

Department of Computer Science &Technology

**2021-2022**

Micro Project Report On

“**OBJECT DETECTION AND MATCHING**”

Bachelor of Technology In

COMPUTER SCIENCE AND ENGINEERING

(DATA SCIENCE)

By

B.LIKHITHA REDDY -21R25A6701 G.HARSH VARDHAN -21R25A6703

A.ASHWANTH -21R25A6702

SHARATH CHANDRA -20R21A6712

UNDER THE GUIDANCE OF

P.SRINIVAS REDDY

(Assistant Professor)



|  |  |  |
| --- | --- | --- |
|  | **CONTENTS** |  |
| **Certificate** |  | **ii** |
| **Declaration** |  | **iii** |
| **Acknowledgement** |  | **iv** |
| **Abstract** |  | **v** |
| **1. Introduction** |  | **1** |
| **2. Literature Review** |  | **2** |
| **3. Hardware Requirements** |  | **3** |
| **4. Software Requirements** |  | **3-10** |
| 4.1 Open CV |  |  |
| 4.2 NumPy |  |  |
| 1. **System Design**    1. System Architecture |  | **11** |
| **6.Implementation** | **12-14** | |
| **7.Testing&Results** | **15** | |
| **8.Conclusion** | **16** | |
| **9.References** | **17** | |



# CERTIFICATE

This is to certify that the project entitled “**OBJECT DETECTION AND MATCHING**” has been submitted by B. LIKHITHA REDDY - (21R25A6702), G. HARSH VARDHAN (21R25A6703), A. ASHWANTH

(21R25A6701), SHARATH CHANDRA-(20R21A6712) in the partial fulfillment of the requirements for the award of degree of Bachelor of Technology in Computer Science and Engineering (Data Science) from MLR Institute of Technology, Hyderabad. The results embodied in this project have not been submitted to any other University or Institution for the award of any degree or diploma.

**Internal Guide Head of the Department**

**External Examiner**



# DECLARATION

I hereby declare that the project entitled “**OBJECT DETECTION AND MATCHING**” is the work done during the period from OCT 2021 to JAN 2022 and is submitted in the partial fulfillment of the requirements for the award of degree of Bachelor of technology in Computer Science and Engineering (Data Science) from MLR Institute of Technology, Hyderabad. The results embodied in this project have not been submitted to any other university or Institution for the award of any degree or diploma.

21R25A6702-B. LIKHITHA REDDY

21R25A6703-G. HARSH VARDHAN

21R25A6701-A. ASHWANTH

20R21A6712-SHARATH CHANDRA



# ACKNOWLEDGEMENT

There are many people who helped me directly and indirectly to complete my project successfully. I would like to take this opportunity to thank one and all.

First of all I would like to express my deep gratitude towards my internal guide **Mr.P.Srinivas Reddy Asst.Prof,Department** of CS for his support in the completion of my dissertation. I wish to express my sincere thanks to **Dr. MADHURAVANI HOD, Department of CSE** and also to principal **Dr. K. Srinivas Rao** for providing the facilities to complete the dissertation.

I would like to thank all our faculty, coordinators and friends for their help and constructive criticism during the project period. Finally, I am very much indebted to our parents for their moral support and encouragement to achieve goals.

21R25A6702-B. LIKHITHA REDDY

21R25A6703-G. HARSH VARDHAN

21R25A6701-A. ASHWANTH

20R21A6712-SHARATH CHANDRA



**ABSTRACT**

This presentation considers the objective of efficient object detection and matching images. These

objectives lead to the proposed classification scheme that classifies the extracted features in new

images into object features and non-object features. It is shown out that this binary classification

scheme has turned out to be an efficient tool that can be used for object detection and matching. This

method considers the objective of accurate matching and robustness. Due to this classification, the

matching process becomes more robust and faster. In this case robust object registration also becomes

fast. It shows the advantages of using classification stage for object matching and registration using the

quantitative evaluation. This approach can be used for real-time object tracking and detection.

# INTRODUCTION

A feature is a piece of information which is relevant for solving the computational task related to a certain application. Features may be specific structures in the image such as points, edges or objects. Features may also be the result of a general neighbourhood operation or feature detection applied to the image. The features can be classified into two main categories:

* The features that are in specific locations of the images, such as mountain peaks, building corners, doorways, or interestingly shaped patches of snow. These kinds of localized features are often called **keypoint features**(or even corners) and are often described by the appearance of patches of pixels surrounding the point location.
* The features that can be matched based on their orientation and local appearance (edge profiles) are called **edges** and they can also be good indicators of object boundaries and occlusion events in the image sequence.

# LITERATURE REVIEW

a). View-based View - based methods work by comparing the object of interest (OOI) and the

precalculated 2D views of that object, denoted by aspect graphs. This method can provide a rough

estimate of the pose of the object, but it suffers from computational load of searching a large

spatial domain. To overcome this issue, Cyr and Kimia [3] proposed a method that grouped

similar OOI together under a single class and similarity scores were calculated to perform

recognition. Eggert et al. [1], Ulrich et al. [8] and Steger et al. [2] introduced various

improvements on the aspect graphs and similarity metric. This approach, however was not widely

used because of its high complexity.

b). In the study of texture-less object detection, template matching is a popular method dating

back as early as the 2000s in lieu of the fact that it provides high mAP and close to real-time

detection. But, up until now, templates were generated using one modality only (depth or colour

or point cloud or 2D). However, in 2011 Hinterstoisser et al. [6] put forth the idea of multi-modal

template matching. The template comprises of both the image cue (colour information) as well as

depth cue (3D) information. The method can detect objects in real-time as well as under heavy

clutter. The method extracts gradients from colour images as they are more robust to illumination

changes compared to gradients extracted from grayscale images. Also, 3D surface normal is

calculated which are generally found on the inside of the object. The gradients and surface-

normals are summed to obtain a multi-modal template. For new objects, one only needs to extract

and store multi-modal features which are instantaneous. Hence, it is real-time object recognition.

# HARDWARE REQUIREMENTS

* Laptop/PC
* Webcam/Camera module

# SOFTWARE REQUIREMENTS

#### 1.Open CV

* OpenCV( Open-Source Computer Vision ) is a cross-platform library using which we can

develop real-time computer vision applications.

* It mainly focuses on image processing.
* Analysis including features like face detection and object detection.
* OpenCV features GPU acceleration for real-time operations.
* The library has more than 2500 optimized Machine Learning algorithms.
* These Algorithms are used to recognize faces, identify
* objects recognize faces.
* identify object’s, find similar images from an image database.

**2. NumPy**

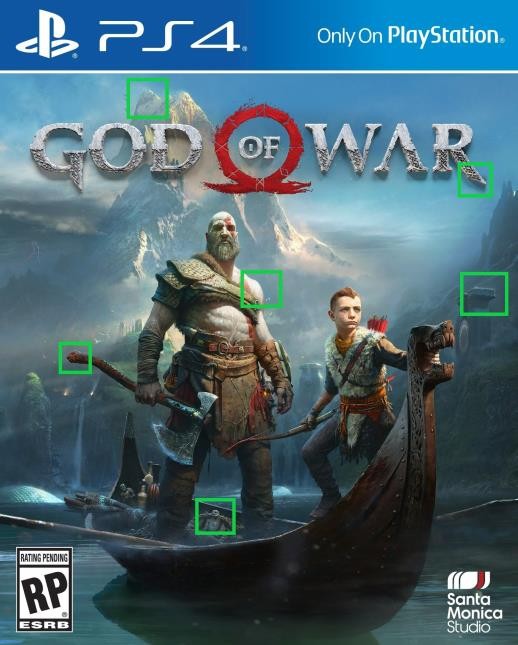
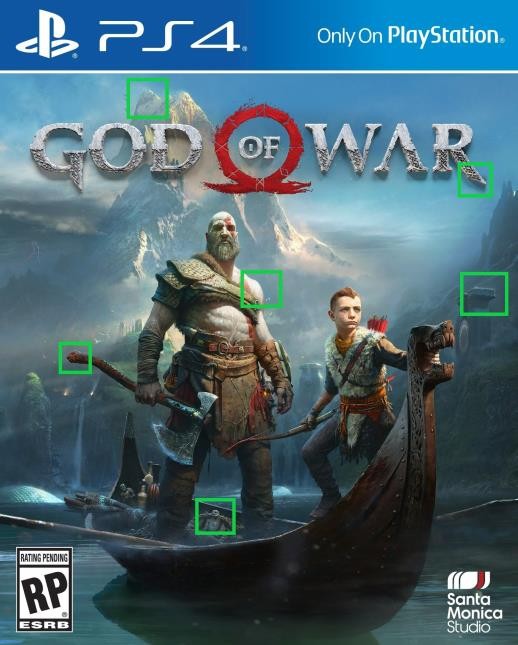
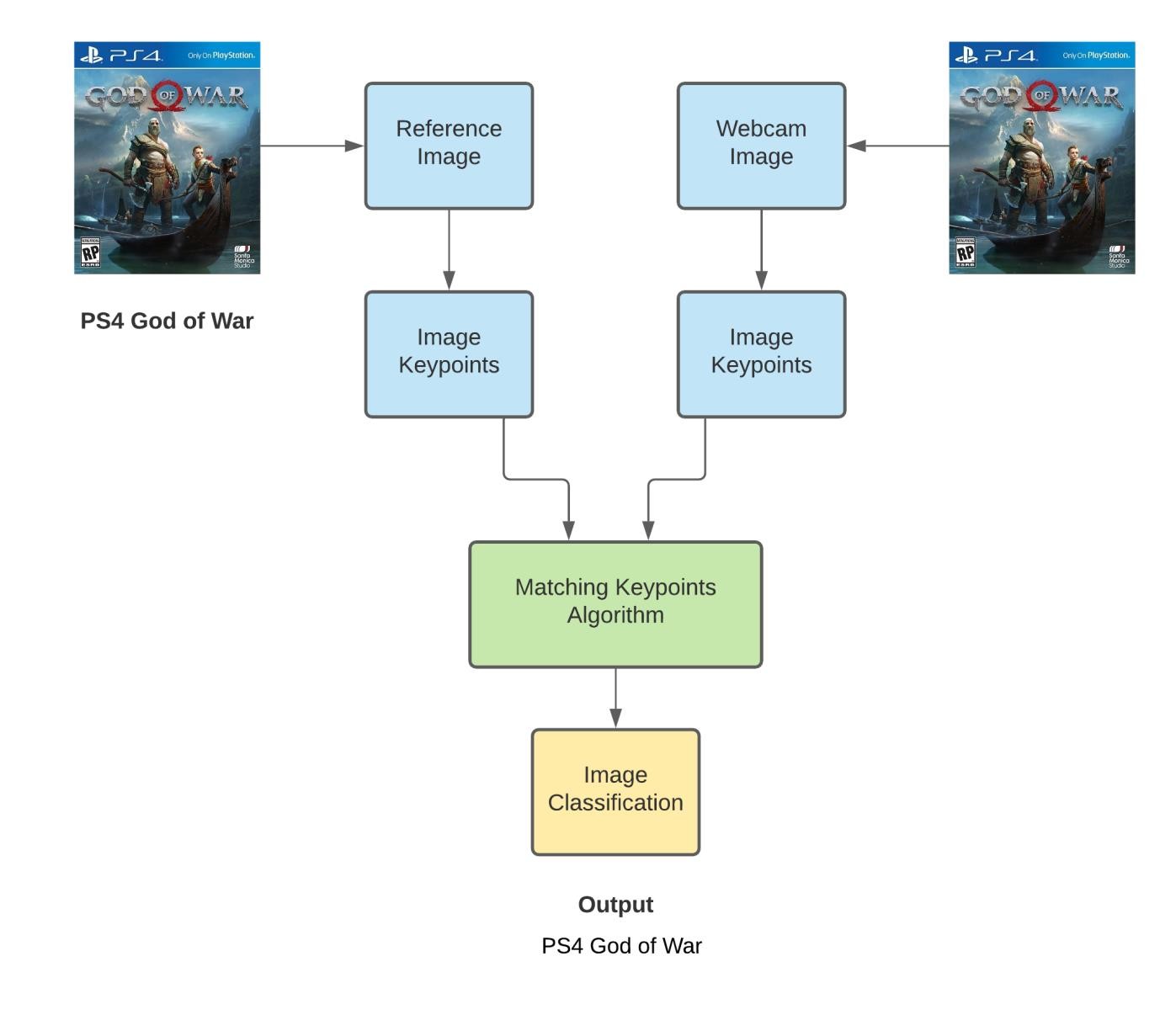
* NumPy is a Python library used for working with arrays.
* It is the fundamental package for scientific computing in Python.
* It is also a library that provides a multidimensional array object

and various derived objects.

* It can perform mathematical, logical, shape manipulation.
* Sorting, selecting, I/O, discrete Fourier transforms.
* Basic linear algebra, basic statistical operations,.
* Random simulation and much more.
* At the core of the NumPy package, is the nd array object.
* This encapsulates n-dimensional arrays of homogeneous data types,

with many operations being performed in compiled code for performance.

**SYSTEM DESIGN**

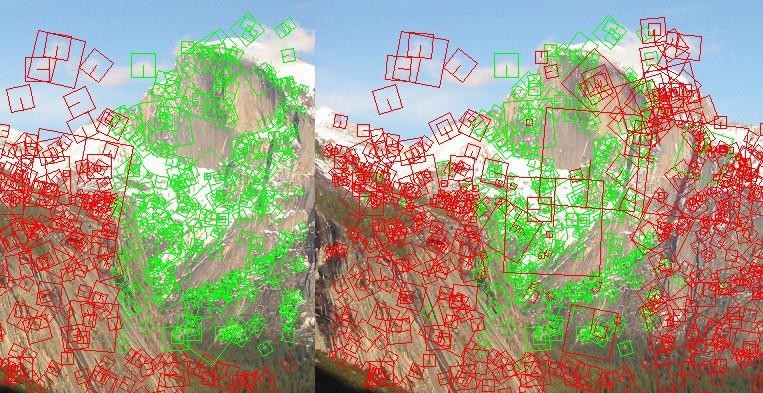


**IMPLEMENTATION**

The Algorithms in OpenCv package will take the reference image and creates a key points

in particular areas of image and will save those key points and descriptor(binary data) as shown

below



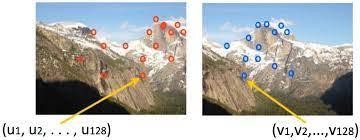
Later when we give input from webcam/camera module in the form of visual data. The algorithms

will again create the key points for that image and try to match with the reference image key

point’s.

If the key points match then it gives output feedback in the form of

classification of that image as we specified as shown below..



# Main Component Of Feature Detection And Matching

**Detection:**

Identify the Interest Point.

**Description:**

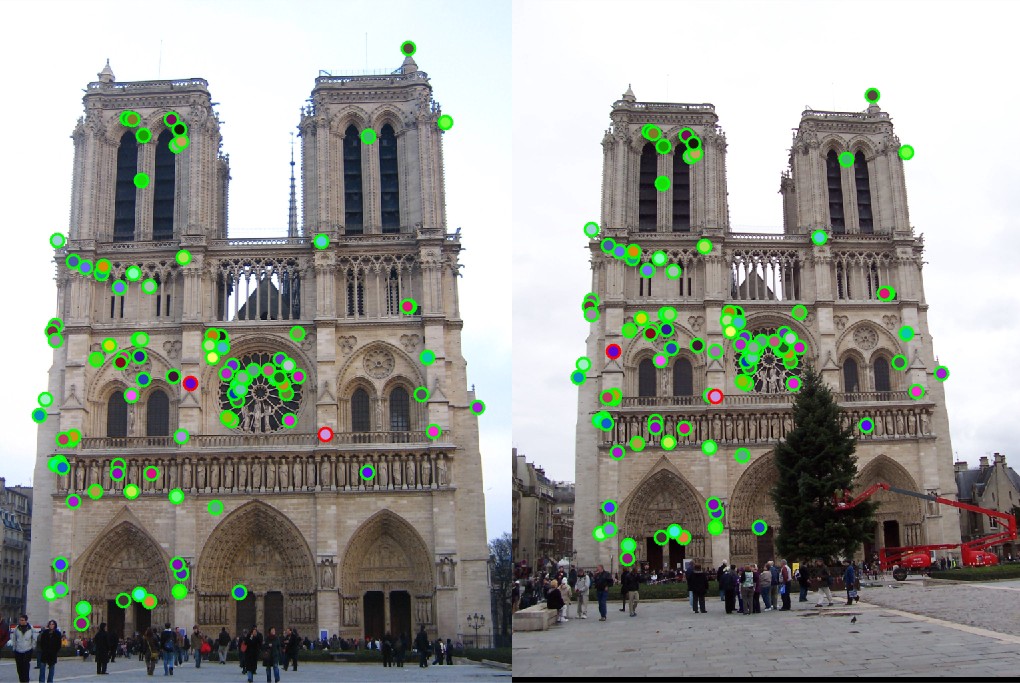
The local appearance around each feature point is described in some way that is (ideally) invariant under changes in illumination, translation, scale, and in-plane rotation. We typically end up with a descriptor vector for each feature point.

**Matching:**

Descriptors are compared across the images, to identify similar features. For two images we may get a set of pairs (Xi, Yi) ↔ (Xi`, Yi`), where (Xi, Yi) is a feature in one image and (Xi`, Yi`) its matching feature in the other image.

**Interest Point:**

Interest point or Feature Point is the point which is expressive in texture.Interest point is the point at which the direction of the boundary of the object changes abruptly or intersection point between two or more edge segments.



## Properties Of Interest Point

* It has a well-defined position in image space or well localized.
* It is stable under local and global perturbations in the image domain as illumination/brightness variations, such that the interest points can be reliably computed with a high degree of repeatability.
* Should provide efficient detection.

**Feature Descriptor**

A feature descriptor is an algorithm which takes an image and outputs feature descriptors/feature vectors. Feature descriptors encode interesting information into a series of numbers and act as a sort of numerical “fingerprint” that can be used to differentiate one feature from another.

# 

Ideally, this information would be invariant under image transformation, so we can find the feature again even if the image is transformed in some way. After detecting interest point we go on to compute a descriptor for every one of them. Descriptors can be categorized into two classes:

**Local Descriptor:**

It is a compact representation of a point’s local neighborhood. Local descriptors try to resemble shape and appearance only in a local neighborhood around a point and thus are very suitable for representing it in terms of matching.

**Global Descriptor**:

A global descriptor describes the whole image. They are generally not very robust as a change in part of the image may cause it to fail as it will affect the resulting descriptor.

# Feature Matching:

Features matching or generally image matching, a part of many computer vision applications such as image registration, camera calibration and object recognition, is the task of establishing correspondences between two images of the same scene/object. A common approach to image matching consists of detecting a set of interest points each associated with image descriptors from image data. Once the features and their descriptors have been extracted from two or more images, the next step is to establish preliminary feature matches between these images.

# TESTING & RESULTS





**CONCLUSION**

This approach for detecting and matching an object with an image will show the use of

feature classification makes the whole process of matching and registration faster and more

efficient. Generally, the performance of matching methods based on interest points depends on

both the properties of the underlying interest points and the choice of associated image

descriptors. Thus, detectors and descriptors appropriate for images contents shall be used in

applications. For instance, if an image contains bacteria cells, the blob detector should be used

rather than the corner detector. But, if the image is an aerial view of a city, the corner detector is

suitable to find man-made structures. Furthermore, selecting a detector and a descriptor that

addresses the image degradation is very important.

REFERENCES

* Bradski and Kaehler. Learning OpenCV: Computer Vision with the OpenCV Library. O’Reilly, 2008.
* M. Brown, R. Szeliski, and S. Winder. Multi-image matching using multi-scale oriented patches. In Proc. CVPR pages 510–517, 2005.
* H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. In Proc. ECCV, pages 404–417, 2006.
* C. Harris and M. J. Stephens. A combined corner and edge detector. In Alvey Vision Conference, pages 147–152, 1988.