Optimized Image Classification Based on Universal Image Distance and Support Vector Machines

Nandita chasta[1], Neha Solanki[2]
[1]Asst. Prof, Aravali institute of technical studies, Udaipur
[2]Asst. Prof, Jaipur engineering college and research center, Jaipur

Abstract- *Image Classification of remotely sensed images is one of the most important field of research in computer engineering. Image classification techniques are being used in object recognition, quality control and OCR systems. Many of the machine vision systems used in industrial applications employ well known image processing algorithms to discriminate between good and bad parts. Algorithms such as thresholding, blob analysis and edge detection, for example, can be found in every machine vision software vendor's toolbox since they can be used in numerous applications to solve a relatively large number of imaging tasks. Image classification may be performed using supervised, unsupervised or semi-supervised learning techniques. In supervised learning, the system is presented with numerous examples of images that must be manually labeled. Using this training data, a learned model is then generated and used to predict the features of unknown images. Such traditional supervised learning techniques can use either generative or discriminative models to perform this task. In this dissertation, UID techniques are used in an optimized manner to represent an image in the form of a vector in finite dimensions. The distance between this representation and that of a prototype image is computed to find the similarity score between the images. This mating score can be used to train any machine learning system under supervised or unsupervised environment. In this dissertation, an SVM based classifier is trained using feature vectors to train a classifier in a supervised environment. The precision and accuracy of the machine is computed over the benchmark techniques of image classification. The overall performance of the proposed methods is evaluated using R simulator in terms of precision, recall and kappa measure. Simulation results establish the validity and efficiency of the approach.*

Keywords- Universal Image Distance, LZ Complexity, Machine Learning, Support Vector Machines

I. INTRODUCTION

Image classification is one of the most focused problem in the modern era of digital image processing and machine learning. In image classification, an image is classified according to its visual content. For example, does it contain an airplane or not. An important application is image retrieval - searching through an image dataset to obtain (or retrieve) those images with particular visual content. Classification includes a broad range of decision-theoretic approaches to the identification of images. All classification algorithms are based on the assumption that the image in question depicts one or more features, for e.g., geometric parts in the case of a manufacturing classification system, or spectral regions in the case of remote sensing, and that each of these features belongs to one of several distinct and exclusive classes [1]. The classes may be specified a priori by an analyst (as in supervised classification) or automatically clustered (i.e. as in unsupervised classification) into sets of prototype classes, where the analyst merely specifies the number of desired categories.

Image classification analyzes the numerical properties of various image features and organizes data into categories. Classification algorithms typically employ two phases of processing: training and testing. In the initial training phase, characteristic properties of typical image features are isolated and, based on these, a unique description of each classification category, i.e. training class, is created. In the subsequent testing phase, these feature-space partitions are used to classify image features.

The description of training classes is an extremely important component of the classification process. In supervised classification,statisticalprocesses(i.e.basedonanapriori knowledge of probability distribution functions) or distribution - free processes can be used to extract class descriptors [2]. Unsupervised classification relies on clustering algorithms to automatically segment the training data into prototype classes. In either case, the motivating criteria for constructing training classes is that they are:

1. Independent, means. a change in the description of one training class should not change the value of another,
2. Discriminatory, i.e. different image features should have significantly different descriptions, and
3. Reliable, all image features within a training group should share the common definitive descriptions of that group.

A convenient way of building a parametric description of this sort is via a feature vector , where n is the number of attributes which describe each image feature and training class. This representation allows us to consider each image feature as occupying a point, and each training class as occupying a sub- space (i.e. a representative point surrounded by some spread, or deviation), within then-dimensional classification space. Viewed as such, the classification problem is that of determining to which sub-space class each feature vector belongs.

Problem Statement

The problem of image classification into any of the pre-specified classes in considered in this dissertation. This classification is performed using the Lempl-Ziv complexity [3] and thereby, the Universal Image Distance measure [4] between the given images.

This technique is used to represent images in terms of feature vectors of prototype images in which the prototypes belongs to the selected image categories. These feature vector images are them manually classified to create a trading set to train an SVM based classifier [5]. The classifier is then tested for images belonging to unspecified categories and precision and accuracy of the classification is measured.

Research Approach

The given image is represented into binary string of 1s and 0s. This is done by converting each pixel to its corresponding grayscale value and then converting it to corresponding 8 bit binary string. Another image is created corresponding to the image under consideration by taking a sliding window and taking the averages and making it slide over the image matrix from top left to bottom right. This image is the average value image for the given image. This image is converted to corresponding pixel values in binary form. The Lempl- Ziv complexity of the binary string is measure. The distance between two images is computed in terms of Universal Image distance which is measured in terms of LZ complexity. Finally the matrix of distances of prototype images is considered and then clustered using K Means technique. This clustering is then used to express any given image in terms of feature vectors which are distances from the prototype categories. Finally an SVM is trained over the manually classified set and then tested over the images to check the precision and accuracy.

II. LITERATURE REVIEW

Digital image classification is a process of defining pixels to classes. It is possible to groups the similar pixels into classes that are associated with the informational categories of interest to users. This process can be implemented by comparing the pixels or group of pixels to some prototypes. These prototypes forms segments on map or an image, or some other informational classes, so that after the classification process, the digital image can be represented in the form of uniform parcels, each identified by color or symbol. Image classification is the most significant part of the digital image analysis. In some cases, classification itself may be the object of analysis. Image classification is an important tool for examination of digital images as well as to produce the final output. The term classifier refers to the computer program that implements a specific procedure for image classification. The purpose of the classification process is to categorize the digital image into any one of categories which are provided by the classifier. The simplest way to classify the image is the Multispectral data [9]. It can be used to perform the classification. The spectral pattern in a pixel or a group of pixels is compared under various techniques and methods, thereby being used as the features or numerical indices for categorization. The main objective of the image classification is to identify as a unique color level, the objects occurring in an image in terms of the pixel groups or feature vectors. As an alternative, more complex classification processes consider groups of pixels within their spatial setting within the image as means of using the textual information. These are spatial classifiers which examine the small areas within the image using both spectral and the textual information to classify the image. Spatial classifiers are more crucial to program and much more expensive to use as compared to point classifiers [10]. Spatial classifiers are more accurate and classification can be done without any prior training requirement to the classifier.

Image classification has made great progress over the past decades in the following three areas: (1) development and use of advanced classification algorithms, such as subpixel, per-field, and knowledge- based classification algorithms; (2) use of multiple remote-sensing features, including spectral, spatial, multi-temporal, and multi- sensor information; and (3) incorporation of ancillary data into classification procedures, including such data as topography, soil, road, and census data. Accuracy assessment is an integral part in an image classification procedure. Accuracy assessment based on error matrix is the most commonly employed approach for evaluating per-pixel classification, while fuzzy approaches are gaining attention for assessing fuzzy classification results. Spectral features are the most important information for image classification. As spatial resolution increases, texture or context information becomes another important attribute to be considered. Classification approaches may vary with different types of remote- sensing data. The success of an image classification depends on

many factors. The availability of high-quality remotely sensed imagery and ancillary data, the design of a proper classification procedure, and the analyst’s skills and experiences are the most important ones. For a particular study, it is often difficult to identify the best classifier due to the lack of a guideline for selection and the availability of suitable classification algorithms to hand.

III. PROPOSED WORK

Lempel-Ziv Complexity

There are several complexity measures to test the randomness of a sequence. Linear complexity computation is one of these measures. Lempel Ziv complexity of a sequence was defined by Lempel and Ziv in 1976. This measure counts the number of different patterns in a sequence when scanned from left to right. There are many variations of Lempel Ziv around, but they all follow the same basic idea. This basic idea is to parse the sequence into distinct phrases. For instance Lempel-Ziv complexity of s = 101001010010111110 is 8, because when scanned from left to right, different patterns observed in s are 1|0|10|01|010|0101|11|110|.

The formal definition of LM complexity and factorization of strings can be given in a mathematical way.

Let P, Q and R be the strings defined over some alphabet A. For any string S, l(S) denotes the length of the string and S(i) denotes the ith element of the string. Also S(i,j) denotes the substring of S which consists of characters of S between the positions i and j, both inclusive. If l(S) =N and j>N, then the substring is terminated at the last character of the string. Also, if i>j, then the result is an empty string.

An extension R of P, denoted by P→R, is reproducible from P if R = PQ if there exists an integer p, such that p < l(P) and Q(k) = R(p-1+k) for k = 1,2,. l(Q).

For example, consider the production aacgt→aacgtcgtcg

Here, P = aacgt, R = aacgtcgtcg and Q = cgtcg. Also,

Q(1) = c = R(3-1+1)

Q(2) = g = R(3-1+2)

Q(5) = g = R(3-1+5)

Thus, the specified production rules becomes valid for p=3.

R is obtained from P (the seed) by first copying all of S and then copying in a sequential manner l(Q) elements starting at the pth location of S in order to obtain the Q part of R.

A string S is producible from its prefix S(1, j) (denoted S(1, j) ⇒ R), if S(1, j) → S(1, l(S) − 1). For example, aacgt → aacgtac and aacgt → aacgtacc both with pointers p = 2. The production adds an extra different character at the end of the copying process.

An m-step production process of S results in parsing of S in which H(S) = S(1, h1) ・S(h1+1, h2) ・ ・ ・ S(hm−1+1, hm) is called the history of S and Hi(S) = S(hi−1+1, hi) is called the ith component of H(S). For example for S = aacgtacc, the history is H(S) = a ・ac ・ g ・ t ・ acc as the history of S.

Stating in the other way, let u = u1u2 uN, where symbols are

drawn from a finite alphabet Σ of cardinality σ(= |Σ|). Let u(i, j) be the substring uiui+1 uj taken from u (u(i, j) ⊂ u) and of length j-i+1.

Let the π operator is defined as:

u(i, j)π = u(i, j − 1) and, consequently, u(i, j)πk = u(i, j − k)

The Lempel-Ziv factorization E(u) of the string u is defined as: E(u) = u(1, h1)u(h1 + 1, h2). u(hm−1 + 1, N)

Thus, the string is expressed in the form of m factors is such, that each factor u(hk−1 + 1, hk) complies with

the sliding window formed by the pixel values of the pixels enclosed under the frame as shown:

1. $u(h_{k-1} + 1, h_k)\pi \subset u(1, h_k)\pi^2$
2. $u(h_{k-1}+1, h_k) \not\subset u(1, h_k)\pi$ except, perhaps, for the last factor $u(h_{m-1}+ 1, N)$

Let us denote by cH(S) the number of components in a history of S. The LZ complexity of S is c(S) = min{cH(S)} where the minimum is over all histories of S. It can be shown that c(S) = cE(S) where cE(S) is the number of components in the exhaustive history of S.

A distance for strings based on the LZ-complexity is defined as follows:

Given two strings X and Y , denote by XY their concatenation, then define

d(X, Y) : = max{c(XY) − c(X), c(YX) − c(Y)}

It is proved by that the following normalized formula performs well in the classification and clustering of strings.

Universal Image Distance

The Universal Image Distance (UID) gives the basic idea to convert the images into a string of characters. Let there be two color images X and Y having equal dimensions. The conversion process is as follows:

1. Given the two color images X and Y in JPG format, extract the RGB values of the individual pixels and take the average and replace the pixel with the corresponding values. This gives the grayscale image corresponding to the color image. Thus, convert both the images to corresponding grayscale images. The numeric value of each of the pixels of the images lies in the range [0,255]. Thus, each of the image can be thought of as a string over the alphabet [0,255].
2. A hypothetical frame is considered, as a sub-image of the image, as shown. This sub-image is formed as a result of

X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X	X	X	X	X	X

Fig. 1: Sub Image Window

A new image can be formed by sliding the window one pixel to the right or lowering it one pixel down as shown in the following figures:
The following figure shows one pixel left shift operation.

X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X	X	X	X	X	X

Fig. 2: Sub Image window, one pixel left movement operation
The following figure shows one pixel down shift operation.

X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X	X	X	X	X	X

Fig. 3: Sub Image Window, One Pixel Down Operation

3. These sets of pixels, belonging to the window, are mapped to an alphabet value in the range [0,255] through Quantization Index Modulation.
4. A string corresponding to the image is thus obtained by scanning the image from left to right and going downwards, moving one pixel at a time, to reach from the top leftmost pixel to the right bottommost pixel. The quantized values of the sliding window gives the characters of the string. The same string sequence is obtained for another prototype string under consideration.
5. With two images being represented as strings, one can get the similarity score between the two images using LZ complexity.

Illustration of the LZ complexity

Consider a hypothetical image matrix as shown in the figure.

Table 1: Hypothetical Image Matrix

244	221	213	223	228	230	231	228	225	244	220	236	220	217	214
207	217	220	224	249	238	229	245	235	242	226	236	249	214	250
205	249	210	208	224	200	220	236	240	225	239	232	201	203	248
221	223	200	218	230	247	237	240	240	219	241	242	216	205	250
207	238	225	231	223	246	229	233	200	233	227	212	209	219	244
201	221	236	227	224	201	231	238	239	203	222	229	245	209	204
247	215	243	239	209	232	211	233	229	233	213	232	210	219	201
207	224	244	205	211	240	237	230	233	228	205	238	233	226	213
248	250	200	224	239	219	212	238	246	234	247	225	213	215	215
201	220	202	249	213	229	202	219	246	225	225	248	235	241	213
237	206	204	223	202	202	216	205	239	238	218	240	212	203	228
213	235	222	242	239	229	216	226	210	206	219	244	246	207	202
213	240	234	200	210	210	209	210	236	246	230	246	217	221	238
244	212	230	202	234	246	223	249	238	234	204	206	234	232	234
207	213	214	235	235	208	204	231	231	219	232	205	201	203	236

The pictorial representation of the above image in terms of grayscale color palette is as shown in fig. 4:

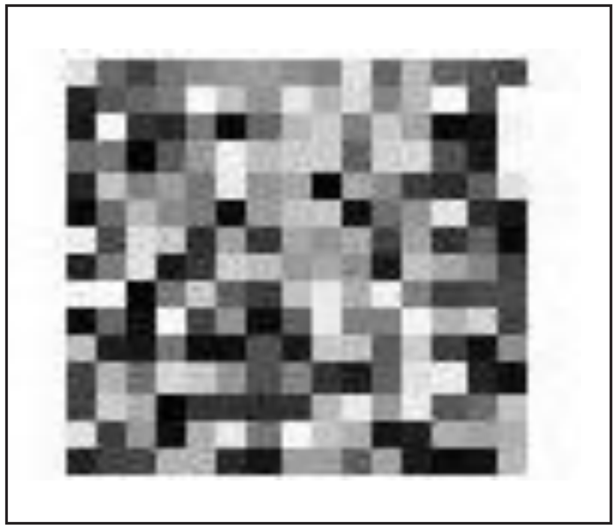
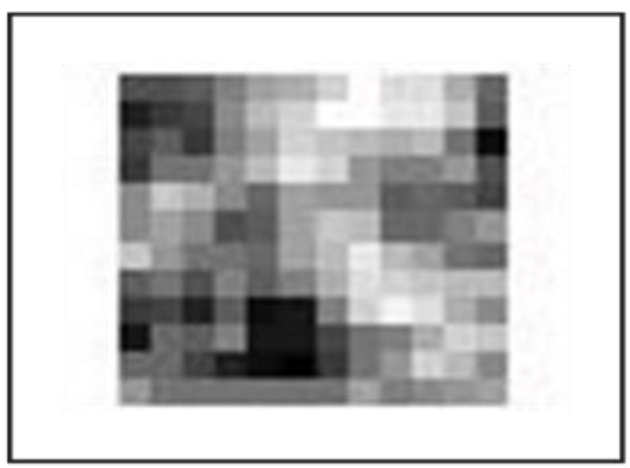


Fig. 4: Grayscale Image Corresponding to the Image Matrix

The matrix corresponding to the moving window of size 3X3, and taking averages thereof, is shown in table 2.

Table 2: Averages of Matrix 3, Using Overlapping Slinding Window of 3X3

221	221	222	225	228	229	232	236	233	233	229	223
217	219	220	226	230	232	236	236	234	234	231	222
220	222	219	225	228	232	231	230	229	230	224	215
219	224	224	227	230	234	232	227	225	225	227	221
226	231	229	226	223	228	227	227	222	223	222	220
226	228	226	221	222	228	231	230	223	223	225	227
231	227	224	224	223	228	230	234	230	228	224	223
222	224	221	225	222	225	229	233	232	231	230	230
219	220	217	222	215	216	225	232	235	233	229	226
216	223	222	225	216	216	220	224	225	229	232	231
223	223	220	217	215	214	219	224	227	232	230	226
227	224	224	224	224	224	224	228	225	226	227	228



The 8 bit binary values corresponding to each of the numeric pixel values, followed by the aggregation in row-wise manner yields the string as shown:

String1 =
[1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 1 0 1 1 1 0 0 0 0 1 1 1
1 0 0 1 0 0 1 1 1 0 0 1 0 1 1 1 1 0 1 0 0 0 1 1 1 0 1 1 0 0 1 1 1 0 1
0 0 1 1 1 1 0 1 0 0 1 1 1 1 0 0 1 0 1 1 1 0 1 1 1 1 1 1 1 1 0 1 1 0 0 1
1 1 0 1 1 0 1 1 1 1 0 1 1 1 0 0 1 1 1 0 0 0 1 0 1 1 1 0 0 1 1 0 1 1 1
0 1 0 0 0 1 1 1 0 1 1 0 0 1 1 1 0 1 1 0 0 1 1 1 0 1 0 1 0 1 1 1 0 1
0 1 0 1 1 1 0 0 1 1 1 1 1 0 1 1 1 1 0 1 1 0 1 1 1 0 0 1 1 0 1 1 1 1
0 1 1 0 1 1 0 1 1 1 1 1 0 0 0 0 1 1 1 1 0 0 1 0 0 1 1 1 0 1 0 0 0 1
1 1 0 0 1 1 1 1 1 1 0 0 1 1 0 1 1 1 0 0 1 0 1 1 1 1 0 0 1 1 0 1 1 1
0 0 0 0 0 1 1 0 1 0 1 1 1 1 1 0 1 1 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0
0 0 0 1 1 1 0 0 0 1 1 1 1 1 0 0 1 1 0 1 1 1 0 1 0 1 0 1 1 1 0 1 0 0
0 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 1 1 1
1 0 1 1 1 0 1 1 1 1 0 0 0 1 0 1 1 1 0 0 1 1 1 1 1 1 0 0 1 0 1 1 1 1
0 0 0 1 0 1 1 0 1 1 1 1 1 1 1 1 0 0 1 0 0 1 1 1 0 0 0 1 1 1 1 1 0 0
0 1 1 1 1 0 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 0 1 1 1 1 0 1 1 0 1 1 1 0
0 1 1 1 0 0 0 1 0 1 1 1 0 0 1 0 0 1 1 1 0 0 0 1 0 1 1 0 1 1 1 0 1 1
1 0 1 1 1 1 0 1 1 1 0 0 1 0 0 1 1 1 0 0 1 1 1 1 1 1 0 0 1 1 0 1 1 0
1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 0 0 0 1 1 1 1 0 0 0 1 1 1 1 1 0 0
1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 1 1 0 1 1 1 1
1 1 1 1 0 0 1 0 0 1 1 1 0 0 1 1 0 1 1 1 0 1 0 1 0 1 1 1 0 0 1 1 0 1
1 1 0 0 1 0 0 1 1 1 0 0 0 0 0 1 1 0 1 1 1 1 1 1 1 0 1 1 1 1 0 1 1 1
0 0 0 0 0 1 1 0 1 1 1 0 1 1 1 0 0 0 0 1 1 1 0 1 1 1 1 0 1 1 1 0 0
0 0 1 1 1 1 0 0 1 0 1 1 1 1 0 1 0 0 1 1 1 1 0 1 0 0 0 1 1 1 0 0 1 1
1 1 1 1 0 0 1 1 0 1 1 1 0 0 1 1 0 1 1 0 1 1 0 1 1 1 1 0 1 1 1 0 0 1
1 0 1 1 0 0 1 1 1 0 1 1 1 1 0 1 1 0 1 0 1 1 1 1 1 0 1 1 0 0 0 1 1 1
0 0 0 0 1 1 1 1 0 1 0 0 0 1 1 1 0 1 0 1 1 1 1 1 0 1 0 0 1 1 1 1 0 0
1 0 1 1 1 1 0 0 0 1 0 1 1 0 1 1 0 0 0 1 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1
0 1 1 1 0 0 0 0 1 1 1 0 1 1 0 0 0 1 1 0 1 1 0 0 0 1 1 0 1 1 1 0 0 1
1 1 0 0 0 0 0 1 1 1 0 0 0 0 1 1 1 1 0 0 1 0 1 1 1 1 0 1 0 0 0 1 1 1
0 0 1 1 1 1 1 0 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 0 1 1 1 0 0 1 1 0 1 1
0 0 1 1 1 0 1 0 1 1 1 1 1 0 1 0 1 1 0 1 1 0 1 1 0 1 1 1 1 1 0 0 0 0
0 1 1 1 0 0 0 1 1 1 1 1 0 1 0 0 0 1 1 1 0 0 1 1 0 1 1 1 0 0 0 1 0 1
1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 1 1 1
0 0 0 0 0 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 1 1 1 0 0 1 0 0 1 1 1 0 0
0 0 1 1 1 1 0 0 0 1 0 1 1 1 0 0 0 1 1 1 1 1 0 0 1 0 0]

The Lempel Ziv complexity of the string can be computed using the techniques described in section 3.1 above. The complexity of the given string is 280. This means that 280 different substrings are required for the process of production of this string.

Table 3: Hypothetical Image Matrix II

212	214	212	226	236	236	222	212	223	218	226	238	222	226	223
233	223	244	216	224	219	200	203	221	201	247	228	250	219	220
214	247	210	216	224	203	215	207	245	249	220	250	226	221	233
212	216	213	245	237	244	238	206	214	216	230	236	246	200	239
222	249	244	225	202	241	227	208	202	228	246	231	241	217	212
241	217	227	244	208	217	240	244	206	217	209	222	202	206	246
210	224	232	205	222	239	242	206	219	210	246	233	224	226	244
250	230	238	227	210	228	207	205	242	227	231	239	239	201	220
216	231	246	242	221	233	207	240	216	217	211	240	218	235	212
234	209	240	239	243	235	209	233	213	240	216	231	221	217	241
215	216	232	241	237	227	226	219	213	206	207	225	234	224	246
204	218	210	221	200	246	229	245	238	246	248	228	221	249	225
223	246	247	233	215	215	222	213	217	240	239	210	207	229	244
231	230	245	202	245	210	215	204	225	211	237	216	215	202	247
222	238	212	227	222	234	213	240	238	225	210	242	244	248	248

The grayscale image corresponding to the image matrix is as shown in the following figure:

Fig. 6: Grayscale Image Corresponding to the Image Matrix 3

The image matrix corresponding to the sliding window average of the image matrix is as shown in Table 4.

Table 4: Averages of Matrix 3, Using Overlapping Slinding Window of 3X3

223	223	223	222	220	213	216	220	228	231	234	231
224	226	225	225	223	215	217	218	227	231	237	231
225	229	224	226	226	221	218	219	228	234	236	230
227	231	227	229	228	229	221	216	219	226	229	222
230	230	223	223	226	229	222	216	220	227	228	222
230	227	224	222	224	225	223	220	223	226	227	221
231	231	227	225	223	223	220	220	224	228	231	228
233	234	234	231	221	222	219	226	224	228	227	227
227	233	238	235	226	225	220	222	215	221	223	227
220	225	229	232	228	230	225	228	225	227	226	228
223	229	226	226	224	227	225	226	228	228	224	225
228	228	224	221	222	222	223	227	233	231	225	220

The grayscale image corresponding to the image matrix is as shown in the following figure:

The grayscale image corresponding to the image matrix is as shown in the following figure:

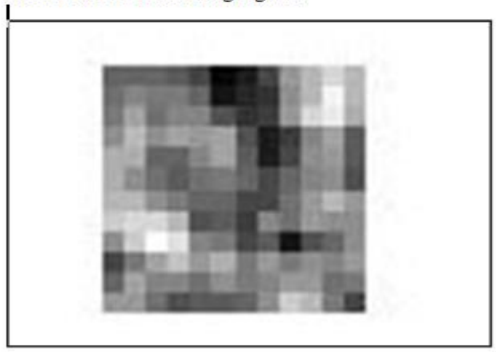


Fig. 7: Grayscale Image Corresponding to the Averaging of Pixel Values of Matrix.

The 8 bit binary values corresponding to each of the numeric pixel values, followed by the aggregation in row-wise manner yields the string as shown:

String2 =
[1 1 0 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 0 1 1 1 1 0 1 1
0 1 1 1 0 0 1 1 0 1 0 1 0 1 1 1 0 1 1 0 0 0 1 1 0 1 1 1 0 0 1 1 1 0 0
1 0 0 1 1 1 0 0 1 1 1 1 1 1 0 1 0 1 0 1 1 1 0 0 1 1 1 1 1 1 0 0 0 0 0
1 1 1 0 0 0 1 0 1 1 1 0 0 0 0 1 1 1 1 0 0 0 0 1 1 1 0 1 1 1 1 1 1 1 0
1 0 1 1 1 1 1 0 1 1 0 0 1 1 1 0 1 1 0 1 0 1 1 1 0 0 0 1 1 1 1 1 0 0
1 1 1 1 1 1 0 1 1 0 1 1 1 1 0 0 1 1 1 1 1 1 0 0 0 0 1 1 1 1 0 0 1 0
1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 1 0 1 1 1 0 0 0 1 0 1 1 0 1 1 1 0 1 1
1 0 1 1 0 1 0 1 1 0 1 1 0 1 1 1 1 1 0 0 1 0 0 1 1 1 0 1 0 1 0 1 1 1
0 1 1 0 0 1 1 1 0 0 1 1 0 1 1 1 0 0 0 1 1 1 1 1 0 0 1 1 1 1 1 1 0 0
0 1 1 1 1 1 0 0 1 0 1 1 1 1 0 0 1 0 0 1 1 1 0 0 1 0 1 1 1 0 1 1 1 0
1 1 1 0 1 1 0 0 0 1 1 0 1 1 0 1 1 1 1 1 0 0 0 1 0 1 1 1 0 0 1 0 1 1
1 0 1 1 1 1 0 1 1 1 0 0 1 1 0 1 1 1 0 0 1 1 0 1 1 0 1 1 1 1 1 1 1 0
1 1 1 1 1 1 1 1 0 0 0 1 0 1 1 1 0 0 1 0 1 1 1 0 1 1 1 1 0 1 1 0 1 1
0 0 0 1 1 0 1 1 1 0 0 1 1 1 0 0 0 1 1 1 1 1 0 0 1 0 0 1 1 0 1 1 1 1
0 1 1 1 0 0 1 1 0 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 0 1 1 1 1 0 1
1 1 0 0 0 0 0 1 1 1 0 0 0 0 1 1 1 0 1 1 1 1 1 1 1 0 1 1 1 0 0 1 1 0
1 1 1 1 1 1 1 1 0 0 0 1 0 1 1 1 0 0 0 1 1 1 1 0 1 1 1 0 1 1 1 1 0 0
1 1 1 1 1 1 0 0 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 0 1 1 1 1
1 1 1 0 1 1 1 1 1 1 1 0 1 1 1 0 0 1 1 0 1 1 1 0 0 1 1 1 0 0 0 0 0 1
1 1 0 0 1 0 0 1 1 1 0 0 1 1 1 1 1 1 0 0 1 0 0 1 1 1 0 1 0 0 1 1 1 1
0 1 0 1 0 1 1 1 0 1 0 1 0 1 1 1 0 0 1 1 1 1 1 0 1 1 1 0 1 1 1 0 1 1
1 1 0 1 1 0 1 1 0 1 1 1 1 1 0 0 0 1 0 1 1 1 0 0 0 0 0 1 1 1 0 0 1 0
0 1 1 1 0 0 0 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 1 1 1 1 1 0 1 0 0 1 1
1 1 0 1 1 1 0 1 1 1 0 1 0 1 1 1 1 1 0 0 0 1 0 1 1 1 0 0 0 0 1 1 1 0
1 1 1 0 0 1 1 0 1 1 1 1 0 1 1 0 1 0 1 1 1 1 1 0 1 1 1 0 1 1 1 0 1 1
1 1 1 1 1 1 0 0 0 1 1 1 1 0 1 1 1 0 0 1 1 1 0 0 0 0 1 1 1 1 0 0 1 0
1 1 1 1 0 1 0 0 0 1 1 1 0 0 1 0 0 1 1 1 0 0 1 1 0 1 1 1 0 0 0 0 1 1
1 1 0 0 1 0 0 1 1 1 0 0 0 0 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 1 0 1 1 1
0 0 1 0 0 1 1 0 1 1 1 1 1 1 1 1 1 0 0 1 0 1 1 1 1 0 0 0 1 0 1 1 1 0 0
0 1 0 1 1 1 0 0 0 0 0 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1 0 0 0 1
0 1 1 1 0 0 1 0 0 1 1 1 0 0 1 0 0 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 1 1
1 1 0 0 1 0 0 1 1 1 0 0 1 0 0 1 1 1 0 0 0 0 0 1 1 0 1 1 1 0 1 1 1 0
1 1 1 1 0 1 1 0 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 1
0 0 1 1 1 1 0 0 1 1 1 1 1 1 0 0 0 0 1 1 1 0 1 1 1 0 0]

The LZ complexity of the given string is 300. This means that 300 different substrings can be used to produce the string str2.

Let String 3 denotes the concatenation of string 1 and string 2. The complexity of String3 comes out to be 521.

Let String 4 denotes the concatenation of string2 and String 1. The complexity of string 3 comes out to be 534.

Using formula described in the previous sections, the distance between the two strings comes out to be:

$d(X, Y) := \max\{c(XY) - c(X), c(YX) - c(Y)\}$
$d(X, Y) = \max\{521\text{-}280, 534\text{-}300\}$

$= \max\{241,234\} = 241$

The above distance count calculates the distances between two images in terms of LZ complexity.

Proposed Model

The proposed model for SVM based classifier for image classification is graphically illustrated as shown in figure 8.

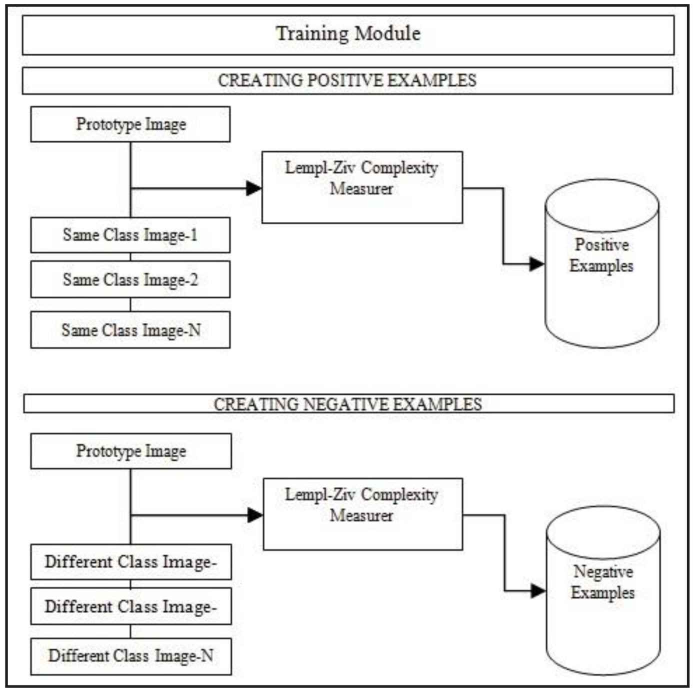


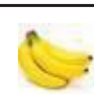
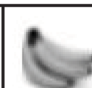
Fig. 8 Model for Image Classification

These positive and negative examples are used to train the SVM for image classification for any incoming image.

IV. ANALYSIS OF PROPOSED WORK

A. Lempl-Ziv Complexity measurement of Prototype Image A prototype image belonging to certain category is considered and is converted to binary string according to the process described. The LZ complexity of the string is then computed using standard algorithm. Due to the limitation in the processing capabilities in view of the combinatorial explosion with the increased string length, images of size 40X40 pixels are considered in the simulation studies.

Table 5: Lempl-Ziv Complexity Values For Different Prototype Images Under Different Window Sizes

S. No.	Prototype Image (Banana/Apple)	Grayscale Image	Window Size	Pixel Window Average based Image	Lempel-Ziv Complexity Values
1			3		2178
2			4		2099

3			5		1999
4			3		2244
5			4		2255
6			5		2074
7			3		2289
8			4		2217
9			5		2102
10			3		2003
11			4		2106
12			5		2046

It turns out that the LZ complexity values of the binary representation of the images generally decreases as the images are more and more averaged over its overlapping segments.

LZ Distance Computation between Image strings

The LZ distance measure between the images and the corresponding strings is shown in Table 6. The computation is performed by first converting the image from color to grayscale, followed by pixel- window wise approximation and then tabulation the complexity score. The complexity score of two strings is then reformulated to find out the Lempl-Ziv distance between the two images.

Table 6: Lempl-Ziv Distance Values For Different Prototype Images Under Different Window Sizes

Image 1	Image 2	Window Size	Image 1 Complexity (String 1) C1	Image 2 Complexity (String 1) C2	Complexity C3 = String1. String2	Complexity C4 = String2. String1	Distance
		3	1146	1077	2236	2169	1092
		3	1696	1654	3352	3328	1674
		4	1202	1136	2331	2308	1172
		4	1774	1713	3399	3452	1739
		3	1146	1654	2732	2800	1586
		4	1202	1713	2817	2923	1615
		5	1293	1764	2985	3098	1692

Image Representation as Feature Vector

As stated previously, for the purpose of image classification, each image is represented by a finite dimensional feature vector whose components are the UID values between the image and a finite set of image prototypes from each of the feature categories.

Prototype of an image refers to the sub-image of the given image which is placed by placing a window over the image in such a way that the sub image is completely encompassed by the image under consideration.

The images are initially manually classified into feature categories. The following algorithm specifies how the images can be represented as feature vectors:

Algorithm for Prototype Clustering

1. Let there be M feature categories. Also, let there be N images (N>M) relevant to the feature categories.
2. For each of the set of images, the prototypes of various sizes are considered. Let there be P prototypes. A distance matrix of size PXP is computed between all the

prototypes. As these prototypes are unlabelled, the matrix contains all the distances between the prototypes which belongs to the same or different feature categories.

3. The matrix thus obtained is operated through clustering techniques to obtain the clusters consisting of various prototypes.

4. If there are M clusters each consisting of images belonging to the same feature category, then the process is successful. The algorithm is terminated, otherwise repeat the same process starting from step 2.

Thus, the clustering of the prototypes into relevant categories indicates that the prototypes are representatives of their respective categories. This algorithm is time consuming in view of the excess computations involved in the creation of the matrix of distances and to repeat the algorithm if the clustering algorithm fails.

This algorithm is to be used with suitable clustering techniques

Proposed Algorithm for representation of images as a feature vector and supervised Learning

Consider an Image I which is to be represented as a vector over M feature categories. The algorithm for this vector conversion proceed as follows:

1. Consider a prototype of a feature category. Let it be of size MXN. Obtain all the overlapping sub-images of the given image I, starting from the top left to the bottom right, of same size as that of the prototype image.
2. Compute the average values of all such distances of all the prototypes belonging to the feature category. Let this distance be R1.
3. Repeat steps 1 and 2 for all the feature categories.
4. Normalize the distances using the following formulae:

$$r_i = \frac{R_i}{\sum_{i=1}^{M} R_i}$$

5. Represent the image as normalized vector of the feature categories.
6. After obtain the representation of all the images in the image corpus, obtain a subset of representative image and label them to the image categories, or the feature categories.
7. Obtain the feature vector representation of all the images belonging to particular feature vector category.
8. Obtain the feature vector representation of some other images which belongs to some different feature category or some other image which does not belong to any of the category.
9. The set identified in point 8 is called the set of positive examples and the set obtained in 8 constitute the set of negative examples.

10. Mark features vectors of positive examples as true and others as false. These examples can be used to train the SVM.

Test the Test-Set over the machine to check the precision and accuracy. If the precision and accuracy of the classifier are below acceptance level, then increase the training set until desired threshold values for accuracy and precision are met.

Simulation For Selection of Prototype images

The prototype images from the three feature categories are shown in Table 4.2 given. There are a total of 60 images, taken from google-map images in three feature categories; viz city, landscape and sea, with each feature category having 20 images in the sequence order.

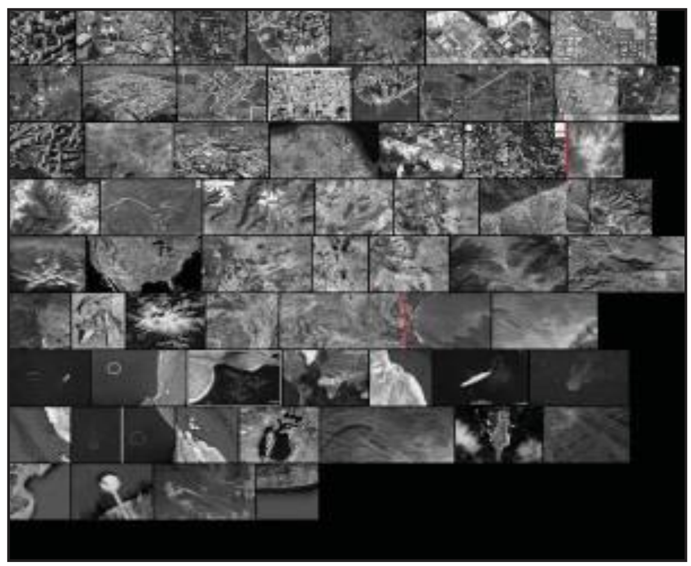


Fig. 9: Prototype Images from Google Maps Under the Categories, City, Landscapes and Sea

The red lines in the figure shows the border, starting from the top left row-wise, the city category images are pasted till the first red line, followed by the images of the landscapes which are finally followed by sea images. These images are in the corpus with each image having the name in the form ‘n.jpg’ where n is the numeric value indicating the index of the file in the order as specified in the above figure row-wise.

The randomly chosen images from the set of 60 images are shown through the bar graph where the X axis shows the index of the file chosen and the vertical axis shows the name of the files. The following plot shows the selected files with the names viz; 49, 54, 8, 55, 38, 6, 17, 33, 57, 58.

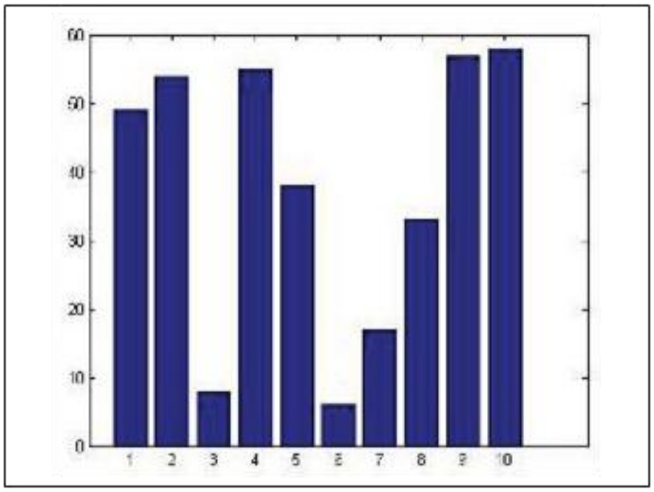


Fig. 10: Selection of Random Samples of Prototype Images

The image matrix of 10 random images from this set and the corresponding distance matrix are as shown:

Table 7: Lz Complexity Matrix For Prototype Images Selected As Shown In fig. 10

Image Index→ ↓	49	54	8	55	38	6	17	33	57	58
49	3	1032	1709	1908	2040	2246	1737	1816	898	1551
54	1032	4	1584	1788	1922	2102	1660	1687	1110	1442
8	1709	1584	4	1936	2001	2103	1849	1850	1761	1872
55	1908	1788	1936	7	2095	2195	1952	1949	1927	1910
38	2040	1922	2001	2095	5	2271	2101	2109	2061	2109
6	2246	2102	2103	2195	2271	4	2078	2069	2176	2209
17	1737	1660	1849	1952	2101	2078	5	1792	1783	1869
33	1816	1687	1850	1949	2109	2069	1792	7	1710	1894
57	898	1110	1761	1927	2061	2176	1783	1710	5	1524
58	1551	1442	1872	1910	2109	2209	1869	1894	1524	5

The K-Means clustering on the above dataset with three classes yields the values of Table 8.

Table 8: K Means Clustering Over Table 7

Image Id	Predicted Class	True Class
49	2	Sea
54	2	Sea
8	3	City
55	3	Sea
38	3	Landscape
6	1	City
17	3	City
33	3	Landscape
57	2	Sea
58	2	Sea

The accuracy of the detection are given in Table 9.

Table 9: Accuracy Of Table 8

Class	Id's	Classification
1	6	True
2	49,54,57,58	True
3	8,55,38,17,33	False (3)

The K-Means clustering on the above dataset with two classes yields Table 10.

Table 10: Clustering Using K Means With Two Classes

Image Id	Predicted Class (K Means)	True Class	Clustering Results
49	1	Sea	True
54	1	Sea	True
8	2	City (Not Sea)	True
55	2	Sea	False
38	2	Landscape (Not Sea)	True
6	2	City (Not Sea)	True
17	2	City (Not Sea)	True
33	2	Landscape (Not Sea)	True
57	1	Sea	True
58	1	Sea	True

The above results indicate that the clustering algorithm fairly accurately clusters the images which belong to one category with those that belongs to other category. In the above results, out of the 10 prototype images, 9 belongs the identified category, with only one being misclassified.
Consider again the random selection of 10 prototype images from the set of 60 images as shown in fig. 10

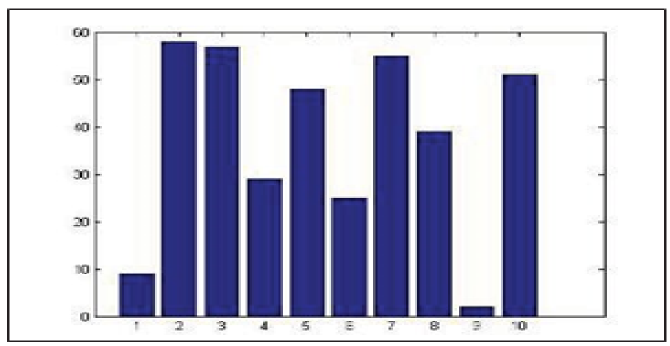


Fig. 10: Selection of Random Samples of Prototype Images

These random images corresponds to the image numbers 9, 58, 57, 29, 48, 25, 55, 39, 2 and 51.
The image matrix of these 10 random images from this set and the corresponding distance matrix are as shown:

Table 11: Lz Complexity Matrix For Prototype Images Selected As Shown In fig. 4.4

Image Index→ ↓	9	58	57	29	48	25	55	39	2	51
9	7	1876	1725	1878	1783	1862	1943	1643	1933	1765
58	1876	5	1524	1927	1518	1995	1910	1786	2080	1525
57	1725	1524	5	1834	1122	1856	1927	1677	1899	936
29	1878	1927	1834	5	1820	2034	1990	1821	1907	1783
48	1783	1518	1122	1820	4	1920	1784	1687	1983	1123
25	1862	1995	1856	2034	1920	5	2032	1759	1968	1902
55	1943	1910	1927	1990	1784	2032	7	1934	2116	1921
39	1643	1786	1677	1821	1687	1759	1934	6	1857	1641
2	1933	2080	1899	1907	1983	1968	2116	1857	5	1973
51	1765	1525	936	1783	1123	1902	1921	1641	1973	6

The K-Means clustering on the above dataset with three classes yields the values of Table 12.

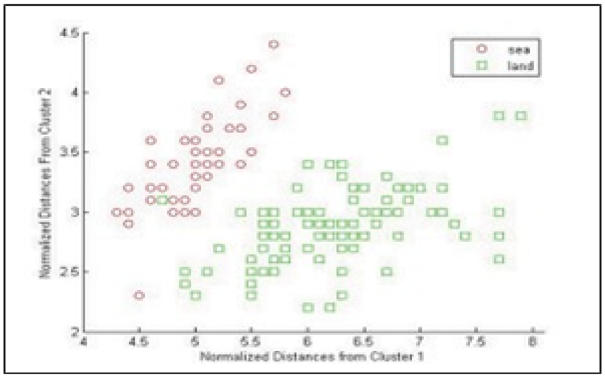


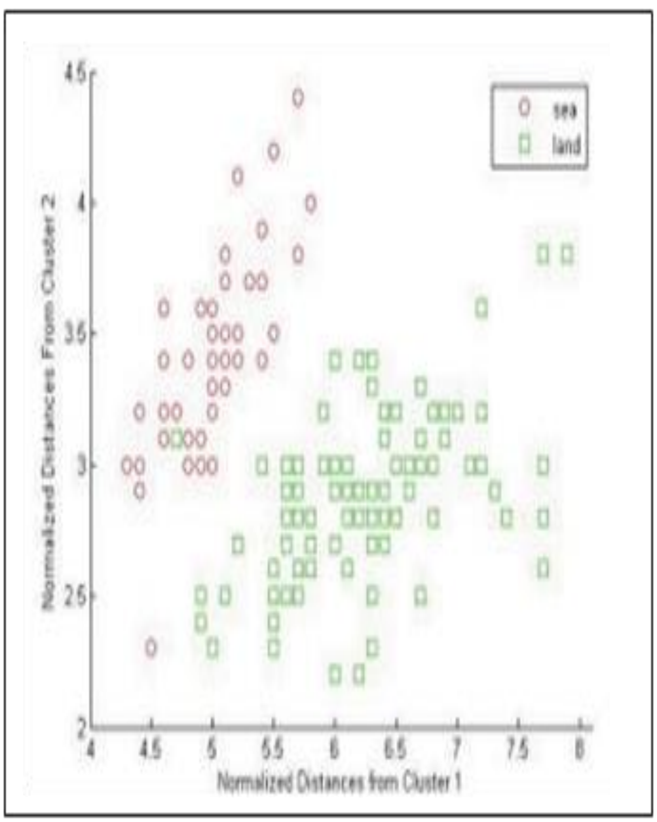
Table 12: K Means Clustering Over Table 11

Image Id	Predicted Class	True Class
9	1	City
58	2	Sea
57	2	Sea
29	3	Landscape
48	2	Sea
25	1	Landscape
55	1	Sea
39	1	Landscape
2	3	City
51	2	Sea

Table 13: Accuracy of Table 11

Class	Id's	Classification
1	9,25,55,39	False (3)
2	48,51,57,58	True
3	2,29	False (2)

The K-Means clustering on the above dataset with two classes yields Table 12.



the online food ordering system is one of the latest servicers fastest foodrestaurants in the western world are adopting. With this method, food is orderedonline and delivered to the customer. This is made possible through the use ofelectronic payment system. Customers pay with their credit cards, although creditcard customers can be served even before they make payment either through casher cheque. So, the system designed in this project will enable customers go online and place order for their food. Due to the great increase in the

Table 12: K Means Clustering Over Table 11

Image Id	Predicted Class	True Class
9	1	City
58	2	Sea
57	2	Sea
29	3	Landscape
48	2	Sea
25	1	Landscape
55	1	Sea
39	1	Landscape
2	3	City
51	2	Sea

Table 13: Accuracy of Table 11

Class	Id's	Classification
1	9,25,55,39	False (3)
2	48,51,57,58	True
3	2,29	False (2)

The K-Means clustering on the above dataset with two classes yields Table 12.

Table 14: Clustering Using K Means With Two Classes

Image Id	Predicted Class (K Means)	True Class	Clustering Results
9	1	City (Not Sea)	True
58	2	Sea	True
57	2	Sea	True
29	1	Landscape (Not Sea)	True
48	2	Sea	True
25	1	Landscape (Not Sea)	True
55	1	Sea	True
39	1	Landscape (Not Sea)	True
2	1	City (Not Sea)	True
51	2	Sea	True

Thus, the prototype images that can be considered for the simulation will be image number 9, 58, 57, 29, 48, 25, 55, 39, 2 and 51. Each of the image considered later is to be classified into one of the two classes, viz; 'sea' or 'not-sea' using an SVM classifier trained over an image set represented as distances from these prototype images.

The SVM Classifier for 150 images from Google Map data are classified for being in the category, sea or land, and classified as shown in figure

Fig 11 SVM Classification result of 150 images, with the classifier trained over the dataset of 60 images

Although the classification results shows that 2 records out of a total of 150 records are mis-classified using the SVM model, still this techniques proves to be fairly accurate for a wide range of binary classification applications. Thus, a classification can be done using the machine learning classifier over the data set of distances of the image corpus from those of the prototype images.

Result Analysis

The accuracy of the results retrieved above can be made on the basis of Precision and Recall. These two terms in context of information retrieval can be described in the following way: Given a total of N images are provided as input to a Binary Classifier trained on a particular set, let the number of images classified by the classifier as of class A be NA, and those of set be NB. Clearly

$$N = N_A + N_B$$

Also, let the total number of images which actually belongs to class A, among those of NA is Na.

The precision and the recall can now be described as follows:

$$Precision = \frac{Number\ of\ Relevant\ Items\ Retrieved}{Number\ of\ Retrieved\ Items} = \frac{N_a}{N_A}$$

Thus

$$Precision = \frac{tp}{tp + fp}$$

Here, tp refers to true positive, fp refers to false positive and fn refers to false negative under standard definitions.

The Recall is the fraction of relevant documents retrieved from among all the documents that actually belongs to the category of particular class.

$$Recall = \frac{Number\ of\ Relevant\ Items\ Retrieved}{Number\ of\ documents\ which\ belongs\ to\ the\ class}$$

Thus

$$Recall = \frac{tp}{tp + fn}$$

The experimental results obtained on a set of 150 records, is tabulated in table 4.1 depicted below:

$$Precision = \frac{48}{48 + 1} = 0.979$$

$$Recall = \frac{48}{48 + 2} = 0.96$$

Thus a fairly good precision and recall is obtained by SVM based classifier on the image data in terms of distance vectors from prototype images.

V. CONCLUSION AND FUTURE SCOPE

Conclusion

A method for automatically defining and measuring features of colored images is used for image classification. This method is suitably modified for optimization to provide fairly good accuracy using an SVM based classifier. The method is based on a universal image distance that is measured by computing the complexity of the string-representation of the two images and their concatenation. An image is represented by a feature-vector which consists of the distances from the image to a fixed set of small image prototypes, defined once by a user. There is no need for any sophisticated mathematical-based image analysis or pre-processing since the universal image distance regards the image as a string of symbols which contains all the relevant information of the image. The method proposed is time consuming for uni-processor or single core CPU but can be suitably modified to work efficiently in real time for multiprocessor systems. An SVM based classifier is also proposed and trained over the data set of images taken from google maps. The results show that standard machine learning algorithms perform well based on the proposed feature-vector representation of the images.

Future Scope

As a future scope of this research, the proposed algorithm is modified to inculcate the features that makes it efficient to run using multicourse computers. The LZ complexity measure for binary strings is complex time consuming task for long strings. Moreover, for computing the distance between two strings, the LZ complexity of the concatenation of the string must also be considered making it much more difficult to solve in real time in uni-processor systems. Also, for each of the category of the image corpus under consideration, the optimal window size must be computed to maximize the likelihood of clustering in the prototype selection phase.

REFERENCES

[1] Gong, P., Howarth, P.J.,"Frequency-based contextual classification and gray-level vector reduction for land-use identification", Photogrammetric Engineering and Remote Sensing, 58, pp. 423–437.

[2] Ju, J., Kolaczyk, E.D., Gopal, S.,"Gaussian Mixture Discriminentanalysisandsub-pixellandcovercharacterization in remote sensing", Remote Sensing of Environment, 84, pp. 550–560, 2003.

[3] Gilbert, E.N.; Kadota, T.,"The Lempel-Ziv algorithm and message complexity," In Information Theory, IEEE Transactions on , Vol. 38, No. 6, pp. 1839-1842, Nov 1992

[4] Chester, U.A.; Ratsaby, J.,"Universal distance measure for images," In Electrical & Electronics Engineers in Israel (IEEEI), 2012 IEEE 27th Convention of, pp. 1-4, 14-17 Nov. 2012.

[5] Gui, J.; Liu, T.; Tao, D.; Sun, Z.; Tan, T.,"Representative Vector Machines: A Unified Framework for Classical Classifiers," in Cybernetics, IEEE Transactions on, vol. pp. no. 99, pp. 1-1

[6] Kar, S.A.; Kelkar, V.V.,"Classification of Multispectral satellite images," In Advances in Technology and Engineering (ICATE), 2013 International Conference on, pp. 1-6, 23-25 Jan. 2013.

[7] GONG, P.,"Integrated analysis of spatial data from multiple sources: Using evidential reasoning and artificial neural network techniques for geological mapping", Photogrammetric Engineering and Remote Sensing, 62, pp. 513–523, 1996.

[8] Xiaobing Li; Yunhao Chen; Yunxia Zhang,"A comparison study on landscape characteristics based on the aggregation of remotely sensed data with multiple spatial resolutions", In Info-tech and Info-net, 2001. Proceedings. ICII 2001 - Beijing. 2001 International Conferences on, Vol. 1, pp. 186- 194, 2001.

[9] Xiaobing Li; Yunhao Chen; Yunxia Zhang,"A comparison study on landscape characteristics based on the aggregation of remotely sensed data with multiple spatial resolutions," In Info-tech and Info-net, 2001. Proceedings. ICII 2001 - Beijing. 2001 International Conferences on, Vol. 1, pp. 186- 194, 2001.

[10] Yong-Su Seo; Kuhn-Il Lee,"Category classification of multispectral image data using spatial information in the small image region," In Geoscience and Remote Sensing Symposium, 1993. IGARSS '93. Better Understanding of Earth Environment., International, pp. 1978-1980, Vol. 4, 18-21 Aug 1993.

[11] Geun-Won Yoon; Jeong-Ho Park; Kyoung-Ho Choi, "Land- cover supervised classification using user-oriented feature database," In Geoscience and Remote Sensing Symposium, 2004. IGARSS '04. Proceedings. 2004 IEEE International, Vol. 4, pp. 2724-2726, 20-24 Sept. 2004.

[12] Ying Lin; Yun Yang,"A Multiple Level Set Model for Multispectral Image Unsupervised Classification," In Education Technology and Training, 2008. International Workshop on Geoscience and Remote Sensing. ETT and GRS 2008. International Workshop on, Vol. 2, pp. 73-76, 21-22 Dec. 2008

[13] Wai Yeung Yan; Shaker, A.; Weibao Zou,"Panchromatic IKONOS image classification using wavelet based features," In Science and Technology for Humanity (TIC-STH), 2009 IEEE Toronto International Conference, pp. 456-461, 26-27 Sept. 2009.

[14] Gevers, T.; Smeulders, A.W.M.,"Image indexing using composite color and shape invariant features," In Computer Vision, 1998. Sixth International Conference on, pp. 576-581, 4-7 Jan 1998.