *Jan 2021 – Present*
  Contributed to college fest web applications built in Django.

IMON MANDAL

+91 8087565071
imonashokmandal@gmail.com
Pune, Maharashtra, India
https://www.linkedin.com/in/imon-mandal-a5a5a61a7/
https://github.com/imonmandal

EDUCATION

BACHELOR OF ENGINEERING

BE in Computer Engineering
Pune Institute of Computer Technology, Pune.
August 2019 - March 2023
College Affiliated to Savitribai Phule Pune University
Aggregate CGPA - 9.52 (till 5th semester)

HIGHER SECONDARY SCHOOL

Amrita Vidyalayam Nigdi Pune - 2016-2018
HSC result: 86.31%

SECONDARY SCHOOL

Jai Hind High School Pimpri Pune - 2007-2016
SSC result: 93.60 %

TECHNICAL SKILLS

- **C++**
- **Python**
- **MySQL**
- **PHP**
- **DSA / Problem Solving**

Also, have a basic knowledge of Java, Mongo DB, HTML, CSS, and JS.

LANGUAGES

- English
- Hindi
- Marathi
- Bengali

PROFILE

I am currently pursuing B.E. in Computer Engineering and willing to enhance my knowledge, skills, and experience by getting involved into challenging work environment. I am an enthusiastic, self-motivated, reliable, responsible and hard working person. I am a team worker and adaptable to all challenging situations.

PROJECTS

TIME TABLE MANAGEMENT SYSTEM

- This project enables teachers to make time tables for classes, teachers, and rooms by avoiding clashes.
- The teacher has to enter data only in the class timetable and the timetable for the room and the teacher will be taken care of.
- The Backend was done using PHP and the Frontend was done using HTML, CSS and JS.

WEATHER APPLICATION

- A user can see today's weather and forecast for the next 5 days.
- The Backend was done by using NodeJS and Frontend was done by using HTML, CSS and JS.

EXPERIENCE

WEB DEVELOPMENT INTERN

AT PICT COLLEGE FEB 2022 - APRIL 2022

- Database management and website design.

CERTIFICATIONS

- Beginning C++ Programming From Beginning to Beyond (Udemy)
- Hackerrank Python (Intermediate)
- The Web Development Bootcamp (Udemy)

INTERESTS

- Football
- Cricket
- Travelling

Society for Computer Technology & Research's

PUNE INSTITUTE OF COMPUTER TECHNOLOGY

(Approved by A.I.C.T.E. & Government of Maharashtra, affiliated to the Savitribai Phule Pune University & Accredited by NBA [All UG Programs] & NAAC)

Survey No. 27, Near Trimurti Chowk, Dhankawadi, Pune - 411043.

Tel. : +91 20 24371101/8063 **Fax No : +91 20 24364741**

Website : www.pict.edu **E-mail : principal@pict.edu**

EASEBUZZ		TRANSACTION ID : **EFS08NYP3U**	DATE : **2022-07-20**
PAID TO (INSTITUTE NAME) **PUNE INSTITUTE OF COMPUTER TECHNOLOGY**		STUDENT :	**MANDAL IMON ASHOK**
		Reg. No. : Roll No :	**C2K19106274** **41338**
Class :	**1-COMP**	Batch :	**BE-COMP-2022-23**

PARTICULARS

Installment	**Amount (Rs)**	**Penalty (Rs)**	**Total (Rs)**
Tution fees	44484.0	0.0	44484.0
Development fees	11033.0	0.0	11033.0
University fees	1218.0	0.0	1218.0
Amount (In Words) Fifty-six thousand, seven hundred and thirty-five Only		**Total (Rs)**	**56735.00**

Note :- *Subject to change from FRA.*
Computer generated receipt does not require seal or signature
All disputes are subject to Pune jurisdiction

This receipt is valid subject to bank realization.

Issued By: PICT (PUNE INSTITUTE OF COMPUTER TECHNOLOGY)