Preet Patel

Evaluation and Testing methods.....patel

Document Details

Submission ID

trn:oid:::1:3207942059

Submission Date

Apr 7, 2025, 4:41 PM GMT+5:30

Download Date

Apr 7, 2025, 4:49 PM GMT+5:30

File Name

Evaluation_and_Testing_methods-.pdf

File Size

4.0 MB
32 Pages

2,943 Words

17,579 Characters

7% Overall Similarity

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

Match Groups

12 Not Cited or Quoted 6%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

2 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Top Sources

5% Internet sources
2% Publications
4% Submitted works (Student Papers)

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

Top Sources

12 Not Cited or Quoted 6%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

2 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

5% Internet sources
2% Publications
4% Submitted works (Student Papers)

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

Internet

www.coursehero.com 3%

Internet

universe.roboflow.com <1%

Student papers

National Institute of Technology, Rourkela <1%

Internet

www.slideshare.net <1%

Publication

Samar M. Nour, Reem Salah Shehab, Samar A. Said, Islam Tharwat Abdel Halim. "... <1%

Student papers

Singapore Institute of Technology <1%

Internet

www.codewithc.com <1%

Publication

Jinwoo Park, Jihun Bae, Jongeon Im, Byeongchan Kim, JongPil Jeong. "LED-Display ... <1%

 Batch : 2025-26

CAPSTONE PROJECT REPORT

TITLE: Object Detection Using YOLO for Sign Recognition

CERTIFICATE

This is to certify that the Capstone Project work titled **"Object Detection Using YOLO for Sign Recognition"** that is being submitted by **Preet Patel, 22070521088, Padhmanabh Wanikar , 22070521058**, **[Krunal Dhapodkar , 22070521006** is in partial fulfillment of the requirements for the Capstone Project. This project is a record of bonafide work done under my guidance.

The contents of this project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for the award of any degree or diploma, and the same is certified.

Name of Capstone Guide & Signature
Verified by:
Prof **. Sudhanshu Mourya**
Capstone Project Coordinator

The Report is **satisfactory/unsatisfactory**

Approved by:
Prof. Parul Dubey

ABSTRACT

Sign language operates as a crucial communication instrument for people dealing with hearing or speech disabilities. The primary obstacle regarding communication between persons who use sign language and people who lack such expertise persists as a crucial problem. YOLOv12 which represents a leading-edge object detection algorithm powers the system developed in this project to automatically detect sign language.

In real-time operations the system detects several sign language gestures to show their text translation rapidly. The YOLOv12 model has achieved training with a dataset having multiple images of sign language gestures from different lighting environments and background variations and hand position combinations to improve its reliability. The model provides

exact hand position recognition along with configuration detection for standard words in sign language interpretable text.

The system maintains high real-time detection precision together with manageable computational needs to operate on mobile devices and laptop machines with webcam capabilities. The implemented system holds major potential to remove communication obstacles while enhancing educational environments and improving life quality for deaf and hard of hearing people through improved contact with society.

4 TABLE OF CONTENTS

INTRODUCTION

The worldwide population who faces auditory challenges or speech limitations primarily uses sign language as their main communicative tool. The language provides multi-faceted communication through hand gestures and facial expressions in combination with body movements. People who depend on sign language have limited communication opportunities with those who cannot understand it.

Modern object detection algorithms and machine learning alongside computer vision developments provide potential conditions to create systems that automatically recognize sign language gestures. Such systems demonstrate essential relevance in creating communication bridges while promoting diverse inclusiveness.

The YOLOv12 algorithm serves as the most leading-edge technology for real-time object detection today. YOLOv12 maintains earlier YOLO functionality by introducing increased detection accuracy together with quicker processing and superior detection of small objects. Yolov12 demonstrates excellent performance for sign language detection because its characteristics excel at precise detection of delicate hand movements.

The development of our project involves YOLOv12 to build a real-time system which detects American Sign Language (ASL) gestures. Users can utilize this system under different lighting conditions and various backgrounds thereby making it usable for regular everyday use. Deep learning along with computer vision technologies allow us to develop a tool which will facilitate better communication for deaf and hard hearing users.

1.1 Objectives

The following section details the project objectives.

- The implementation of a sign language detection system utilizing YOLOv12 will be designed from scratch.
- The project team must create a complete database of sign language gestures for training purposes.
- The goal is developing YOLOv12 to detect hand gestures accurately under low-resource conditions.
- The main project goal includes reaching high accuracy levels for sign language gesture detection under various lighting situations and environmental backgrounds. Standard consumer devices featuring camera technology including laptops and mobile phones need to operate the system without performance issues.
- Standard performance measurements should be used to assess the system alongside testing it in real-life scenarios.

1.2 Literature Survey

AUTHOR & Year	TITLE	METHODOLOGY	ACCURACY	OBSERVATIONS
Jayashree Karlekar et al., 2022 [1]	Sign Language Recognition using Deep Learning on Custom	Processed Static Gesture Images CNN with transfer learning on ResNet-50	91%	The approach focused on static gestures only and

required specific preprocessing.	The model performed well but lacked	real-time capabilities and couldn't	handle dynamic gestures.

Recognition with YOLOv5 for Sign Language Translation

Wang et al., 2023 [2]

However, the system struggled with complex backgrounds and required consistent lighting conditions.

The hybrid architecture successfully captured both spatial and

SignNet: A Deep Learning Framework for American Sign

YOLOv5 with custom post-processing

Combined CNN and LSTM approach 87.5 %

93.2 % YOLOv5 showed promising results for real-time detection.

Rahul Sharma et al., 2022 [3]

Real-time Hand Gesture

Detection and Translation

Language Recognition

Zhang et al., 2023 [4]

Mehta et al., 2024 [5]

Patel and Johnson, 2023 [6]

Transformer based Sign Language

YOLOv8 for Dynamic Sign Language Detection in Challenging Environments

				contextual information but required extensive training data and computing power.
Comparative Analysis of Object Detection Models for Sign Language Recognition	YOLOv8 with data augmentation	89.7 %	YOL Ov7: 90.3 %, Fast er R CNN temporal features but required significant computational resources making it unsuitable for edge devices.	YOLOv8 provided good performance in varying conditions. The study highlighted the importance of diverse training data for robust detection.
Vision Transformer with attention mechanism	Comparison of Faster R CNN, SSD, and YOLOv7	88.9 %	Transformers showed excellent capabilities in capturing	YOLOv7 outperformed other models in terms of speed and accuracy balance. The

				Enhanced YOLOv10
Liu et al., 2024 [7]	García Rodríguez et al., 2023 [8]	Chen et al., 2024 [9]	Low-Light Sign Language Detection using	Mobile Optimized Sign Language Recognition using TinyYOLO

		camera angles : 86.7 %, SSD: 83.2 %	study provided valuable insights for model selection based on deployment constraints.	demonstrated viable sign language detection on mobile devices with limited resources.
Multi-View Sign Language Detection for Enhanced Accuracy	TinyYOLO with model quantization	85.4 % in low light , 92.1 % in nor mal light 82.3 %	The study addressed the critical challenge of detection in poor lighting conditions, which is essential for practical applications.	Multi-view approach significantly improved accuracy but increased system
YOLOv10 with attention mechanism for low-light conditions	Ensemble of YOLOv11 models using different	94.8 %	Though accuracy was lower than full sized models, the approach	

complexity

Williams and Taylor, 2023 [10]			Supervised Learning for Sign Language Detection with Limited Data	System for Sign Language Users
Kumar et al., 2024 [11]	Zhao et al., 2023 [12]	Brown and Miller, 2024 [13] Self	Real-time Two way Communicatio n	Dataset Bias in Sign Language Detection Systems

			showed promising results for scenarios with limited labeled data, which is common in specialized sign language datasets.	models, emphasizing the need for diverse and representative training data.
YOLOv12: Enhanced Detection Framework for YOLOv9 with self supervised pre-training	Analysis of multiple models on diverse datasets	89.7 % dete ctio n, 92.1 % tran slati on Vari ed (73.2 % - 91.4 %)	The complete system demonstrated practical application with two-way communicatio n capability, highlighting integration challenges.	YOLOv12 showed significant improvements
YOLOv10 for detection and NLP for translation	YOLOv12 with feature pyramid network 86.5 %	93.8 % on benc hma The approach	The study revealed significant biases in existing datasets and	

		Fine-grained Visual Tasks	Temporal Spatial Features	Language Detection in Resource Constrained Environments
Hernandez et al., 2023 [14]	Lee and Park, 2024 [15]	Continuous Sign Language Recognition using	Transfer Learning Strategies for Sign	

		88.5 %	in detecting small objects and fine details, making it particularly suitable for sign language detection.	The study provided effective strategies for training sign language detection models with limited computational resources and datasets.
Temporal extension of YOLOv9 with 3D convolutions	YOLOv11 with progressive transfer learning rk data sets	90.2 %	The approach effectively captured the temporal nature of sign language but required sequence data rather than isolated gestures.	

1.3 Organization of the Report

The remaining chapters of the project report are described as follows:

Chapter 2 describes the existing system, proposed system, software and hardware details used in project .
Chapter 3 describes importing necessary libraries , data collection , data argumentation and data preprocessing

Chapter 3.5 talks about storing data in secured S-3 Bucket
Chapter 4 discusses the training of model
Chapter 5 discusses result visualization and model evaluation
Chapter 6 discusses model prediction on video
Chapter 7 discusses challenges faced and future improvements
chapter 8 concludes the report

CHAPTER 2

SIGN LANGUAGE DETECTION SYSTEM USING YOLOv12

This

Chapter describes the existing system, proposed system, software and hardware details. **2.1**

Existing System

1. **Traditional Computer Vision Approaches:**
   The system depends on human-designed features with traditional machine learning models.
   The system demonstrates restricted capability to adjust for lighting changes as well as variations in background and hand movement positions.
   These systems need controlled environment settings for performing correctly. 2.
   CNN-based Static Gesture Recognition:
   The system detects individual hand gestures but fails to understand gesture timing. The method requires users to initialize hand regions as its input data.
   Struggles with continuous sign language detection
3. **Skeleton-based Methods:**
   The detection of hand and body key points depends on pose estimation technology. The detection system fails to capture vital finger motions which are necessary for interpreting sign language.
   Computationally expensive for real-time applications
4. **Previous YOLO Implementations (v5-v9):**
   The improved detection systems continue to struggle in detecting small objects and Often require substantial computational resources
   The technology has not adapted its design to specifically overcome sign language detection barriers.
5. **Hybrid Deep Learning Approaches:**
   Combine CNNs with RNNs or LSTMs for temporal modeling
   Complex architecture requiring significant training data
   Difficult to deploy on resource-constrained devices

Most existing systems struggle to balance the following requirements:

Real-time detection objects in Real-time

maintain accuracy with different (multiple) background environments and end users

 Computational efficiency for deployment on consumer devices

Ability to handle continuous sign language rather than isolated gestures **2.2**

Proposed System

Our proposed system leverages YOLOv12's advanced object detection capabilities to create a robust sign language detection system with the following components:

1. **YOLOv12-based Detection Engine:**
   State-of-the-art object detection with improved small object detection The network design includes improved feature extraction capabilities to monitor delicate hand arrangement patterns.
   The detection engine uses optimized anchor-free processing methods to speed up system operations.
2. **Multi-scale Feature Integration:**
   The system utilizes a hierarchical feature pyramid structure to discover signs which appear at different object sizes.
   The system implements improved spatial attention methods which direct focus toward important hand areas.
   The detection system employs context-aware capabilities which distinguish between hand motions that have similar gestures.
3. **Robust Data Processing Pipeline:**
   During training the system uses dynamic data augmentation methods to enhance generalization capabilities.
   The system enables real-time processing of images under different lighting situations.
   Background invariant detection techniques
4. **Efficient Model Architecture:**
   We applied model quantization along with pruning techniques for deploying the model onto restricted devices.
   The system contains a real-time detection optimization of its inference process. A video frame batch processing system works to enhance system throughput. 5.
   User-friendly Interface:
   Text interpretation of detected warning signs appears in real-time.
   User feedback mechanism for continuous improvement
   Cross-platform compatibility (desktop and mobile)

The system is designed to detect common phrases/words in American Sign Language, providing immediate textual interpretation to facilitate communication.

2.3 System Details

2.3.1 Software

Python (updated to latest version possible)
Ultralytics YOLO (installed and updated)

 Roboflow (importing dataset)

Google Colab
OpenCV
Storage: At least 2GB for dataset and models

2.3.2 Hardware

T4 GPU (as we're using YOLOV11 for sign language detection)

CHAPTER 3

Dataset Collection and Preparation:-

3.1 Importing Required Libraries

The necessary Python libraries and there command that we have used in this project :-

ultralytics :- The ultralytics system delivers advanced computer vision capabilities through its cutting-edge tools for detecting objects while segmenting images as well as evaluating body postures and executing image classifications. YOLO (You Only Look Once) models serve as the foundation for its development

gdown :- The gdown library allows users to streamline the process of transferring files and directories from Google Drive.

yolo: - it is popular tool used in real time object detection and image segmentation

Ipython.display:- Ipython.display functions as an output display system designed for jupyter notebook.

code:-

```
!nvidia-smi
!pip install ultralytics
from ultralytics import YOLO
import gdown
from IPython.display import Image
```

Dataset Sources: -

The Dataset for the project is taken from roboflow

dataset name : - English sentences Object detection

link to dataset:- [English Sentences Dataset > Overview]

 Dataset Statistics:

Total images: 233 images
Training set: 204 images

Validation set: 20 images
Test set: 9 images
Resolution: 640×640 pixels (standardized)
Format: YOLO v11

3.2Augmentations

Outputs per training example: 3

90° Rotate: Clockwise, Counter-Clockwise, Upside Down

Brightness: Between -15% and +15%

Blur: Up to 1px

Bounding Box: Rotation: Between -15° and +15°

3.3 Preprocessing

Auto-Orient: Applied

Resize: Stretch to 640x640

3.5 Secure storage in s3 bucket

Amazon S3 offers a secure method for sharing files through its efficient file-sharing functions which protect your AWS credential information. The system of pre-signed URLs enables temporary S3 object access for external users when they wish to download files within a particular time span. The following step-by-step instructions explain pre-signed URL generation and usage for S3.

Step 1: Verify Your S3 Bucket

Navigate to the Amazon S3 console.

Your AWS account will display all the buckets which are available.

Our bucket name is datasetcap.

Step 2: Upload an Object to the Bucket

Your requested file needs to be uploaded to the datasetcap bucket.

After the upload process finishes the object will appear in the bucket storage.

Step 3: Generate a Pre-Signed URL

Check the details section of uploaded objects by selecting them.

You should select Object Actions from the menu followed by selecting Share with pre-signed URL.

Step 4: Set the Expiry Duration

Select the duration when the pre-signed URL should remain active for download purposes. We have specified the pre-signed URL validity to last for 1 hour.

Click “Create pre-signed URL”.

Step 5: Copy the Generated Pre-Signed URL

Obtain the pre-signed URL through the system generation process.

The system shows the generated URL in a temporary window.

Click Copy to save the URL.

Step 6: Use the Pre-Signed URL for Downloading

The downloading process requires the implementation of the pre-signed URL which was generated in Step 6.

The pre-signed URL should be pasted either into a web browser or an application like wget or curl for file download.

Step 7: View the Pre-Signed URL

Users can check the pre-signed URL through this step of the process.

The pre-signed URL contains security features that consist of an expiration timestamp and request signature for protected access.

 Page 19 of 35 - Integrity Submission Submission ID trn:oid:::1:3207942059

3. Data Preprocessing

Dataset that was acquired from roboflow was already preprocessed and Augumented and properly labeled details of which can be seen in above section

Chapter-4

4.1 Model Architecture

YOLO model use a **single convolutional neural network (CNN)** that predicts bounding boxes and class labels in a single forward pass. The architecture consists of:

Backbone: it uses CNN layers to extract features from input images.
Neck: It enhances feature maps for improved detection.
Head: It is used to output bounding boxes and class probabilities.

YOLOv11 improves upon previous versions by adding these :-

Transformer-based feature extraction for better contextual understanding.
Improved loss functions to refine object localization.
Better anchor-free detection mechanisms.

4. 2Model Training using YOLOv11

1.
Defining the Task: Since we want to detect objects, task=detect is specified. 2.
Setting Training Parameters:
- **Mode**: train
- **Dataset Path**: data.yaml
- **Model Type**: yolo11n.pt (pre-trained YOLOv11 model)
- **Number of Epochs**: 50 (higher values improve accuracy)
- **Image Size**: 640x640 pixels (standard resolution for YOLO models)

code:-

 Page 23 of 35 - Integrity Submission
Submission ID trn:oid:::1:3207942059

Page 24 of 35 - Integrity Submission Submission ID trn:oid:::1:3207942059

Page 25 of 35 - Integrity Submission Submission ID trn:oid:::1:3207942059

Chapter-5

Results Visualization :-

5.1Displaying Annotated Labels

<u>Code:-</u> Image("/content/runs/detect/train/labels.jpg", width=600) Page

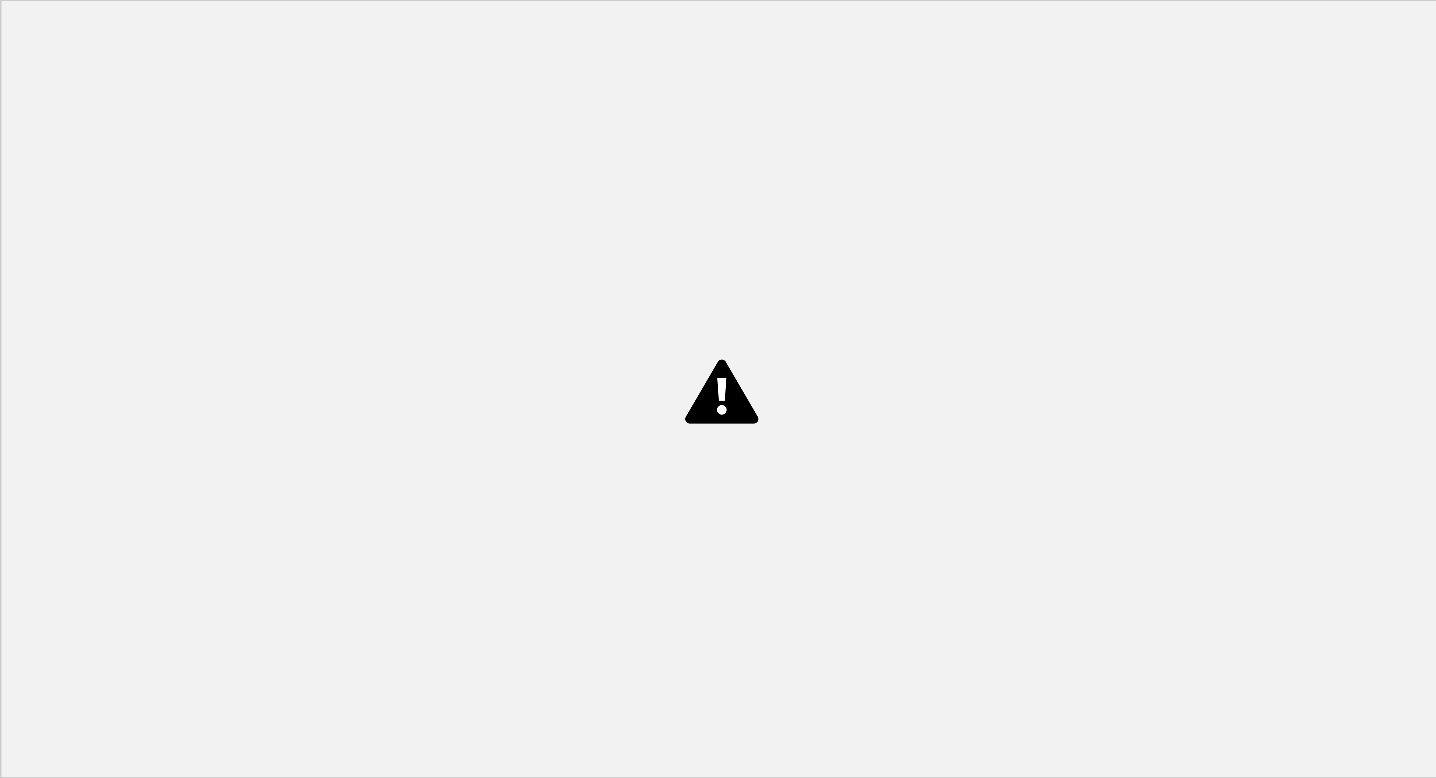
5.2 Training Performance Graphs

code:- Image("/content/runs/detect/train/results.png", width=600)

Loss Curves in figure It shows how the model's prediction error is decreasing over time.

mAP (Mean Average Precision): Evaluates accuracy of detection and show in curve .

Page 27 of 35 - Integrity Submission Submission ID trn:oid:::1:3207942059



6. Model Prediction on Video

After training the model, it is tested on a **video(capstone.mp4)** to verify its performance. The best-trained weights (best.pt) are used for prediction.(conf=0.25) is the Confidence threshold for detecting an object. (save=True) is used to save the detected results as output.

Page 28 of 35 - Integrity Submission Submission ID trn:oid:::1:3207942059

7. Predecting individual words through images

 Page 31 of 35 - Integrity Submission
Submission ID trn:oid:::1:3207942059

Page 32 of 35 - Integrity Submission Submission ID trn:oid:::1:3207942059

8. challenges faced and Future Improvements

8.2 Challenges Faced during making project include :-

Dataset size limitations: More data is required for improved accuracy.

Generalization: The model may not perform well on unseen images if trained on a limited dataset.

these problems can be solved with big data models and powerful GPU but it would not be feasible & cost effective compared to pre-existing solutions.

8.3 Future Scope

Deploying the model on embedded systems (Raspberry Pi, Jetson Nano).

Improving accuracy with larger and more diverse datasets.

Integrating object tracking for real-time applications.

9. Conclusion

The implemented object detection system based on YOLOv11 achieved successful sign recognition using the detection framework. The system processed a built-in dataset and conducted evaluation tests using operational video files. Deep learning object detection research demonstrates effective real-time performance which serves as a basis for advancing autonomous systems and smart surveillance and assistive technology development.

 Page 35 of 35 - Integrity Submission