# De-duplication of Photograph Images using Histogram Refinement

N. Pattabhi Ramaiah*, C. Krishna Mohan*
*Department of Computer Science and Engineering
Indian Institute of Technology Hyderabad, Hyderabad-502205, Andhra Pradesh, India
Email: ramaiah.iith@gmail.com, ckm@iith.ac.in

***Abstract*—Content based image retrieval (CBIR), a technique which uses the content like color, texture and shape to search images from the large scale databases, is an active research area. In this paper, de-duplication process of photographs was implemented using CBIR. The CBIR technique uses color histogram refinement feature. The photograph data was divided into different clusters using k-means clustering algorithm. The clusters count depends on the numbers of photographs in each district of the state. The photo de-duplication exercise was carried out in a large photograph database which contains 22 million (approximately) photograph images. The experimental results shows that there were 0.35 million (approximately) duplicate photographs.**

***Keywords*-color; texture; histogram refinement; k-means clustering; Daubechies-4 wavelet transform; de-duplication;**

## I. Introduction

The Targeted Public Distribution System (TPDS) is a mechanism for ensuring access and availability of food grains and other essential commodities at subsidized prices to the households. Identification of eligible beneficiaries and ensuring delivery of commodities to them effectively and efficiently is the main challenge for TPDS. As part of this, one department of Civil Supplies in India has issued around 22 million ration cards covering around 80 million citizens and this process was decentralized. The department noticed that there are some bogus ration cards and decided to execute the de-duplication process on entire data. De-duplication is carried out in two different ways, one is biometric-based and the other is photo-based. The reason to go for photo-based de-duplication is that there are some ration cards without biometrics. In this paper, an attempt is made to explain the de-duplication process of photograph images.

Photo-based De-duplication means finding the duplicate ration cards based on the family photograph in the large scale database. The operators generated some duplicate ration cards using the family photographs of the already existing ration cards. They manipulated the photographs in such a way that they edit the photograph in an image editor tool, crop the corners of the photograph in rectangular shape, erase the corners' image data and finally zoom the photograph image into the actual photograph image size.

The methods described in [1 - 3], use color histogram refinement technique using color coherent vectors, color and texture based CBIRs. In this paper, we propose an algorithm to de-duplicate photograhs using histogram refinement for CBIR (Content-Based Image Retrieval). Histogram refinement splits the pixels in a given bucket into several classes based on some local property. Within the given bucket, only pixels in the same class are compared. We have fixed the Equal Error Rate at the threshold of 1500, by observing the False Accept Rates and False Reject Rates on sample training dataset which is created from the database of 22 million photographs.

The entire photographic data was clustered into different clusters. Here the clustering takes place in two levels, one is district-level clustering and the other is k-means clustering. District-level clustering means dividing the data into different clusters based on district names. There may be chances that a family can have two or more ration cards in different districts. The reason for not considering the state as single unit for the de-duplication process is to speed up the process. There are 23 clusters formed based on district name. The next level of clustering is k-means clustering which uses the color and texture features of the photograph.

This paper is organized as follows. In Section II, the CBIR technique using color histogram refinement is presented. Feature extraction of photograph image is explained in Section II-A. K-means clustering algorithm is presented in Section II-B. De-duplication process of photographs are explained in Section II-C. Experimental results and Conclusions are given in Sections III and IV respectively.

## II. CBIR technique using color histogram refinement

The proposed CBIR method uses the family photograph images. Fig 1 represents the sample photograph image in RGB color space which has the dimensions of $320 \times 240$ image size. Generally, images are two-dimensional arrays of bytes which represent pixels. Each pixel has a color value which is ranging from 0 to 255.

The approach of de-duplication of photographs using CBIR is presented as follows:

1) Feature extraction from photograph image.
2) Clustering of photographs using k-nearest neighbor algorithm.
3) De-duplication of photographs.

In the following subsections, each of the above steps is explained in detail.

978-1-4244-9477-4/11/$26.00 ©2011 IEEE

Figure 1. Family Photograph of a Household Ration card

### A. Feature extraction from photograph images

*Color histograms:* Color histograms are more popular to compare images. They tend to be robust against small changes in camera viewpoint. For example, Swain and Bellard [7] use color histograms for analyzing the data. A color histogram is represented by a vector H=$< c_0, c_1, ...c_{n-1} >$ where n = 256 and $c_j$ contains the number of pixels of color j in the image. Color images are reprented in RGB color space. Each image consists of three color histograms namely $R_h$ histogram, $G_h$ histogram and $B_h$ histogram. The color histogram H is an average of $R_h$ , $G_h$ and $B_h$ histograms. Images with same histograms may have entirely different appearances. Histogram refinement [8] technique solves this problem.

*Histogram refinement:* This method divides the color histograms into two different histograms based on local features. The local feature, diagonal line is used to split the color histogram into two histograms namely left diagonal histogram $H_L$ and right diagonal histogram $H_R$ . The resulting split histograms can be compared using the L1 distance measure. The left diagonal color histogram is $H_L$=$< cL_0, cL_1, ...cL_{n-1} >$ and the right diagonal color histogram is $H_R$=$< cR_0, cR_1, ...cR_{n-1} >$. The number of pixels of color j in the image becomes $c_j$=$cL_j$+$cR_j$

$$H = H_L + H_R =< cL_0 + cR_0, cL_1 + cR_1, ...cL_{n-1} + cR_{n-1} > \quad (1)$$

*Feature extraction:* The photograph images have pixels data of 2D array of 320x240 which is in RGB (Red, Green, Blue) color space. The image is partitioned into 4800 4x4 blocks. Each block is represented with one feature vector, consisting of six features [10]. Three of them are the average color components in a 4 x 4 block. The other three represent energy in high frequency bands of wavelet transforms [9], that is, the square root of the second order moment of wavelet coefficients in high frequency bands.

The RGB color space is only rarely used for querying as it does not correspond well to the human color perception. It seems reasonable to be used for photograph images taken under almost identical conditions each time. Although the duplicate photograph have the same capturing conditions as the original photograph, it may be altered while editing by the operators. The CIE (International Commission on Illumination) Luv sapce is much better with respect to human perception. This means that differences in the color space are similar to the differences between colors that humans perceive. The CIE, defined three standard primaries (X, Y, and Z) to replace red, green, and blue, because all visible colors could not be specified with positive values of red, green and blue components. With this newly created X, Y, and Z primaries, all visible colors could be specified with only positive values of the primaries. The CIE Luv color space is a perpetually uniform derivation of this standard CIE XYZ space. The photograph image is transformed from RGB space to CIE Luv space [5], and then the features of the three color components are calculated . To obtain the other three features, we apply the Daubechies-4 wavelet transform to the L component of the image. After a one-level wavelet transform, a 4 x 4 block is decomposed into four frequency bands as shown in Fig 2.

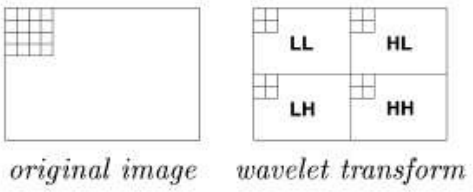


Figure 2. Decomposition of images into frequency bands by wavelet transforms

Each band contains 2 x 2 coefficients. Without loss of generality, suppose the coefficients in the HL band are $(C_{k,l}, C_{k,l+1}, C_{k+1,l}, C_{k+1,l+1})$. Then one feature is

$$f = \left[ \frac{1}{4} \sum_{i=0}^{1} \sum_{j=0}^{1} C_{k+i,l+j}^2 \right]^{\frac{1}{2}}, \quad (2)$$

The other two features are computed similarly from the LH and HH bands. Finally compute the feature vector by taking the average of all the corresponding feature vectors of 4800 blocks.

### B. k-means clustering algorithm

The k-means algorithm [4] and its development algorithms are in the family of prototype based clustering algorithms. The general steps of the prototype based clustering are:

1) Let x be the feature vector consisting of 6 features. Initialize the centers $c_i$ arbitrarily where i = 1 to n, n is number of clusters

2) For each feature vector x, compute its minimum distance with each center $c_i$ and assign the data point to $i^{th}$ cluster.
3) For each center $c_i$, recomputed the new center from all feature vectors **x** belong to this cluster.
4) Repeat steps 2 and 3 until convergence.

The input for the k-means clustering algorithm is feature vectors and the number of clusters. Each vector has the length 6 (3-color components, 3-texture components). The cluster count depends on the numbers of ration cards in the district. Once the clustering is over, each cluster undergoes the de-duplication process which is explained in section II-C.

### C. De-duplication process of photographs

There are two phases involved in the process of photograph de-duplication. In Phase-I, the pre-processing steps for the de-duplication process is explained. In Phase-II, the actual de-duplication process is explained.

*Phase-I:*

1) Resize the photographs to 320x240 (if required).
2) For each photograph, apply the histogram refinement technique and compute the histogram pair ($H_L$, $H_R$) where $H_L$ is the left diagonal histogram and $H_R$ is the right diagonal histogram.
3) For each photograph, compute the feature vector of length 6, consisting of 3 color and 3 texture components.
4) Divide the entire data into 23 clusters based on district name.
5) Each district-level cluster is further clustered using the k-means clustering algorithm. The input data for this algorithm is feature vector which is computed in Phase-I (Step 3). Number of clusters depend on the total number of photographs in each district.
6) Apply de-duplication process for each cluster.

*Phase-II (for each cluster):*

1) Pick one histogram pair from the set of histograms pairs $\{(H_L, H_R)\}$, say query histogram pair ($HQ_L$, $HQ_R$) which is not yet participated as a query histogram pair.
2) The similarity score ($L_1$-distance) is calculated between the query histogram pair and the set of all the histogram pairs which are not participated as query histogram pair. The $L_1$-distance between the pairs ($HQ_L$, $HQ_R$) and ($H_L$, $H_R$) is defined as follows:

$$score = \sum_{i=0}^{n-1}(|HQ_L^i - H_L^i| + |HQ_R^i - H_R^i|) \quad (3)$$

3) List the top 20 matches which have similarity score less than or equal to 1500 (empirical threshold).
4) The results are verified manually whether the results are correct or not. It is required because there is no guarantee that all the results are genuine. There are more chances of False Accepts.

## III. Experimental Results

Table 1 shows the household information of 23 districts and the corresponding duplicates in each district with number of classes clustered in each district. The de-duplication process used two 64-bit Windows 2000 servers having quadcore processor. Three 32-bit Windows XP systems were used to extract features and for clustering. For each server, two de-duplication instances run, one instance is top-to-bottom de-duplication instance and the other is bottom-to-top de-duplication instance. Each instance used 25 threads to compute the similarity score using $L_1$-distance. The photo de-duplication process was carried out in a large scale database of which contains 22916243 photograph images. The experimental results shows that there were 353650 duplicate phtographs. Fig 3(a), 3(c), 3(e), 3(g) and 3(i) are the original photographs. Fig 3(b), 3(d), 3(f), 3(h) and 3(j) are the duplicate photographs.

Table I
De-duplication of photographs results

| | Household Cards or Family Photographs | | |
|---|---|---|---|
| District | # of Images | Duplicates | # of Classes |
| D-1 | 774074 | 12685 | 3 |
| D-2 | 646537 | 10044 | 3 |
| D-3 | 1226020 | 18616 | 5 |
| D-4 | 1358529 | 22736 | 5 |
| D-5 | 1041159 | 13658 | 4 |
| D-6 | 1208362 | 22333 | 5 |
| D-7 | 1390698 | 20251 | 5 |
| D-8 | 858607 | 12737 | 3 |
| D-9 | 833137 | 9330 | 3 |
| D-10 | 1067074 | 29073 | 4 |
| D-11 | 811924 | 10990 | 3 |
| D-12 | 1116510 | 17864 | 4 |
| D-13 | 1027658 | 13442 | 4 |
| D-14 | 1066708 | 15067 | 4 |
| D-15 | 1300346 | 14805 | 5 |
| D-16 | 1262954 | 18207 | 5 |
| D-17 | 725254 | 10604 | 3 |
| D-18 | 627405 | 11038 | 3 |
| D-19 | 704406 | 8270 | 3 |
| D-20 | 1087370 | 18397 | 4 |
| D-21 | 1037474 | 16599 | 4 |
| D-22 | 756072 | 12097 | 3 |
| D-23 | 987965 | 14807 | 4 |

## IV. Conclusion

In this paper, a de-duplication process was implemented in large scale database of photographs. In the proposed method, an attempt is made to eliminate the duplicate ration cards from the database using histogram refinement technique. To speed up the de-duplication process, the entire data is clustered into different clusters using district-level clustering and k-means clustering of each district. The proposed method eliminated nearly 0.35 million (approximately) duplicate ration cards.

(a) (b)

(c) (d)

(e) (f)

(g) (h)

(i) (j)

Figure 3. Duplicate Household Ration cards

## REFERENCES


[1] J. Smith, S. Chang, *Tools and Techniques for Color Image Retrieval*, Proc. SPIE, vol. 2670, pp. 1630-1639, 1996.

[2] J. Puzicha, T. Hofmann, and J. Buchmann, *Histogram Clustering for Unsupervised Segmentation and Image Retrieval*, Pattern Recognition Letters, vol. 20, no. 9, pp. 899-909, 1999.

[3] M. Stricker and M. Orengo, *Similarity of Color Images*, Proc. SPIE Conf. Storage and Retrieval for Image and Video Databases III, vol. 2420, pp. 381-392, 1995.

[4] D. Malyszko, S. T. Wierzchon, *Standard and Genetic k-means Clustering Techniques in Image Segmentation*, (CISIM'07) 0-7695-2894-5/07 IEEE 2007

[5] A. K. Jain, *Fundamental of Digital Image Processing*, Prentice Hall, 1989.

[6] Gonzalez, Rafael C. and Woods, Richard E., *Digital Image Processing*, 2nd ed. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1992.

[7] Michael Swain, Dana Ballard, *Color indexing*, International Journal of Computer Vision, 7(1):11-32, 1991.

[8] Greg Pass, Ramin Zabih, *Histogram Refinement for Content-Based Image Retrieval*, IEEE Workshop on Applications of Computer Vision, pp. 96-102, 1996.

[9] I. Daubechies, *Ten Lectures on Wavelets*, Capital City Press, 1992.

[10] Y.Chen, J. Z. Wang, *A region-based fuzzy feature matching approach to content-based image retrieval*, IEEE Trans. Pattern Analysis and Machine Intelligence, 2002.