Computer Aided Identification of Flags using Color Features

Avishikta Lodh  
School of Education Technology  
Jadavpur University  
Kolkata, India

Ranjan Parekh  
School of Education Technology  
Jadavpur University  
Kolkata, India

ABSTRACT
The current work proposes a methodology for recognition of flags of the countries of the world from their digital images. Statistical features extracted from the color channels are combined together to generate the feature vector for discrimination between the flags. Since colors and their layouts can be quite similar between various flags, identifying them reliably can be challenging if the number of images increases. The proposed approach partitions the images into non-overlapping cells and generates a vector for each cell. The features are subsequently fed to a K-NN classifier for class discrimination. A dataset involving 400 flag images is used to test performance of the system.

General Terms
Image processing, color based image classification.

Keywords
Flag recognition, color features, k-NN classifier.

1. INTRODUCTION
Image processing technology is one of the most rapidly growing technologies today, with its application in different aspects of Remote Sensing, Intelligent Transportation System, Moving Object Tracking, Defense Surveillance, Bio-medical imaging techniques, Automatic Visual Inspection System. Computer Vision is another emerging field which is closely related to image processing. It aims at automating tasks that the human eyes can achieve. Image processing [1] is among various methods that computer vision makes use of, to achieve its goals.

Flag recognition of different countries is a difficult and troublesome task. In an attempt to simplify this, an automatic flag recognition system based on color features has been proposed. The task is challenging as many of the countries have very similar and sometimes identical colors at similar locations. The problem is compounded by the fact that images taken from different sources vary in color shades, horizontal and vertical measurements of color patches and even the physical dimensions of the flags. So reliable identification of country flags is possible if all these variations are taken into consideration and proper adjustments are incorporated into the system design.

A number of features, data modeling techniques and classifiers have been proposed for flag classification. Rahman, Z. et. al. [2] proposed a method to identify flags based on Support Vector Machine (SVM) where the machine is trained with percentage of different colors in the flag. After image acquisition and pre-processing they compute a database that consists of color percentage of 9 different colors in the flag and an additional information which is the label. They divide the RGB color gamut in 9 different basic colors (White, Red, Green, Blue, Orange, Yellow, Magenta, Cyan and Black).

Once the color percentage database is calculated they train a Support Vector Machine with 193 plain flag images, from which they used more that 40 flag images to test their system. According to the claims, their experimental results suggest more that 95% accuracy on different flags and fail to identify 5% images with complex or multiple objects.

Jetley, S. et. al. [3] propose a generalized identification method that would not only work on flags but also other insignia and object recognition tasks. They take advantage of the discriminative information that is spread across the spatial distribution of colors. Their work introduces an improved Micro-Structure Descriptor (MSD) based approach that incorporates revised criteria for HSV based color binning. They reinforce their approach with gradient analysis to achieve and accuracy of 99.2% on the training set and 76.4% on the test set.

Hart et. al. [4] proposed an interactive flag identification system that makes use of CBIR technique. They have used HSV color space and a four-layer Fuzzy-Neural algorithm for feature extraction that leads to an accuracy of 85% to 90% from a set of 186 flags, depending on whether the correct flag is among the first 10 or 20 top choices respectively.

This paper proposes a scheme for flag recognition. The organization of the paper is as follows: section 2 outlines the proposed approach with discussions on feature computation and classification schemes, section 3 provides details of the dataset and experimental results obtained, section 4 compares the proposed approach vis-à-vis other contemporary approaches, while section 5 brings up the overall conclusions and scopes for future research.

2. PROPOSED APPROACH
A block diagram depicting major functional blocks and data flow pathways of the proposed approach is shown in Fig. 1.

Fig 1: Block diagram of proposed approach

The flag images are considered to be synthetic images generated using image editing / graphics packages and hence it is assumed that they are devoid of noise. To reduce computational overloads each image is resized to standard dimensions as part of a pre-processing step. Also if any of the images contain white backgrounds, these are removed so that only the flag portions are present.
In the next step the image is partitioned into non-overlapping rectangular cells in order to localize the color distribution. An image of dimensions $M$ rows by $N$ columns is vertically divided into $H$ cells and vertically into $V$ cells such that the dimensions of each cell is $m$ by $n$ given by,

$$m = \frac{M}{H}, n = \frac{N}{V} \quad (1)$$

The actual value of $H$ and $V$ are chosen based on experimental results. For each cell $i$ the color information is split into three primary channels $R, G, B$. For each channel the mean and standard deviation is computed, and all six statistical features are combined together to model a cell information.

$$C_i = \{\mu(R), \sigma(R), \mu(G), \sigma(G), \mu(B), \sigma(B)\} \quad (2)$$

An entire flag image is represented by a collection of all such cells,

$$I = i = 1H * V C_i \quad (3)$$

The color feature of each flag image thus contains $6HV$ elements.

For classification of the flags a set of training images are used to train a $k$-NN classifier. Each class is mapped to its country name represented as a text string. During the testing phase, a set of similar but not identical images are fed to the classifier to study the performance of the system. The predicted class is compared with the ground truth to assess the recognition accuracy. The K-Nearest Neighbor algorithm is a classification algorithm that we are using for its simplicity, reliability and speed. Assuming there is some data $(x_1, y_1) .. (x_n, y_n)$, where $x_i \in \mathbb{R}^d$, $y_i$ being a $d$ dimensional real space, and given a classification problem, $y \in \{1, .. , N\}$, a K-Nearest Neighbor algorithm classifies a new point $x$ according to a majority vote of the $K$ nearest neighbors or $K$ nearest data points.

As compared to other classification algorithms K-Nearest Neighbor is easy to implement and reliable over a vast majority of classification problems. For our method we primarily relied on the K-Nearest Neighbor algorithm although we tested other classification techniques as well.

### 3. EXPERIMENTATIONS AND RESULTS

#### 3.1 Dataset

A dataset consisting of 400 images is used to test the performance of the proposed system. This consists of two sets of 200 country flags downloaded from different sources viz. [11,12,13]. The images are in RGB color and stored in PNG format. The corresponding country names along with a 2-letter mnemonic is given below. The mnemonics are henceforth used to represent the corresponding country names.

Algeria(aa), Andorra(ad), UAE(ae), Afghanistan(af), Antigua & Barbuda(ag), Angola(ao), Argentina(ar), Austria(at), Australia(au), Azerbaijan(az), Bosnia & Herzegovina.ba, Barbados(bb), Bangladesh(bd), Belgium(be), Burkina Faso(bs), Bulgaria(bg), Bahrain(bh), Benin(bj), Bermuda(bm), Brunei(br), Bolivia(bv), Brazil(br), Bahamas(bs), Bhutan(bt), Botswana(bw), Belarus(by), Belize(bz), Canada(ca), Democratic Republic of the Congo(cc), The Central African Republic(cf), Congo(cg), Sweden(ch), Côte d’Ivoire(ci), Cook Islands(ck), Chile(ch), Cameroon(cm), China(cn), Colombia(co), Costa Rica(cr), Croatia(ct), Cuba(cu), Cape Verde(cv), Christmas Island Union(cx), Cyprus.cy, Czech Republic(cz), Germany(de), Djibouti(dj), Denmark(dk), Dominica(dm), The Dominican Republic(do), Ecuador(ec), Estonia(ee), Egypt(eg), Western Sahara(esh), Eritrea(er), Ethiopia(et), Finland(fi), Fiji(fj), France(fr), Gabon(ga), Iceland(ig), Grenada(gd), Georgia(ge), Gambia(gb), Gambiat(gm), Guinea(gn), Equatorial Guinea(gq), Greece(gr), Guatemala(gt), Guinea-Bissau(gw), Guyana(gy), Hong Kong(hk), Honduras(hn), Haiti(ht), Hungary(hu), Indonesia(id), Ireland-ie), Israel(il), India(in), Iraq(iq), Iran(ir), Iceland(is), Italy(it), Jamaica(jm), Jordan(jo), Japan(jp), Kenya(ke), Kyrgyzstan(kg), Cambodia(kh), Kiribati(ki), Comoros(km), Saint Kitts & Nevis(kn), Kosovo(ko), North Korea(kp), South Korea(kr), Kuwait(kw), Kazakhstan(kz), Laos(la), Lebanon(lb), Saint Lucia(lc), Lesotho(lo), Liechtenstein(li), Lithuania(lt), Luxembourg(lux), Latvia(lv), Lebanon(lb), Morocco(ma), Moldova(md), Madagascar(mg), Marshall Islands(mi), Micronesia(mi), Macedonia(mk), Mali(mli), Myanmar(mm), Mongolia(mn), Macau(mo), Mauritania(mr), Malta(mt), Mauritius(mu), Maldives(mv), Mali(mw), Mexico(mx), Malaysia(my), Mozambique(mz), Namibia(na), Nigeria(ni), Namibia(ni), Netherlands(nl), Norway(no), Nepal(np), Nauru(nr), New Zealand(nz), Oman(om), Panama(pa), Peru(pe), Papua New Guinea(gp), Philippines(ph), Pakistan(pk), Poland(pl), Portugal(pt), Palau(pw), Paraguay(py), Qatar(qa), Serbia(rs), Russia(ru), Rwanda(rw), Saudi Arabia(sa), Solomon Islands(sb), Seychelles(sc), Sudan(sd), Sweden(se), Singapore sg), Slovenia(si), Slovakia(sk), Sierra Leone(sl), Yugoslavia(sy), Somalia(so), Spain(sp), Suriname(sr), Sao Tome & Principe(st), El Salvador(sv), Syria(sy), Swaziland(sz), Tibet(tb), Chad(td), Togo(tg), Thailand(th), Tajikistan(tj), East Timor(tl), Turkmenistan(tm), Vietnam(vn), Tonga(to), Turkey(tr), Trinidad & Tobago(tt), Tuvalu(tv), Taiwan(tw), Tanzania(tz), Ukraine(ua), Uganda(ug), United Nations(un), United States of America(us), Uruguay(uy), Uzbekistan(uz), Vatican City(va), Saint Vincent & The Grenadines(vc), Venezuela(ve), Vietnam(vn), Vanuatu(vu), Samoa(ws), South Africa(za), Zambia(zm), Zimbabwe(zw).

The training dataset of 200 flag images is shown in Fig. 2.
3.2 Training Phase
Samples in the dataset consists of both inter-class similarity as well as intra-class differences. Following figure depict some examples of inter-class similarity, where visual similarity between different flags pose challenges for a computer vision application for discrimination. Some of the pairs of most similar flags from the training set is shown in Fig. 3.
The variation of the 72-element feature vector of the first nine flags of the training set are shown in Fig. 4.

**Fig 4: Plot of 72-element feature vector for each flag**

Fig. 5 shows the variation of the features $\mu(R), \mu(G), \mu(B)$ for all the 200 flags of the training set averaged over all the cells of each image.

**Fig 5: Mean variations of $\mu(R), \mu(G), \mu(B)$ for all flags**

Fig. 6 shows the variation of the features $\sigma(R), \sigma(G), \sigma(B)$ for all the 200 flags of the training set averaged over all the cells of each image.

**Fig 6: Mean variations of $\sigma(R), \sigma(G), \sigma(B)$ for all flag**

### 3.3 Testing Phase

Apart from similarity between different flags, another challenge for computer vision applications arise from intra-class dissimilarities i.e. differences in appearances of the same flag taken from different sources. Differences arise due to varying color shades, variations in size and orientations of patterns like stars and motifs, as well as variations in the overall dimensions of the flags. Fig. 7 shows some instances of intra-class differences between appearances of the same flag taken from the training set (left) and testing set (right).

**Fig 7: Intra-class differences**

Fig. 8 shows how the color features differ between the same flags of the two sets. The blue line indicates the training sample (T) and the red line the test sample (S).

**Fig 8: Color feature variation for intra-class differences**
3.4 Classification

The classification is done using the k-NN classifier with k set to 1 and using Euclidean distance as metric. Fig. 9 shows the classification plot depicting the correspondence between the training set and testing set. With a 4 × 3 partitioning scheme, out of 200 samples, 198 samples were correctly classified leading to a recognition accuracy of 99%.

Fig 9: Classification plot using k-NN with 4 × 3 partitioning scheme

There are two mis-classifications out of 200 samples: T-161 (sk) is being classed as S-152 (ru) and T-170 (sy) is being classed as S-58 (eg). The misclassified samples are shown below, in Fig. 10.

Fig 10: Mis-classified samples

4. ANALYSIS

To arrive at the optimum partitioning scheme, recognition accuracies have been computed using a number of different partitions. The feature vector specified in equations (2) and (3) have been generated from each partition and fed to a k-NN classifier for classification. It is observed that the 4 × 3 partitioning scheme is the most flexible and efficient scheme when both classification accuracy and running time is concerned.

Table 1 indicates the accuracies obtained using different schemes.

<table>
<thead>
<tr>
<th>Partitions</th>
<th>Descriptor Length</th>
<th>% accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 × 3</td>
<td>54</td>
<td>98</td>
</tr>
<tr>
<td>3 × 4</td>
<td>72</td>
<td>96.5</td>
</tr>
<tr>
<td>3 × 5</td>
<td>90</td>
<td>98.5</td>
</tr>
<tr>
<td>4 × 3</td>
<td>72</td>
<td>99</td>
</tr>
<tr>
<td>4 × 4</td>
<td>96</td>
<td>98</td>
</tr>
<tr>
<td>4 × 5</td>
<td>120</td>
<td>99</td>
</tr>
<tr>
<td>5 × 3</td>
<td>90</td>
<td>98</td>
</tr>
<tr>
<td>5 × 4</td>
<td>120</td>
<td>97.5</td>
</tr>
<tr>
<td>5 × 5</td>
<td>150</td>
<td>98.5</td>
</tr>
</tbody>
</table>

Accuracy for partitioning scheme up to 5 × 5, is shown in Fig. 11.

Another important observation is that, with increasing number of rows in a partitioning scheme, the accuracy is better. This is justified as most symbolism in flags is expected to be placed horizontally in a vertical middle alignment. Thus an increasing number of rows in partitioning scheme is helpful.

Table 2 lists various accuracies obtained using k-NN with 4 × 3 partitioning scheme and various distance metrics.

<table>
<thead>
<tr>
<th>Distance scheme</th>
<th>% accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Euclidean</td>
<td>99</td>
</tr>
<tr>
<td>Standardized Euclidean</td>
<td>97</td>
</tr>
<tr>
<td>Cityblock</td>
<td>92.5</td>
</tr>
<tr>
<td>Chebychev</td>
<td>97.5</td>
</tr>
<tr>
<td>Mahalanobis</td>
<td>96</td>
</tr>
<tr>
<td>Correlation</td>
<td>97</td>
</tr>
</tbody>
</table>

Accuracy was also tested using an SVM classifier. The classification plot is shown in Fig. 12. With 4 × 3 partitioning scheme, Out of 200 samples, 169 were correctly classified.
giving an accuracy of 84.5%. It was also found that SVM performs rather slow when compared to k-NN.

Finally in order to visualize the distribution of the data based on the proposed color features, the mean of the three channels for all partitions are combined together into a combined mean value (mean $\mu$) and likewise the std of the three color channels for all partitions are combined together into a combined std value (mean $\sigma$). Each flag is then plotted in this two dimensional space. The resulting plot for the training set is shown in Fig. 13.

![SVM classifier performance](image)

**Fig 12:** Classification plot using SVM

![Flag data plotted in 2 dimensions](image)

**Fig 13:** Flag data plotted in 2 dimensions

5. CONCLUSIONS AND FUTURE SCOPE

The paper proposes an automated system for flag identification from synthetic images and shows detailed analysis of the system that has been implemented. During the tests the first two color moments, namely, mean and standard deviation is found to be effective for synthetic images. The system tries to keep as much spatial features as possible by partitioning every input image in spatial regions. It is found that, a 4 x 3 partitioning scheme works best for flags. It gave us the maximum accuracy of 99% with both the K-Nearest Neighbor and Euclidean Distance classification. Further tests with natural flag images show a not so satisfactory result. One future endeavor will be to expand the research to account for natural flag images. Studies suggest that combination of color moments and Gabor texture features might produce better results in image categorization [5]. So, along with color moments, other features, such as color correlograms, HSV histograms etc [6][7][8][9][10] should also be investigated to see how they act with both synthetically generated flag images as well as natural images of flags.

6. REFERENCES


