1. Discuss the different elements of Image Analysis.

many applications when images are viewed in stereo, as visualization (and therefore, recognition) of targets is enhanced dramatically. Viewing objects from directly above also provides a very different perspective than what we are familiar with. Combining an unfamiliar perspective with a very different scale and lack of recognizable detail can make even the most familiar object unrecognizable in an image. Finally, we are used to seeing only the visible wavelengths, and the imaging of wavelengths outside of this window is more difficult for us to comprehend.

Recognizing targets is the key to interpretation and information extraction. Observing the differences between targets and their backgrounds involves comparing different targets based on any, or all, of the visual elements of **tone, shape, size, pattern, texture, shadow, and association**. Visual interpretation using these elements is often a part of our daily lives, whether we are conscious of it or not

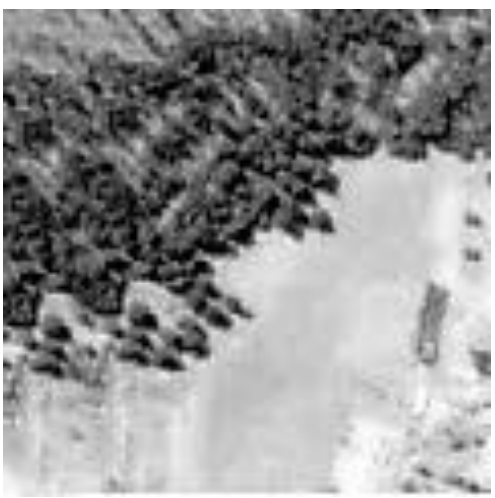
---

Tone refers to the **relative brightness or colour of objects in an image**. Generally, tone is the fundamental element for distinguishing between different targets or features. Variations in tone also allows the elements of shape, texture, and pattern of objects to be distinguished.

Shape refers to the general form, structure, or outline of individual objects. Shape can be a very distinctive clue for interpretation. Straight edge shapes typically represent urban or agricultural (field) targets, while natural features, such as forest edges, are generally more irregular in shape, except where man has created a road or clear cuts. Farm or crop land irrigated by rotating sprinkler systems would appear as circular shapes.

Size of objects in an image is a function of scale. It is important to assess the size of a target relative to other objects in a scene, as well as the absolute size, to aid in the interpretation of that target. A quick approximation of target size can direct interpretation to an appropriate result more quickly. For example, if an interpreter had to distinguish zones of land use, and had identified an area with a number of buildings in it, large buildings such as factories or warehouses would suggest commercial property, whereas small buildings would indicate residential use.

Pattern refers to the spatial arrangement of visibly discernible objects. Typically an orderly repetition of similar tones and textures will produce a distinctive and ultimately recognizable pattern. Orchards with evenly spaced trees, and urban streets with regularly spaced houses are good examples of pattern.



Texture refers to the arrangement and frequency of tonal variation in particular areas of an image. Rough textures would consist of a mottled tone where the grey levels change abruptly in a small area, whereas smooth textures would have very little tonal variation. Smooth textures are most often the result of uniform, even surfaces, such as fields, asphalt, or grasslands. A target with a rough surface and irregular structure, such as a forest canopy, results in a rough textured appearance. Texture is one of the most important elements for distinguishing features in radar imagery.

Shadow is also helpful in interpretation as it may provide an idea of the profile and relative height of a target or targets which may make identification easier. However, shadows can also reduce or eliminate interpretation in their area of influence, since targets within shadows are much less (or not at all) discernible from their surroundings. Shadow is also useful for enhancing or identifying topography and landforms, particularly in radar imagery.

Association takes into account the relationship between other recognizable objects or features in proximity to the target of interest. The identification of features that one would expect to associate with other features may provide information to facilitate identification. In the example given above, commercial properties may be associated with proximity to major transportation routes, whereas residential areas would be associated with schools, playgrounds, and sports fields. In our example, a lake is associated with boats, a marina, and adjacent recreational land.

2. Interpret the different components of Pattern recognition system with an example

better understand the process it is best to separate it into different components which will be described in the following part of the text.

- the **input**, the first component of a pattern recognition system represents the data that is being processed. This data is usually gathered via different sensors, cameras, and similar tools or obtained from a database. When the data is obtained it is in its raw form, which can be used right away, or it can be altered to suit the later steps. Most often used is a form of a vector.

- The second step is **pre-processing**. This includes manipulation of data to produce a dataset on which a specific computer algorithm can be applied. For example, when pre-processing images, you can do a re-sizing of the images so that all of the images in the data set have the same number of pixels, and apply different types of filtering such as a low pass and a high pass filter which in this case are used for noise reduction in the data. Preprocessing is done to increase the overall accuracy of the recognition and to reduce the storage requirements.

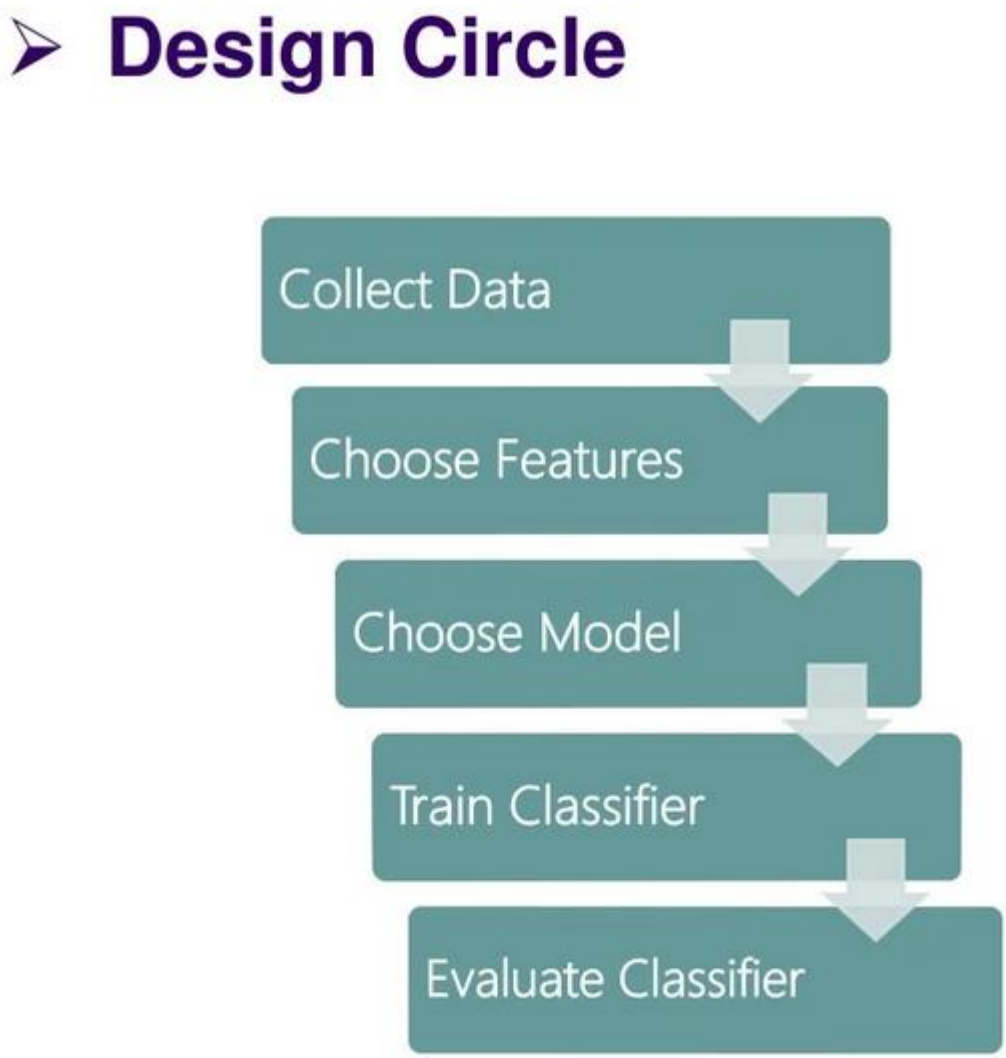
- **Dimensionality reduction** is used to reduce the number of variables in the data set by obtaining a set of principal variables. Different approaches are feature selection and feature extraction. Feature extraction derives an informative and non-redundant set of variables from an initial dataset. In image processing, this is used to detect and isolate different shapes on an image, and it can be used to isolate different letters from one another using similar principles.

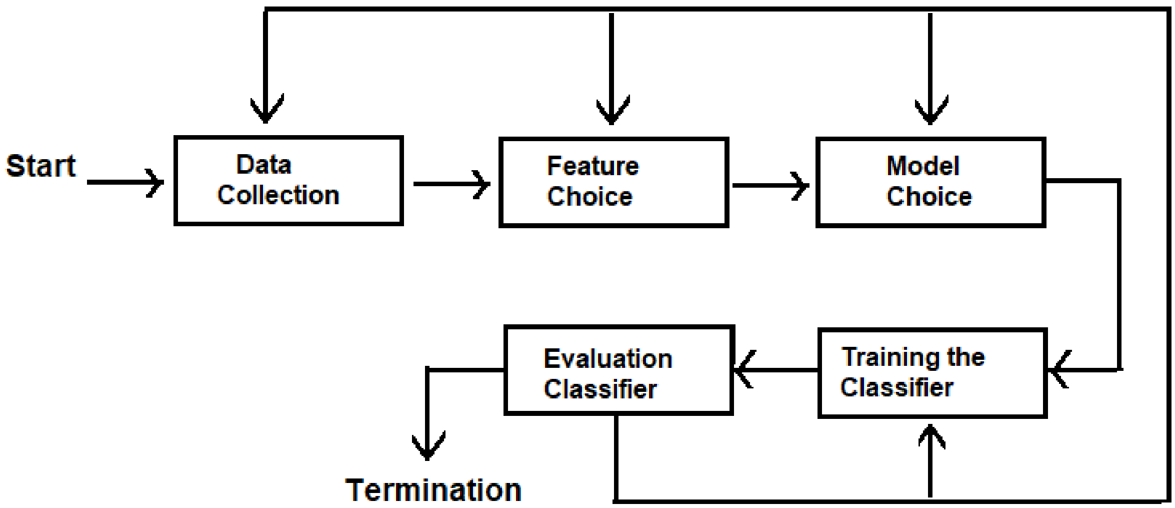
- **Prediction** is the next component of a pattern recognition system. Machine learning models are being selected in this step. You can use various different machine learning models and two of the biggest groups are supervised and unsupervised learning. Supervised learning solves classification and regression problems. Unsupervised learning is used to detect a structure within the given dataset. It uses clusterization and dimension reduction algorithms.

- **Applying different algorithms** on the dataset means the output will have several different models available to select. This is done in the next step where the best model is chosen for the data. Validation of models is done in several different ways, and the goal is to minimize the prediction error that occurs. There are methods like holdout, cross-validation, bootstrap and separation of data into training, validation and testing subsets

3. Explain the design cycle of a Pattern recognition system.

https://www.slideshare.net/AKMamun/design-cycles-of-pattern-recognition





Activity Cycle

4. Discuss the different forms of learning with examples.
https://www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_types_of_learning.htm

Supervised Learning

Supervised learning is commonly used in real world applications, such as face and speech recognition, products or movie recommendations, and sales forecasting. Supervised learning can be further classified into two types - **Regression** and **Classification**.

Regression trains on and predicts a continuous-valued response, for example predicting real estate prices.

Classification attempts to find the appropriate class label, such as analyzing positive/negative sentiment, male and female persons, benign and malignant tumors, secure and unsecure loans etc.

In supervised learning, learning data comes with description, labels, targets or desired outputs and the objective is to find a general rule that maps inputs to outputs. This kind of learning data is called **labeled data**. The learned rule is then used to label new data with unknown outputs.

Supervised learning involves building a machine learning model that is based on **labeled samples**. For example, if we build a system to estimate the price of a plot of land or a house based on various features, such as size, location, and so on, we first need to create a database and label it. We need to teach the algorithm what features correspond to what prices. Based on this data, the algorithm will learn how to calculate the price of real estate using the values of the input features.

Supervised learning deals with learning a function from available training data. Here, a learning algorithm analyzes the training data and produces a derived function that can be used for mapping new examples. There are many **supervised learning algorithms** such as Logistic Regression, Neural networks, Support Vector Machines (SVMs), and Naive Bayes classifiers.

Common **examples** of supervised learning include classifying e-mails into spam and not-spam categories, labeling webpages based on their content, and voice recognition.

Unsupervised Learning

Unsupervised learning is used to detect anomalies, outliers, such as fraud or defective equipment, or to group customers with similar behaviors for a sales campaign. It is the opposite of supervised learning. There is no labeled data here.

When learning data contains only some indications without any description or labels, it is up to the coder or to the algorithm to find the structure of the underlying data, to discover hidden patterns, or to determine how to describe the data. This kind of learning data is called **unlabeled data**.

Suppose that we have a number of data points, and we want to classify them into several groups. We may not exactly know what the criteria of classification would be. So, an unsupervised learning algorithm tries to classify the given dataset into a certain number of groups in an optimum way.

Unsupervised learning algorithms are extremely powerful tools for analyzing data and for identifying patterns and trends. They are most commonly used for clustering similar input into logical groups. Unsupervised learning algorithms include Kmeans, Random Forests, Hierarchical clustering and so on.

Semi-supervised Learning

If some learning samples are labelled, but some other are not labelled, then it is semi-supervised learning. It makes use of a large amount of **unlabelled data for training** and a small amount of **labelled data for testing**. Semi-supervised learning is applied in cases where it is expensive to acquire a fully labelled dataset while more practical to label a small subset. For example, it often requires skilled experts to label certain remote sensing images, and lots of field experiments to locate oil at a particular location, while acquiring unlabeled data is relatively easy.

Reinforcement Learning

Here learning data gives feedback so that the system adjusts to dynamic conditions in order to achieve a certain objective. The system evaluates its performance based on the feedback responses and reacts accordingly. The best known instances include self-driving cars and chess master algorithm AlphaGo.

5. Distinguish between the Feature selection and Feature extraction with an
example.
https://youtu.be/VJ83NxCrLaE

6. What are feature vectors? Explain the concept of dimensionality reduction
with respect to feature vectors.
7. Explain the concept of feature space with respect to machine learning.
https://dataorigami.net/blogs/napkin-folding/17536555-feature-space-in-machine-learning
https://youtu.be/dPDuvrkGkh8

8. Discuss the concept of clustering with an example
https://www.javatpoint.com/clustering-in-machine-learning

9. Illustrate the different distance measures used in Clustering.
https://www.analyticsvidhya.com/blog/2020/02/4-types-of-distance-metrics-in-machine-learning/

10.Briefly explain supervised and unsupervised parameter estimation with an example
https://www.geeksforgeeks.org/supervised-unsupervised-learning/

11. Briefly explain the different clustering algorithms available
https://www.geeksforgeeks.org/different-types-clustering-algorithm/#:~:text=Centroid%20based%20methods%20%3A,is%20minimum%20with%20the%20center.
https://www.javatpoint.com/data-mining-different-types-of-clustering

12. Illustrate hierarchical clustering algorithm with an example
https://www.javatpoint.com/hierarchical-clustering-in-machine-learning

13. Demonstrate k-means algorithm with an example.
https://www.javatpoint.com/k-means-clustering-algorithm-in-machine-learning

2. Discuss the pattern classification carried out using a statistical classifier.
many of the techniques we shall describe have been developed over a range of diverse disciplines, there is naturally a variety of sometimes contradictory terminology. We shall use the term 'pattern' to denote the p-dimensional data vector x = (x1, . . . , x p)T of measurements

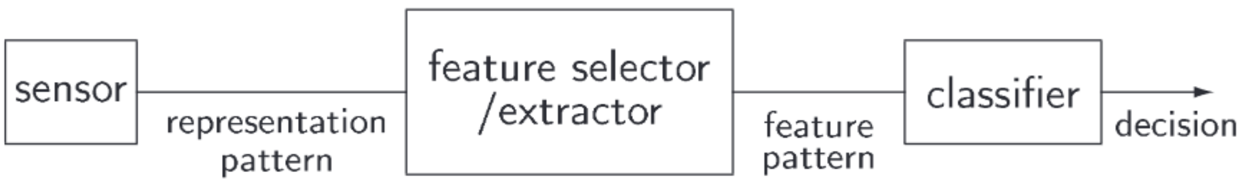


Figure 1.1 Pattern classifier.

The main topic in this book may be described by a number of terms including *pattern classifier design* or *discrimination* or *allocation rule design*. Designing the rule requires specification of the parameters of a pattern classifier, represented schematically in Figure 1.1, so that it yields the optimal (in some sense) response for a given input pattern. This response is usually an estimate of the class to which the pattern belongs. We assume that we have a set of patterns of known class $\{(\boldsymbol{x}_i, z_i), i = 1, \ldots, n\}$ (the *training* or *design* set) that we use to design the classifier (to set up its internal parameters). Once this has been done, we may estimate class membership for a pattern $\boldsymbol{x}$ for which the class label is unknown. Learning the model from a training set is the process of *induction*; applying the trained model to patterns of unknown class is the process of *deduction*.

Thus, the uses of a pattern classifier are to provide:

- A descriptive model that explains the difference between patterns of different classes in terms of features and their measurements.
- A predictive model that predicts the class of an unlabelled pattern.

1.2 Stages in a pattern recognition problem

A pattern recognition investigation may consist of several stages enumerated belo
stages may be present; some may be merged together so that the distinction be
operations may not be clear, even if both are carried out; there may be some ap
specific data processing that may not be regarded as one of the stages listed below
the points below are fairly typical.

1. Formulation of the problem: gaining a clear understanding of the aims of t
gation and planning the remaining stages.
2. Data collection: making measurements on appropriate variables and record
of the data collection procedure (ground truth).
3. Initial examination of the data: checking the data, calculating summ
producing plots in order to get a feel for the structure.
4. Feature selection or feature extraction: selecting variables from the
are appropriate for the task. These new variables may be obtaine
nonlinear transformation of the original set (feature extraction). To
partitioning of the data processing into separate feature extraction
processes is artificial, since a classifier often includes the optimisa
extraction stage as part of its design.
5. Unsupervised pattern classification or clustering. This may be view
data analysis and it may provide a successful conclusion to a study. C
it may be a means of preprocessing the data for a supervised classifi
6. Apply discrimination or regression procedures as appropriate. The cla
using a training set of exemplar patterns.
7. Assessment of results. This may involve applying the trained classif
dent *test set* of labelled patterns. Classification performance is often s
form of a confusion matrix:

3. Describe the different performance evaluation measures used for measuring a classifier.

Accuracy

The overall **accuracy** of a model is simply the number of correct predictions divided by the total number of predictions. An accuracy score will give a value between 0 and 1, a value of 1 would indicate a perfect model.

$$accuracy = CorrectPredictions/TotalPredictions$$

This metric should rarely be used in isolation, as on imbalanced data, where one class is much larger than another, the accuracy can be highly misleading.

Confusion Matrix

A **confusion matrix** is an extremely useful tool to observe in which way the model is wrong (or right!). It is a matrix that compares the number of predictions for each class that are correct and those that are incorrect.

In a confusion matrix, there are 4 numbers to pay attention to.

True positives: The number of positive observations the model correctly predicted as positive.

False-positive: The number of negative observations the model incorrectly predicted as positive.

True negative: The number of negative observations the model correctly predicted as negative.

False-negative: The number of positive observations the model incorrectly predicted as negative.

Precision

Precision measures how good the model is at correctly identifying the positive class. In other words out of all predictions for the positive class how many were actually correct? Using alone this metric for optimising a model we would be minimising the false positives. This might be desirable for our fraud detection example, but would be less useful for diagnosing cancer as we would have little understanding of positive observations that are missed.

$$precision = TP/TP + FP$$

Recall

Recall tell us how good the model is at correctly predicting **all** the positive observations in the dataset. However, it does not include information about the false positives so would be more useful in the cancer example.

$$recall = TP/TP + FN$$

F1 score

The **F1 score** is the harmonic mean of precision and recall. The F1 score will give a number between 0 and 1. If the F1 score is 1.0 this indicates perfect precision and recall. If the F1 score is 0 this means that either the precision or the recall is 0.

$$F1 = 2 \times precision \times recall/precision + recall$$

Kappa

The **kappa** statistic compares the observed accuracy to an expected accuracy or the accuracy expected from random chance. One of the flaws of pure accuracy is that if a class is imbalanced then making predictions at random could give a high accuracy score. Kappa accounts for this by comparing the model accuracy to the expected accuracy based on the number of instances in each class.