



SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE

A

MINI PROJECT REPORT

ON

“Feature Extraction using Zernike Moments”

SUBMITTED TO THE SAVITRIBAI PHULE PUNE
UNIVERSITY, PUNE IN THE FULFILLMENT OF THE
REQUIREMENT
OF

Laboratory Practice VI

Fourth Year Computer Engineering

Academic Year 2024-25

BY

Name of Students:

Exam No.:

Abhang Pratik Sunil

B1905704201

Borase Darshana Ravasaheb

B1905704228

Darekar Omkar Balasaheb

B1905704232

Gunde Chaitanya Diliprao

B1905704256

Under the Guidance of

Mr. N. V. Kamble



Sinhgad Institutes

DEPARTMENT OF COMPUTER ENGINEERING
STES'S SINHGAD INSTITUTE OF TECHNOLOGY AND SCIENCE
NARHE, PUNE – 411041



Sinhgad Institutes

Department of Computer Engineering
Sinhgad Institute of Technology and Science, Narhe, Pune

CERTIFICATE

This is to certify that,

Name of Students:

Exam No.:

Abhang Pratik Sunil

B1905704201

Borase Darshana Ravasaheb

B1905704228

Darekar Omkar Balasaheb

B1905704232

Gunde Chaitanya Diliprao

B1905704256

studying in BE Computer Engineering Course SEM-VIII has successfully completed their LP-VI Lab Mini-Project work titled **“Feature Extraction using Zernike Moments”** at Sinhgad Institute of Technology and Science, Narhe in the fulfillment of the bachelor’s degree in engineering of Savitribai Phule Pune University, during the academic year 2024-2025.

Mr. N. V. Kamble

Guide

Dr. G. S. Navale

Head of Department

Dr. S. D. Markande

Principal

SINHGAD INSTITUTE OF TECHNOLOGY AND SCIENCE, NARHE, PUNE– 411041

Place : Pune

Date :

ACKNOWLEDGEMENT

We take this opportunity to acknowledge each and every one who contributed towards our work. We express our sincere gratitude towards guide **Mr. N. V. Kamble**, Assistant Professor at Sinhgad Institute of Technology and Science, Narhe, Pune for his valuable inputs, guidance and support throughout the course.

We wish to express our thanks to **Dr. G. S. Navale**, Head of Computer Engineering Department, Sinhgad Institute of Technology and Science, Narhe for giving us all the help and important suggestions all over the Work.

We thank all the teaching staff members for their indispensable support and priceless suggestions. We also thank our friends and family for helping in collecting data, without their help **Feature Extraction using Zernike Moments** report have been completed. At the end our special thanks to **Dr. S. D. Markande**, Principal Sinhgad Institute of Technology and Science, Narhe for providing ambience in the college, which motivates us to work.

Name of Students:

Sign:

Abhang Pratik Sunil

Borase Darshana Ravasaheb

Darekar Omkar Balasaheb

Gunde Chaitanya Diliprao

CONTENTS

Sr. No.	Title	Page No.
1.	Introduction	3
2.	Objective	4
3.	Implementation	5
4.	Result and Analysis	11
5.	Conclusion	12
6.	References	13

1. INTRODUCTION

Moments have been used in image processing and classification type problems since Hu introduced them in his groundbreaking publication on moment invariants [4]. Hu used geometric moments and showed that they can be made to be translation and scale invariant. Since then more powerful techniques have been developed.

A notable example is Teague's work on Zernike Moments (ZM); he was the first to use the Zernike polynomials (ZP) as basic functions for the moments [6]. ZM's have been used in a multitude of applications with great success and some with 99% classification accuracy [1]. The use of ZPs as a basis function is theoretically beneficial because they are orthogonal polynomials which allows for maximum separation of data points, given that it reduces information redundancy between the moments. Their orthogonal properties make them simpler to use during the reconstruction process as well.

Furthermore, the magnitude of ZM's is rotationally invariant, which is crucial for certain image processing applications, such as classifying shapes that are not aligned. The breakdown of this paper is as follows. Section 2 introduces standard geometric moments and their application in achieving rotational and translational invariance of the images. Section 3 defines Zernike polynomials and Zernike moments. Section 4 explains the rotational invariant properties of the magnitudes of Zernike moments. Section 5 discusses how to achieve reconstruction from Zernike moments. Furthermore, it describes how to use these reconstructions to apply weights on these moments so as to make the moments which capture the most amount of information about the image contributes the most amount of 'votes' for the classification. Section 6 gives a description of the experiments used to test the unsupervised classification accuracy of Zernike moment feature vectors.

In Section 7 we will present our conclusion of our results, and our justification for why these results should be used in applicable image classification problems. Finally, In Section 8 a brief overview of future research is presented.

2.OBJECTIVE

1. To understand the concept of feature extraction and its importance in machine learning.
2. To understand the concept of Zernike moments and their mathematical properties.
3. To implement feature extraction using Zernike moments in a programming language.
4. To apply the feature extraction technique to a dataset of images and analyze the results.
5. To compare the performance of Zernike feature extraction with other techniques.

3. IMPLEMENTATION

Let's go ahead and introduce our two images. The first image, our reference image, can be seen below:



Here we can see a [Game Boy ROM cartridge](#) of game, Pokemon.

Given this reference image of the Pokemon game cartridge, our goal is to detect it in the following image filled with distractor objects:

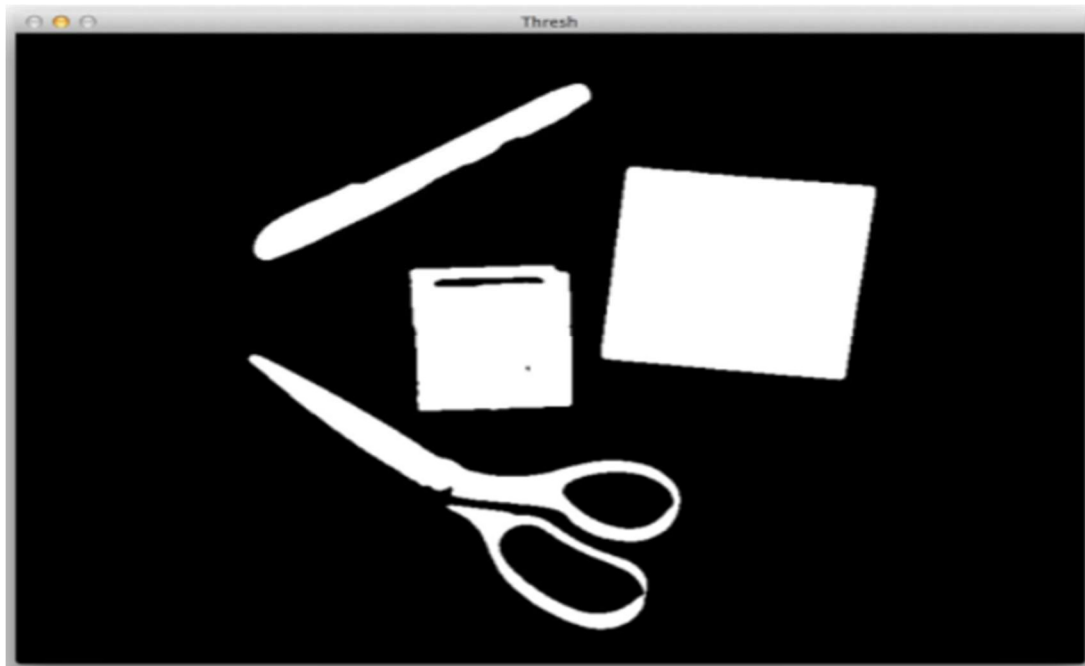


In this image we can see that we have the Pokemon game cartridge. But also also have a bunch of other distractor objects, such as scissors, a highlighter, and a sticky note.

The first thing we'll do here is define a *describe_shapes* function. This function will be applied to the Pokemon game cartridge in the reference image along with *every* object in the second image. Given that this section will be called so many times, I thought it best to create a dedicated function to describing the shape regions.

- The *describe_shapes* function will take a single argument, an image that contains the objects and shapes we want to quantify using Zernike Moments.

We'll need to apply a little bit of pre-processing to our image before we can extract our shape features, so we'll convert the image to grayscale, blur it to remove high frequency noise and allow us to focus on the structural aspects of the image, followed by thresholding to segment the objects from the background.



We'll also perform a series of dilations and erosions to close gaps between any parts of objects.



Now that the [Morphological Operations](#) have been applied, we can then detect the contours of each of the objects in the image.

We start looping over each of the individual contours. And then for each of these contours, we allocate memory for a *mask* followed by drawing the contoured region on the mask.

Now that we have *just* the contoured region drawn on the mask, we can then compute the bounding box and extract the ROI (from the mask image). Again, this step mentioned above to ensure that *only* the current shape was drawn on the mask and *nothing* else.

Given *the ROI of the shape*, we can now extract *Zernike Moments* by making a call to the *mahotas.features.zernike_moments* function. This function accepts 3 parameters: *the image/ROI* that we want to quantify, the *radius* of the region, and the *degree* of the polynomial.

**** *Tuning the radius*** can be *non-trivial based on the sizes of the objects* in the image. If the sizes of the objects vary dramatically, as they do in our distractor image above, the value of this radius is often not immediately obvious.

To solve this problem, we have 2 solutions:

- The first solution is to simply *resize the ROI* to a known size. This ensures that we will be able to hardcode the radius value and that our images are described in a consistent manner.
- The **problem** with this approach is that resizing to a fixed size *can destroy the aspect ratio* of the shape. This is especially troublesome if we are trying to quantify the shape of a region — by throwing away the aspect ratio, we throw away information regarding the dimensions of the shape, which is completely counterintuitive to our goal!

The other solution is what this example utilizes — we can *dynamically* compute the radius of the Zernike Moments simply by computing the *cv2.minEnclosingCircle* function (a [simple contour property](#)). This method will return the minimum size radius that can be

used to enclose the *entire* object. By applying this method, we can ensure the radius of the moments covers the entire ROI.

- We also supply `degree=8` to the *Zernike moments* function, which is the default degree of the polynomial. In most cases you'll need to tune this value until it obtains adequate results.

Finally, we can return a **2-tuple** from us *describe_shape's* function consisting of (1) the contours of the objects/shapes in the image, followed by (2) the Zernike Moments feature vectors corresponding to each shape.

So now that we have the *shape's* function defined, let's see how we can use it to recognize the Pokemon game cartridge in an image:

We load our reference image of the Pokemon game cartridge from disk and describe shape of the game cartridge.

We load the distractor image from disk and quantify *all* of the shapes in the image using Zernike Moments.

To detect the actual game cartridge, we use SciPy's [distance](#) sub-module to compute the Euclidean distance between all pairs of the *gameFeatures* and *shapeFeatures*.

- Notice how we have only one row, which is the Zernike Moments associated with the game cartridge features. We also have multiple columns — one for each of the shapes in the *shapeFeatures* list.

The index of the column that *minimizes* this distance is thus our Pokemon game cartridge. How can, we be sure??

Zernike Moments are used to quantify the shape of an object. If we assume shapes that have *similar feature vectors* also have *similar visual contents*, then the shape that minimizes the distance between Zernike Moments must be our reference image!

We'll start by looping over the contours in our distractor's image. If the index of the current contour does not match the index of the contour with minimum distance in our distance matrix D , then we draw a rotated bounding box surrounding it in *red*.

We then draw a *green* rotated bounding box surrounding our identified game cartridge region, followed by displaying the text "FOUND!" directly above the bounding box.

4. RESULT & ANALYSIS



On the *left* we have our original reference image of the Pokemon game cartridge. And on the *right*, we have our distractor image containing not only the Pokemon game cartridge, but other objects and shapes meant to “confuse” our algorithm.

However, by applying Zernike Moments features we were able to discard the distractor shapes and detect the actual game cartridge shape in the image.

5. CONCLUSION

In conclusion, when describing multiple shapes in an image using Zernike Moments, it is important to follow a few key steps. First, extract the region of interest (ROI) for each object in the image. Then, apply Zernike Moments to each ROI to capture the shape characteristics. When utilizing Zernike Moments, it is crucial to consider the radius and degree parameters. The radius determines the size of the disc onto which the shape is mapped. It is essential to choose a radius that is sufficiently large to include all relevant pixels of the shape.

6. REFERENCES

- [1] Khotanzad A. and Hong Y. H. Invariant image recognition by zernike moments. IEEE, 12(5):489 – 497, 1990.
- [2] Suk T. Flusser J. and Zitova B. Moments and Moment Invariants in Pattern Recognition. Wiley and Sons Ltd., 2009.
- [3] Fan X.X. Fu B., Liu J. and Quan Y. A hybrid algorithm of fast and accurate computing zernike moments. IEEE, 2007.
- [4] Hu M. K. Visual pattern recognition by moment invariants. IRE Transactions on Information Theory, 8(2):179 – 187, 1962.
- [5] Yu M. Feature-Weighted Hierarchical Sparse Learning for Semantic Concept Identification.
- [6] Teague M. R. Image analysis via the general theory of moments. Optical Society of America, 70(8):920 – 930, 1979.
- [7] Prokip R.J. and Reeves A.P. A survey of moment-based techniques for unoccluded object representation and recognition. CVGIP: Graphical Models and Image Processing, 54(5):438 – 460, 1991.
- [8] Witlin R.S. The witlin center for advanced eyecare, 2008.