AI in Image Processing

A Project Report Submitted
in Partial Fulfillment of the Requirements
for the Degree of

Bachelor of Technology

by

Kaushal Kishore
(111601008)

under the guidance of

Dr. Chandra Shekar

INDIAN INSTITUTE
OF TECHNOLOGY
PALAKKAD

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

*This is to certify that the work contained in this thesis entitled "**AI in Image Processing**" is a bonafide work of **Kaushal Kishore (Roll No. 111601008)**, carried out in the Department of Computer Science and Engineering, Indian Institute of Technology Palakkad under my supervision and that it has not been submitted elsewhere for a degree.*

Dr. Chandra Shekar
Assistant Professor
Department of Computer Science & Engineering
Indian Institute of Technology Palakkad

Acknowledgements

Write acknowledgements, if your want to.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Image processing libraries these days (eg. Open CV) uses the conventional methods which have the possibility to be outperformed by methods which leverage the power of artificial intelligence. Some recent research have shown that some of these AI based methods are able to perform atleast as good as conventional approaches. The aim of this project is to implement, apply and possibly improve upon the existing approaches in Digital Image Processing and Computer Vision. These common tasks can include (not limited to) applications like: Image Compression, Denoising, Super Resolution, Flow Estimation, Object Detection, etc.

1.1 Image Processing

Image processing is manipulating an image in order to enhance it or extract information from it. It is widely useed in medical visualization, biometrics, self-driving vehicles, gaming, surveillance, and law enforcement. It can used in various ways: visualization, restoration, imformation retrieval, pattern recognition, etc.

General approach of image processing involves eight key phases: image acquisition, image enhancement, image restoration, color space transformation, compression or decom-

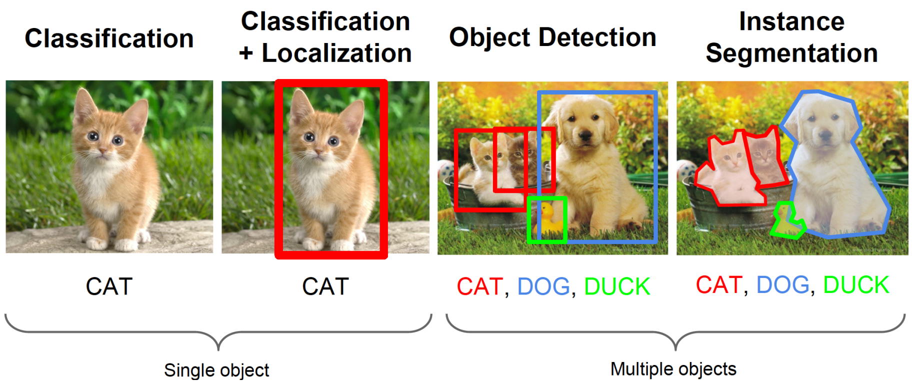

Fig. 1.1: Examples of pattern recognition
Source: www.cs.cornell.edu

pression, morphological processing, recognition, and representation. It is very difficult to carry out these steps manually on a very big data, this is where AI and ML algorithms become very helpful.

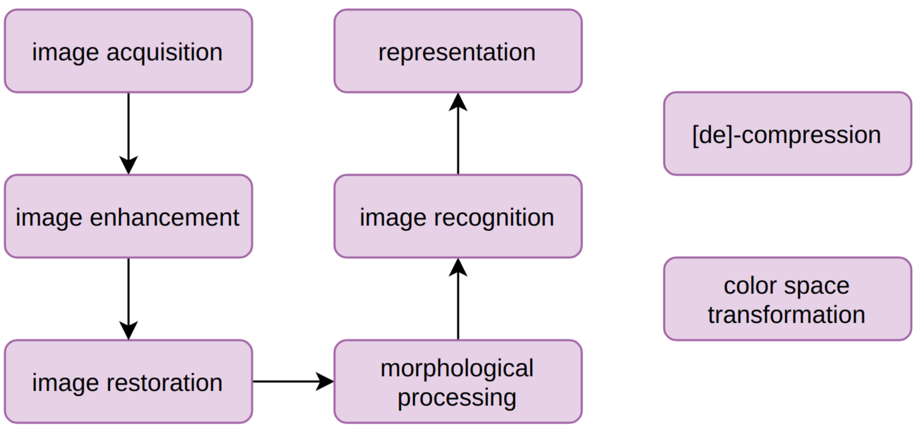

Fig. 1.2: Key phases of image processing

1.2 Recent advances

Modern AI algorithms have enabled computer to perform detection, segmentation, recognition, compression, extraction, generation, and discrimination. Every year a state of the art model is invented to solve the existing problem in a better way. It is now an established fact that machines are now better than humans in counting, classifying and segmenting instances.

1.3 Organization of The Report

Chapter 2

Image Compression

A data compression algorithm transforms the data to occupy a less space. The original data is encoded by a program called encoder, to a compressed representation using a fewer number of bits. Decoder is responsible for decompressing the compressed representation.

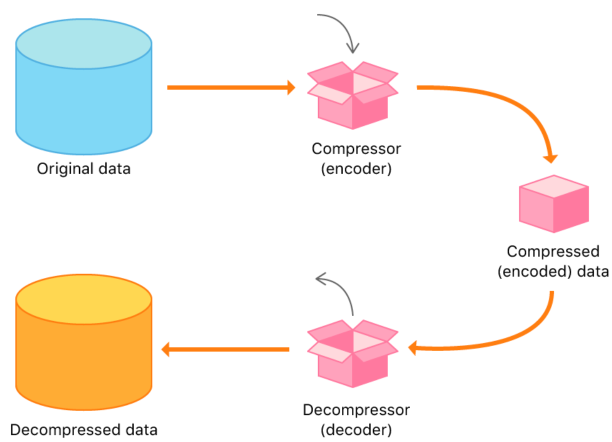

Fig. 2.1: Compression phases
Source: `https://developer.apple.com/documentation/compression`

Image compression is very crucial in order to reduce the size of disk space used as well as reduce the amount of internet bandwidth used while loading images. It's also important to compress images for people accessing the internet via low bandwidth connections.

2.1 Lossy vs. Lossless

The compression technique where the decompressed data is exactly same as original data is called as lossless compression otherwise it is known as lossy compression technique because some information is lost during coding-encoding phase. Two well-known codecs for image compression are JPEG and PNG. PNG is lossless and JPEG is lossy.

Lossy	Lossless
A compression that permits reconstruction only of an approximation of the original data, though usually with an improved compression rate.	A class of data compression that allows the original data to be perfectly reconstructed from the compressed data.
Reduces the quality.	Does not reduce the quality.
Data reduction is higher.	Data reduction is lower.
Commonly used to compress multimedia data such as audio (MP3), video and image (JPEG) files.	Used commonly for text, data files, etc.

Table 2.1: Lossy vs Lossless

2.2 Lossy Compression using PCA

Principal components analysis (PCA) is one of a family of techniques for taking high-dimensional data, and using the dependencies between the variables to represent it in a more tractable, lower-dimensional form, without losing too much information. PCA is one of the simplest and most robust ways of doing such dimensionality reduction.

PCA is mathematically defined as an orthogonal linear transformation that transforms the data to a new coordinate system such that the greatest variance by some scalar projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on.

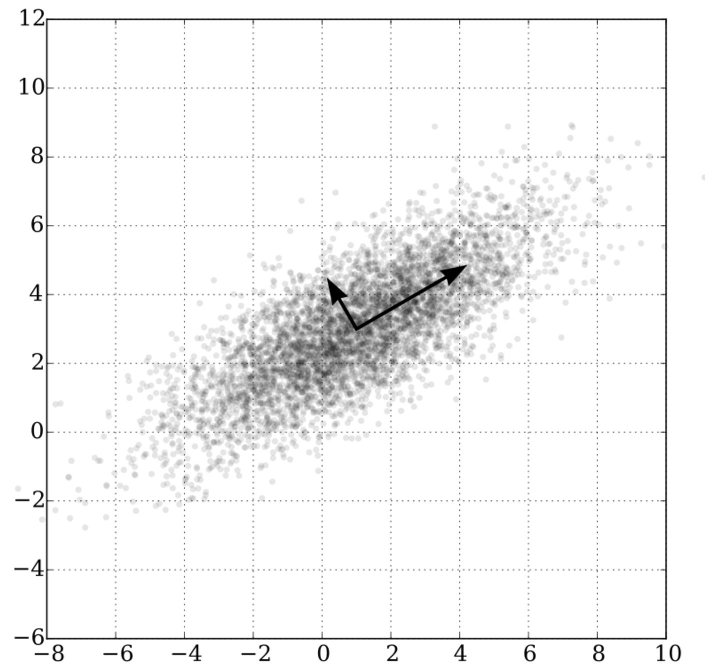

Fig. 2.2: PCA of a multivariate Gaussian distribution centered at (1,3) with a standard deviation of 3 in roughly the (0.866, 0.5) direction and of 1 in the orthogonal direction. **Source:** `https://en.wikipedia.org/wiki/Principal_component_analysis`

Let W be a $d \times d$ matrix whose columns are the principal components of X. The transformation $T = XW$ maps a data vector $x_{(i)}$ from an original space of d variables to a new space of d variables which are uncorrelated over the dataset. However, not all the principal components need to be kept. Keeping only the first L principal components, produced by using only the first L eigenvectors, gives the truncated transformation:

$T_L = XW_L$

where the matrix T_L now has n rows but only L columns. In other words, PCA learns a linear transformation $t = W^T x, x \in \Re^d, t \in \Re^L$, where the columns of $d \times L$ matrix W form an orthogonal basis for the L features (the components of representation t) that are decorrelated. By construction, of all the transformed data matrices with only L columns, this score matrix maximises the variance in the original data that has been preserved, while minimising the total squared reconstruction error $\left\|TW^T - T_L W_L^T\right\|_2^2 \, or \, \|X - X_L\|_2^2$.

The basic steps for computing the PCA is as follows:

1. Standardize the d-dimensional dataset.

2. Construct the covariance matrix.

3. Decompose the covariance matrix into its eigenvectors and eigenvalues.

4. Sort the eigenvalues by decreasing order to rank the corresponding eigenvectors.

5. Select L eigenvectors which correspond to the L largest eigenvalues, where L is the dimensionality of the new feature subspace $L \leq d$.

6. Construct a projection matrix W_L from the "top" L eigenvectors.

7. Transform the d-dimensional input dataset X using the projection matrix W_L to obtain the new L-dimensional feature subspace.

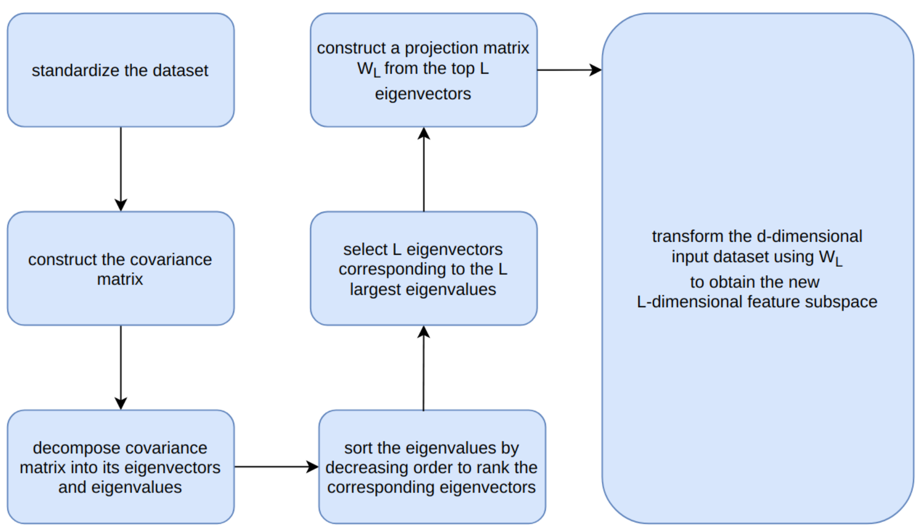

Fig. 2.3: Steps for dimensionality reduction using PCA

Chapter 3

Algorithm I

give details of your algorithm

3.1 Conclusion

In this chapter, we proposed a distributed algorithm for construction of xyz. The complexity of this algorithm is $O(n \log n)$. Next chapter presents another distributed algorithm which has linear time complexity based on xyz.

Chapter 4

Algorithm II

The algorithm presented in previous chapter has $O(n)$ time complexity. We further propose another distributed algorithm in this chapter based on xyz which has linear time complexity.

4.1 Construction

Write ...

4.2 Improved Method

Write...

4.3 Conclusion

In this chapter, we proposed another distributed algorithm for XYZ. This algorithm has both time complexity of $O(n)$ where n is the total number of nodes. In next chapter, we conclude and discuss some of the future aspects.

Chapter 5

Conclusion and Future Work

write results of your thesis and future work.

References