

## **MODULE-03**

### **INDIVIDUAL TASK**

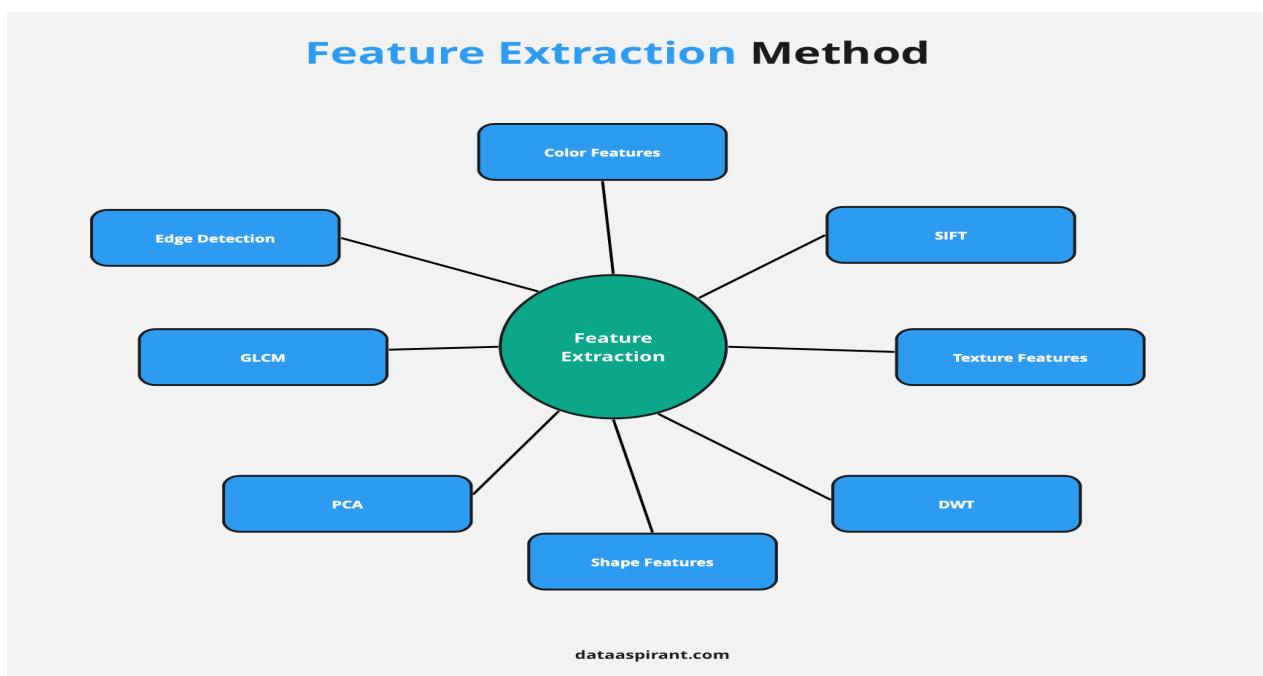
- **Feature Extraction Thought Experiment: Select a dataset (e.g., photos, shopping lists) and describe which features would be important to a machine learning model.**

#### **Introduction to Feature Extraction:**

- Feature extraction is a process of transforming raw data into meaningful features for analysis.
- It simplifies data by highlighting relevant information and reducing noise.
- Effective feature extraction is crucial for improving machine learning model performance.

#### **Choosing a Dataset:**

- For this presentation, we'll select a dataset of digital photos.
- Photos contain rich visual information suitable for feature extraction.
- This dataset allows exploration of both low level and high-level features.



## **Characteristics of the Photo Dataset:**

- The dataset includes images of various objects, objects, scenes, and lighting conditions.
- Images are typically stored in formats like JPEG or PNG.
- Metadata such as resolution and timestamps may also be available.

## **Goals of Feature Extraction for Photos:**

- To identify key visual elements that distinguish different images.
- To reduce the dimensionality of image data for easier processing.
- To enhance the effectiveness of image classification and retrieval tasks.

## **Raw Data in Photos:**

- Raw images are pixel matrices with RGB color channels.
- Each pixel contains intensity values for red, green, and blue.
- The high dimensionality of raw images makes direct analysis challenging.

TDT4259 – Applied Data Science

Lecture 2: Bringing value from datasets

Nisha Dalal  
Adj. Associate Professor

nisha.dalal@ntnu.no

## Agenda

- Introduction
- 10:20 – 10:40 Dataset presentation from Aneo
- 10:40 – 11:00 Dataset presentation from Equinor
- 11:00 – 11:15 Break
- 11:15 – 11:30 Dataset presentation for IDI, NTNU
- 11:30 – 11:45 Open-source dataset presentations
- 11:45 – 12:00 Report Template and Grading criteria

## Project (Group)

You will work in groups of 5–6 students.

## Low-Level Features:

- Low-level features include color histograms, edges, and textures.
- These features are straightforward to compute from pixel data.
- They provide basic information about the visual appearance of images.

## Color Histograms:

- Color histograms quantify the distribution of colors within an image.
- They are useful for distinguishing images based on dominant color schemes.
- Histograms are invariant to image size and orientation.

 [\(https://gurus.pyimagesearch.com/\)](https://gurus.pyimagesearch.com/)

[□ \(https://gurus.pyimagesearch.com/lessons/color-histograms/#\)](https://gurus.pyimagesearch.com/lessons/color-histograms/#)

### PylImageSearch Gurus Course

[□ \(https://gurus.pyimagesearch.com/\)](https://gurus.pyimagesearch.com/)

## 10.3: Color histograms

The previous lesson discussed the most basic type of image descriptor — [color channel statistics](#) (<https://gurus.pyimagesearch.com/lessons/color-channel-statistics/>). Today we are going to review a type of basic image descriptor, but one that is substantially more powerful and is pervasive across all of computer vision in some form or another. I'm talking about the histogram. Specifically, the *color histogram*.

Just as our basic color channel statistics (i.e. the mean and standard deviation) characterize the color distribution of an image, so does a color histogram. But unlike a the mean and standard deviation which attempt to summarize the pixel intensity distribution, a color histogram *explicitly represents it!* In fact, a color histogram *is* the color distribution!

We'll still be operating under the same assumption that images with similar color distributions contain equally similar visual contents — this assumption may or may not hold in your particular application. For many computer vision systems (especially for those operating under controlled lighting conditions) color histograms become an extremely valuable and powerful image descriptor.

Instead of building another example to rank a dataset of images in terms of their similarity as we did in the color channel statistics lesson, we're actually going to apply a little machine learning today. We're going to take another small dataset of images — but instead of ranking, we are going to *cluster* and *group* them into two distinct classes using color histograms.

Imagine being able to take the photos from your last family vacation and *automatically* group them into specific events and places based only on the image contents. In essence, that's exactly what we will be doing today.

## Edge Detection and Textures:

- Edge detection algorithms identify boundaries and shapes in images.
- Texture features capture surface patterns and roughness.
- Techniques like Sobel filters and Gabor filters are commonly used.

## **High-Level Features:**

- High-level features include object presence and scene context.
- They are often extracted using deep learning models.
- Such features enable more complex image understanding.

## **Deep Learning for Feature Extraction:**

- Convolutional Neural Networks (CNNs) automatically learn hierarchical features.
- Pre-trained models like VGG, ResNet, can be used for feature extraction.
- These features are highly effective for classification and similarity tasks.

## **Process of Feature Extraction:**

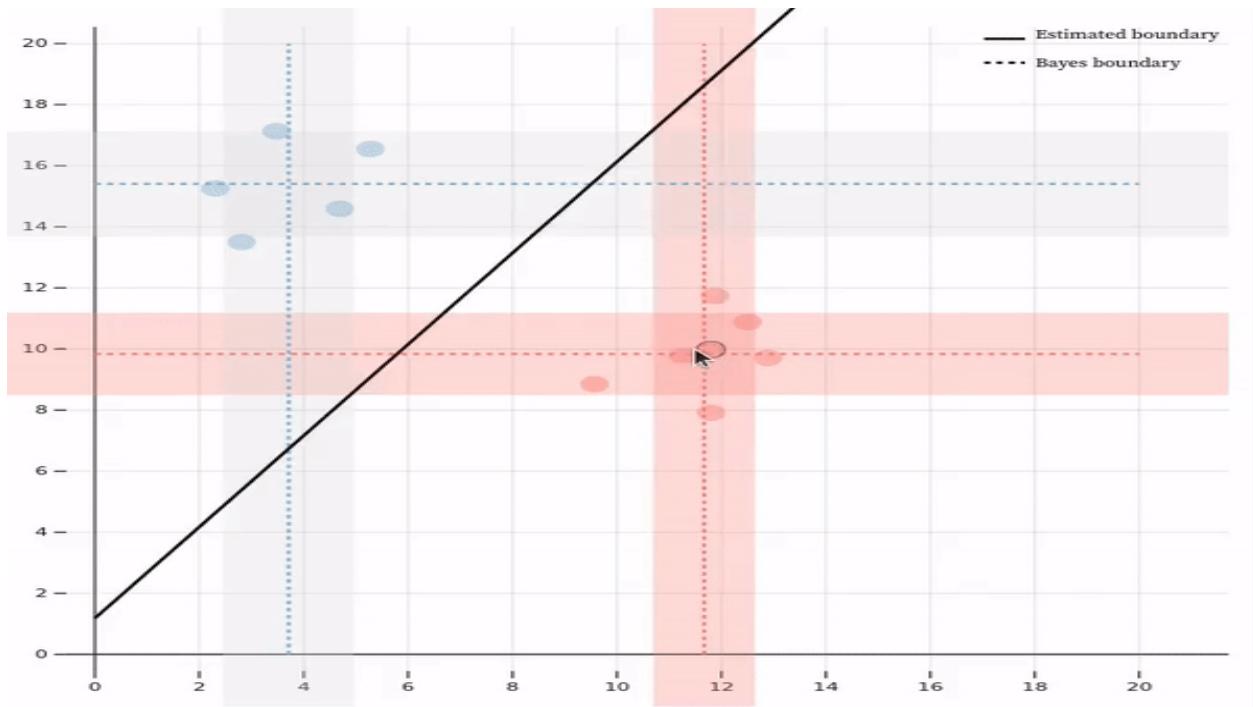
- Using CNNs Input images are processed through multiple convolutional layers.
- Each layer extracts increasingly abstract features.
- The output of the last convolutional layer serves as the feature vector.

## **Applications of Extracted:**

- Content-based image retrieval relies on feature vectors.
- Object recognition and scene classification are enhanced by features.
- Features also assist in image captioning and tagging.

## **Dimensionality Reduction Techniques:**

- Techniques like PCA and t-SNE help visualize and reduce feature vectors.
- They maintain essential information while simplifying data.
- These techniques facilitate clustering and classification.



### Challenges in Feature Extraction:

- Variability in lighting, angles, and occlusions affects features.
- High-dimensional features may require significant computational resources.
- Selecting the right features depends on the specific task and dataset.



### **Future Trends in Feature Extraction:**

- Integration of multimodal data for richer features.
- Use of unsupervised and self-supervised learning.
- Development of more efficient and interpretable feature extraction methods.

### **Conclusion:**

- Feature extraction transforms raw images into meaningful representations.
- The choice of features impacts the success of image analysis tasks.
- Ongoing advances continue to improve the effectiveness of feature extraction techniques.