Pattern Recognition and its application to image processing

• Dr  Suresh Sundaram
  • sureshsundaram@iitg.ernet.in
Reference books

- Pattern Classification : - Duda, Hart, Stork
- Pattern recognition and Machine Learning :- Christopher Bishop
- Neural networks for Pattern Recognition :- Christopher Bishop
- Introduction to Machine Learning :- Alpaydin
Journals

- IEEE TPAMI
- Pattern Recognition
- Pattern Recognition Letters
- Pattern Analysis and Applications
- IEEE TIP

- IEEE Multimedia
- Speech Technology
• ICPR
• ICVGIP
• ICASSP
• NIPS
• ICML
• ECCV
• ACCV….
• Person identification systems -> Biometrics, Aadhar,
• How did we learn the alphabet of the English language?

Trained ourselves to recognize alphabets, so that given a new alphabet, we use our memory / intelligence in recognizing it.
Machine Perception

• How about providing such capabilities to machines to recognize alphabets?

• The field of pattern recognition exactly does that.
• Build a machine that can recognize patterns:
  – Speech recognition
  – Fingerprint identification
  – OCR (Optical Character Recognition)
  – DNA sequence identification
A basic PR framework

- Training samples
- Testing samples
- An algorithm for recognizing an unknown test sample
- Samples are labeled (supervised learning)
Typical supervised PR problem

- Alphabets – 26 in number (upper case)

- # of alphabets/ classes to recognize – 26.
- Collect samples of each of the 26 alphabets and train using an algorithm.
- Once trained, test system using unknown test sample/ alphabeth.
pattern $\rightarrow$ feature extractor $\xrightarrow{X} \text{classifier} \rightarrow$ class label

- Feature extractor makes some measurements on the input pattern.
- $X$ is called *Feature Vector*. Often, $X \in \mathbb{R}^n$.
- Classifier maps each feature vector to a class label.
- Features to be used are problem-specific.
A pattern is an entity, vaguely defined, that could be given a name, e.g.,

- fingerprint image,
- handwritten word,
- human face,
- speech signal,
- DNA sequence
- alphabet
From
Jim Elder
829 Loop Street, Apt 300
Allentown, New York 14707

To
Dr. Bob Grant
602 Queensberry Parkway
Omar, West Virginia 25638

Nov 10, 1999

We were referred to you by Xena Cohen at the University Medical Center. This is regarding my friend, Kate Zack.

It all started around six months ago while attending the "Rubeq" Jazz Concert. Organizing such an event is no picnic, and as President of the Alumni Association, a sponsor of the event, Kate was overworked. But she enjoyed her job, and did what was required of her with great zeal and enthusiasm.

However, the extra hours affected her health; halfway through the show she passed out. We rushed her to the hospital, and several questions, x-rays and blood tests later, were told it was just exhaustion.

Kate's been in very bad health since. Could you kindly take a look at the results and give us your opinion?

Thank you!
Jim
故天将降大任于是人也，必先苦其心志，劳其筋骨，饿其体肤，空乏其身，行拂乱其所为，所以动心忍性，曾益其所不能。

(a) Handwriting

(b) Corresponding Machine Print
Face recognition
Fingerprint recognition
• Object classification
• Signature verification (genuine vs forgery)
• Iris recognition
• Writer adaptation
• Speaker recognition
• Bioinformatics (gene classification)
• Communication System Design
• Medical Image processing
• Bag of algorithms that can used to provide some intelligence to a machine.

• These algorithms have a solid probabilistic framework.

• Algorithms work on certain characteristics defining a class - referred as ‘features’.
What is a feature?

- Features across classes need to be discriminative for better classification performance.
• Presence of a dot in ‘i’ can distinguish these ‘i’ from ‘l’ and is a feature.

• Features values can be discrete or continuous in nature (floating value).

• In practice, a single feature may not suffice for discrimination.
• Bag of algorithms that can used to provide some intelligence to a machine.

• These algorithms have a solid probabilistic framework.

• Algorithms work on certain characteristics defining a class - refered as ‘features’.
What is a feature?

- Features across classes need to be discriminative for better classification performance.
• Presence of a dot in ‘i’ can distinguish these ‘i’ from ‘l’ and is a feature.

• Features values can be discrete or continuous in nature (floating value).

• In practice, a single feature may not suffice for discrimination.
In practice, a single feature may not suffice for discrimination.

- A possible solution is to look out for many features and select a set (possibly with feature selection algorithms). The goal is to improve the recognition performance of unseen test data.

- The different features selected can be represented with a vector called as ‘feature vector’.
• Suppose we select d features, we can represent them with a d-dimensional feature vector.

• Pixels of an image of size M X N can be represented with a MN*1 dimensional feature vector.
Feature selection

• Domain Knowledge helps in extracting features

• Feature discriminability measures are available like Fisher scores to measure the effectiveness of features.
List of features used in literature

- Pixels in an image
- Edge based features in an image
- Transformed coefficients

DFT (Shape description)
DCT (Compression)
Wavelets (Palm print recognition)
KLT /PCA (Face recognition)
Gabor (Texture classification, script identification)
MFCCs (Speech systems)
Features

• Feature to be discriminative
• Specific to applications…… no universal feature for all pattern recognition problems …. Ugly Duckling Theorem

• To be robust to translation, rotation, occlusion, scaling
• Continuous, real valued
• Discrete
• Binary
• Mixed
• Features depend on the problem. Measure ‘relevant’ quantities.

• Some techniques available to extract ‘more relevant’ quantities from the initial measurements. (e.g., PCA)

• After feature extraction each pattern is a vector

• Classifier is a function to map such vectors into class labels.

• Many general techniques of classifier design are available.

• Need to test and validate the final system.
Curse of dimensionality

- If limited data is available, too many features may degrade the performance. We need as large number of training samples for better generalization to beat the `curse of dimensionality`!

- Need arises to come up with techniques such as PCA to pick the `relevant features`.
“Sorting incoming Fish on a conveyor according to species using optical sensing”

- Species
  - Sea bass
  - Salmon
• Problem Analysis

– Set up a camera and take some sample images to extract features

• Length
• Lightness
• Width
• Number and shape of fins
• Position of the mouth, etc…

• This is the set of all suggested features to explore for use in our classifier!
• Preprocessing
  
  – Use a segmentation operation to isolate fishes from one another and from the background

• Information from a single fish is sent to a feature extractor whose purpose is to reduce the data by measuring certain features

• The features are passed to a classifier
• Classification

  – Select the length of the fish as a possible feature for discrimination
The length is a poor feature alone!

Select the lightness as a possible feature.
• Adopt the lightness and add the width of the fish as a new feature

Fish \[ \mathbf{x}^T = [x_1, x_2] \]

Lightness Width
• We might add other features that are not correlated with the ones we already have. A precaution should be taken not to reduce the performance by adding such “noisy features”

• Ideally, the best decision boundary should be the one which provides an optimal performance such as in the following figure:
Use simple models to complicated ones: Occam’s razor
• Sensing
  – Use of a transducer (camera or microphone)

• Segmentation and grouping
  – Patterns should be well separated and should not overlap
• Feature extraction
  – Discriminative features
  – Invariant features with respect to translation, rotation and scale.

• Classification
  – Use a feature vector provided by a feature extractor to assign the object to a category

• Post Processing
  – Exploit context input dependent information other than from the target pattern itself to improve performance
The Design Cycle

- Data collection
- Feature Choice
- Model Choice
- Training
- Evaluation
- Computational Complexity
• Data Collection

– How do we know when we have collected an adequately large and representative set of examples for training and testing the system?
• Feature Choice

– Depends on the characteristics of the problem domain. Simple to extract, invariant to irrelevant transformation insensitive to noise.
• Model Choice

  – Unsatisfied with the performance of our fish classifier and want to jump to another class of model
• Training

  – Use data to determine the classifier. Many different procedures for training classifiers and choosing models
• Evaluation

  – Measure the error rate (or performance and switch from one set of features to another one
• Computational Complexity

  – What is the trade-off between computational ease and performance?

  – (How an algorithm scales as a function of the number of features, patterns or categories?)
Learning paradigms

• Supervised learning
  – A teacher provides a category label or cost for each pattern in the training set

• Unsupervised learning
  – The system forms clusters or “natural groupings” of the input patterns
Unsupervised Learning

• The system forms clusters or “natural groupings” of the input patterns….

• Clustering is often called an un­supervised learning task as no class values denoting an a priori grouping of the data instances are given
Segmentation of an image into $k$ clusters by a popular iterative algorithm called $k$-Means Algorithm.

Original image

Segmented image using $k$ Means Clustering (k=3)
• **Reinforcement learning** is an area of machine learning inspired by behaviorist psychology, concerned with how software agents ought to take *actions* in an *environment* so as to maximize some notion of cumulative *reward*. 
Semi-supervised learning is a class of supervised learning tasks and techniques that also make use of unlabeled data for training - typically a small amount of labeled data with a large amount of unlabeled data.

It falls between unsupervised learning (without any labeled training data) and supervised learning (with completely labeled training data).
• Implement the k nearest neighbor using MATLAB

• Implement k means clustering on an image
K Means algorithm
Algorithm 1 (K-means clustering)

1 begin initialize \( n, c, \mu_1, \mu_2, \ldots, \mu_c \)
2 do classify \( n \) samples according to nearest \( \mu_i \)
3 recompute \( \mu_i \)
4 until no change in \( \mu_i \)
5 return \( \mu_1, \mu_2, \ldots, \mu_c \)
6 end
K means
K Means algorithm
K means
K means
k means
K means
k means
k means
k means
Figure 9.3 Two examples of the application of the $K$-means clustering algorithm to image segmentation showing the initial images together with their $K$-means segmentations obtained using various values of $K$. This also illustrates the use of vector quantization for data compression, in which smaller values of $K$ give higher compression at the expense of poorer image quality.
Example of Nearest Neighbor Rule

- Two class problem: yellow triangles and blue squares. Circle represents the unknown sample $x$ and as its nearest neighbor comes from class $\theta_1$, it is labeled as class $\theta_1$. 
Lab Session
• Implement k means algorithm on the text image given to you.....here we segment the text from background

the quick brown fox
jumps over the lazy dog.

THE QUICK BROWN FOX
JUMPS OVER THE LAZY DOG.
clc
clear all
close all
a = imread('C:\Users\dell\Documents\MATLAB\PhD\ADMAT_Demos\input_text.jpg');
imshow(255-rgb2gray(a));

b = reshape(double(255-rgb2gray(a)),[],1);
[idx, c] = kmeans(b, 2, 'emptyaction', 'singleton');

Fin_a = reshape(idx, [size(rgb2gray(a))]);
[ind1, ind2] = find(Fin_a(:, :) == 2);
[ind3, ind4] = find(Fin_a(:, :) == 1);

figure();
imshow(255-rgb2gray(a));
• Implement the nearest neighbour classifier to distinguish digit 1 from 8.
• 50 Training samples of digit 1 in NNClass1
• 50 Training samples of digit 8 in NNClass2 folder

• 10 samples in NNTest folder…we need to categorize each sample into either 1 or 8.

• Explanation of code by TA
Thank You