# Color Image Quantization Techniques Based on Image Decomposition for Memory Consumption in Cloud Environment

[1]Dr. Ramakrishnan, [2]S. Ahamed Ali, [3]K. R. Mohan Raj, [4]S. Subramanian
[1,2,3,4]Velammal Engineering College, Chennai, Tamilnadu, India

## Abstract

Cloud computing becomes a hot research topic in the recent years. In the cloud computing, software applications and databases are moved to the centralized large data centres, which is called cloud. In the cloud, due to lack of physical possession of the data and the machine, the video and audio data's may possess more memory space. Existing work on cloud mainly focuses on cloud storage without taking memory consumption into consideration. In this paper, we propose a new Colour Quantisation (CQ) technique is introduced. The CQ technique is based on video/image split into sub-images and the use of self-organised neural network classifiers (SONNC). Initially, the dominant colours of each sub-image are extracted through SONNCs and then are used for the quantisation of the colours of the entire image scheme to consume memory in cloud computing based on decomposition technique. We also discuss how to optimize memory for cloud users to minimize the utility cost. Detailed analysis and simulations have demonstrated the effectiveness and efficiency of the proposed scheme. Applying a systematic design methodology to the developed CQ algorithm, an efficient cloud architecture achieving high-speed processing and less memory consumption, is derived.

## Keywords

Cloud Computing, Colour Quantisation Technique. Public Cloud, Private Cloud

## I. Introduction

Cloud Computing has been envisioned as the next-generation architecture of IT enterprise, due to its long list of unprecedented advantages in the IT history: on-demand self-service, ubiquitous network access, location independent resource pooling, rapid resource elasticity, usage-based pricing and transference of risk [1]. As a disruptive technology with profound implications, Cloud Computing is transforming the very nature of how businesses use information technology. One fundamental aspect of this paradigm shifting is that data is being centralized or outsourced into the Cloud. From users perspective, including both individuals and IT enterprises, storing data remotely into the cloud in a flexible on-demand manner brings appealing benefits: relief of the burden for storage management, universal data access with independent geographical locations, and avoidance of capital expenditure on hardware, software, and personnel maintenances, etc [2].

While Cloud Computing makes these advantages more appealing than ever, it also brings new and challenging security threats towards users' outsourced data. Since Cloud Service Providers (CSP) are separate administrative entities, data out- sourcing is actually relinquishing user's ultimate control over the fate of their data. As a result, the correctness of the data in the cloud is being put at risk due to the following reasons. First of all, although the infrastructures under the cloud are much more powerful and reliable than personal computing devices, they are still facing the broad range of both internal and external threats for data integrity. Examples of outages and security breaches of noteworthy cloud services appear from time to time [3-5]. Secondly, for the benefits of their own, there do exist various motivations for cloud service providers to behave unfaithfully towards the cloud users regarding the status of their outsourced data. Examples include cloud service providers, for monetary reasons, reclaiming storage by discarding data that has not been or is rarely accessed, or even hiding data loss incidents so as to maintain a reputation [6-8]. In short, although outsourcing data into the cloud is economically attractive for the cost and complexity of long-term large-scale data storage, it does not offer any guarantee on data integrity and availability. This problem, if not properly addressed, may impede the successful deployment of the cloud architecture.

Color image quantization is a process that reduces the number of distinct colors used in an image, usually with the intention that the new image should be as visually similar as possible to the original image. Computer algorithms used to perform color quantization on bitmaps have been studied since the 1970s. Color quantization is critical for displaying images with many colors on devices that can only display a limited number of colors, usually on devices with smaller memory footprint, processing speed and less power consumption. Color quantization also enables efficient compression of certain types of images.

Image colour quantisation is a very useful tool for segmentation, compression, presentation and transmission of images. Usually, true type colour images consist of more than 16 million different colours in a 24-bit colour space. Reduction of the image colours, which is the result of the application of a CQ technique, is a process that reduces the number of image colours to a limited number of dominant image colours. This technique is advantageous in many cases, because it is easier to process and understand an image represented by a limited number of dominant colours. Colour quantisation and related techniques, due to large amount of data, are associated with large computational complexity. This implies that their implementation, either in software or hardware, should be effective, for instance, in terms of speed and memory size.

Color quantization helped to justify these tradeoffs by making it possible to display many high color images in 16 and 256-color modes with limited visual degradation. The Windows operating system and many other operating systems automatically perform quantization and dithering when viewing high color images in a 256 color video mode, which was important when video devices limited to 256 color modes were dominant. Modern computers can now display millions of colors at once, far more than can be distinguished by the human eye, limiting this application primarily to embedded devices such as mobile devices

In general, a CQ technique consists of two main stages: (i) the palette design stage, and (ii) pixel mapping stage. More specifically, in the foregoing stage a limited number of image dominant colours are obtained, which actually determine the palette of colours. In the latter stage, the colour of each pixel is assigned to the nearest of the palette colours. Apart from the two main stages, especially in the case that the applied CQ technique converges to limited number of colours, a pre-processing technique is desirable in order to estimate the proper Number of Dominant Image Colours (NDIC). In this case the CQ technique converges to a proper number of dominant colours, without supervision. If we consider only the image colours, then the CQ problem can be seen as an unsupervised vector classificat ion problem in a 3-D colour space, where each vector represents a triplet of the image colour components. However, as is shown by Papamarkos [1], in many cases it is preferable to associate each pixel not only with its colour, but also with spatial features extracted from the neighbouring pixels. Doing this, the spatial characteristics of the final image after colour quantisation approximate better the spatial characteristics of the initial image.

All the above techniques construct colour palettes by performing global clustering in the colour space. In other words, they perform colour quantisation taking into account all the image colours simultaneously. Thus, the constructed colour palette represents a codebook in which all the image colours are optimally mapped. However, the global clustering approach exhibits two main drawbacks related to its hardware implementation, which are the high computational complexity cost and the huge memory requirements. In order to overcome these limitations, we propose a novel decomposition scheme, which is based on splitting of the initial clustering problem into a pre-defined number of sub-problems each of which can be easily realised by an application-specific embedded system. During the final step of the proposed technique, we compose the final colour palette of the entire image exploiting the colour palettes of all the sub-problems.

### A. Step 1

A SONNC is trained to perform global CQ and create the colour palette of the entire image. The new SONNC is trained only with the vectors that correspond to the dominant colour classes of all sub-images, taking into account the contribution of each class. Using a modified training weight-updated formula, the training operation is achieved.

### B. Step 2

The final mapping step, where all image pixels are mapped on the vectors of the created palette, derives the final quantised image.

### C. Step 3

An optional post processing step, the colour correction step, can be applied after the final mapping procedure to improve the colour distortion. In this step, the final colours of the created palette are corrected to the average colour values of the pixels classified in the corresponding classes.

## II. The Proposed Color Reduction Technique

To simplify our approach, let us consider a digital image of n pixels and the corresponding data set X , consisting of n input vectors Xk are obtained.

$$X_k = \left[x_{k,1} \quad x_{k,2} \quad \dots \quad x_{k,D}\right]^T \text{ , } k = 1,\dots,n$$

Each input vector Xk consists of the pixel's color components and additional spatial features extracted from a neighborhood region of this pixel. These feature vectors are the inputs to the SONNC network classifier. Now the training has been performed. After training, the final color palette is obtained. Depending on the initial settings, the SONNC network converges to c neurons. In other words, c weight vectors, which express the position of the created neurons in the output space and the connections between neurons, are obtained. Finally, the quantized image is constructed by replacing the initial colors with the closer colors of the created palette.

## III. Image Quantisation and Feature Vector Extraction in Cloud

The proposed SONNC network consists of two separated layers of fully connected neurons, the input layer and the output mapping layer. In contrast with the Kohonen SOFM, the space of the mapping layer has always the same dimensionality with the input space, and also, the created neurons take their position in order the structure of neurons to approximate the structure of the input training vectors. In other words, the topology of the created neurons always approximates the topology of the training vectors. All vectors of the training data set X' are circularly used for the training of the SONNC network. In contrast with the GNG network, where a new neuron is always inserted at the end of each epoch, in the proposed algorithm three new criteria are applied that control the growing of the output lattice of neurons. In summary, we introduce the following three criteria:

- A neuron is removed if, for a predefined number of consecutive epochs, none of the input vectors has been classified to the corresponding class.
- A new neuron is added, close to the neuron with the maximum contribution in the total quantization error if the average distance of the training vectors, classified to the corresponding class, from their neighbouring classes is greater from a threshold value.
- All neurons are sorted according to their importance. In the less important class, the classified vectors have the smallest average distance from their neighbouring classes. This class is removed if its importance is below a threshold value.

## IV. The Training Steps of the SONNC Network

The training procedure for the SONNC neural classifier starts by considering first two output neurons ( c= 2 ). The local counters Ni , i=1,2 of created neurons are set to zero. The initial positions of the created output neurons, that is, the initial values for the weight vectors Wi, i=1,2 are initialized by randomly selecting two different vectors from the input space. All the vectors of the training data set X' are circularly used for the training of the SONNC network. The training steps of the SONNC are as follows:

### A. Step 1

At the beginning of each epoch the accumulated errors AE1, AE2 are set to zero. The variable AE1 expresses, at the end of each epoch, the quantity of the total quantization error that corresponds to Neuroni , while the variable AE2 , represents the increment of the total quantization error that we would have if the Neuroni was removed.

### B. Step 2

For a given input vector Xk , the first and the second winner neurons Neuronw1 , Neuronw2 are obtained.

### C. Step 3

The local variables AE1 and AE2 change their values according to the relations:

AE(n) = AE(n) + || X(k)-W(n) ||

### D. Step 4

Now the learning rate li have been calculated. The learning rate l1 is applied to the weights of Neuroni if this is the winner neuron (w1= i), while l2 is applied to the weights of Neuroni if this belongs to the neighbourhood domain of the winner neuron . The learning rate l2 is used in order to have soft competitive effects between the output neurons. That is, for each output neuron, it is necessary that the influence from its neighbouring neurons to be gradually reduced from a maximum to a minimum value. The values of the learning rates li are not constant but they are reduced according to the local counter Ni. Doing this, the potential ability of moving of neuron i toward an input vector is reduced with time.

### E. Step 5

In accordance with the Kohonen SOFM, the weight vector of the winner neuron Neuronw1 and the weight vectors of its neighbouring neurons are adapted.

### F. Step 6

With regard to generation of lateral connections, SONNC employs the following strategy. The CHR is applied in order to create or remove connections between neurons. As soon as the neurons Neuronw1 and Neuronw2 are detected, the connection between them is created or is refreshed.

### G. Step 7

At the end of each epoch it is examined if all neurons are in idle state, or equivalently, if all the local counters Ni , i[1,c] are greater than the predefined value Nidle and the neurons are well trained. In this case, the training procedure stops, and the convergence of SGONG network is assumed. The number of input vectors needed for a neuron to reach the "idle state" influences the convergence speed of the proposed technique. If the training procedure continues, the lateral connections between neurons with age greater than the maximum value are removed. Due to dynamic generation or removal of lateral connections, the neighbourhood domain of each neuron changes with time in order to include neurons that are topologically adjacent.

Step 8 At the end of each epoch, three criteria that modify the number of the output neurons and make the proposed neural network to become self-growing are applied. These criteria are applied in the following order:

- remove the inactive neurons,
- add new neurons,
- and finally, remove the non important neurons.

## IV. Local K-means Algorithm for Color Image Quantization

Commonly used color quantization algorithms take a recursive pre-clustering approach. Such techniques do not account for complex interrelationships between color clusters. They also tie the error minimization process to the rough approximation of the quantized regions. This algorithm deals with a color quantization problem for the windows system. In such environments it is desirable to account for already allocated colors in the shared color map. This work examines the use of recently designed K-means algorithm. This work also describes how this post-clustering scheme can be modified for the efficient quantization within windows system.

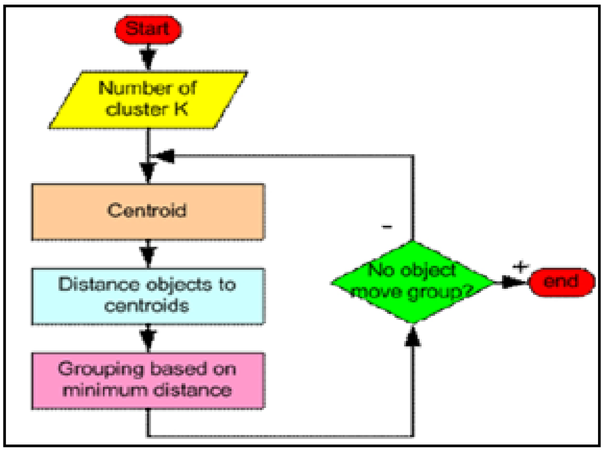
Start
Number of cluster K
Centroid
Distance objects to centroids
Grouping based on minimum distance
No object move group?
-
+
end

Fig. 1:

### A. Step 1

Begin with a decision on the value of k = number of clusters.

### B. Step 2

Put any initial partition that classifies the data into k clusters. You may assign the training samples randomly, or systematically.

### C. Step 3

Take each sample in sequence and compute its distance from the centroid of each of the clusters. If a sample is not currently in the cluster with the closest centroid, switch this sample to that cluster and update the centroid of the cluster gaining the new sample and the cluster losing the sample.

### D. Step 4

Repeat step 3 until convergence is achieved, that is until a pass through the training sample causes no new assignments. If the number of data is less than the number of cluster then we assign each data as the centroid of the cluster. Each centroid will have a cluster number. If the number of data is bigger than the number of cluster, for each data, we calculate the distance to all centroid and get the minimum distance. This data is said belong to the cluster that has minimum distance from this data. Since we are not sure about the location of the centroid, we need to adjust the centroid location based on the current updated data. Then we assign all the data to this new centroid. This process is repeated until no data is moving to another cluster anymore

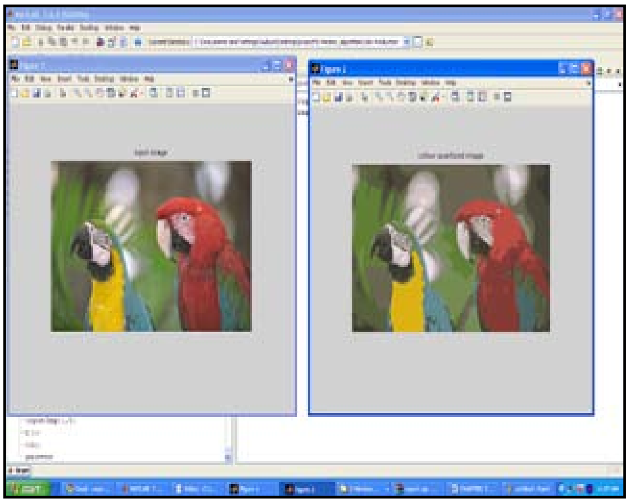

Fig. 2:

## V. Application of Local cq

The process of image splitting and application of local have been shown in the following fig.

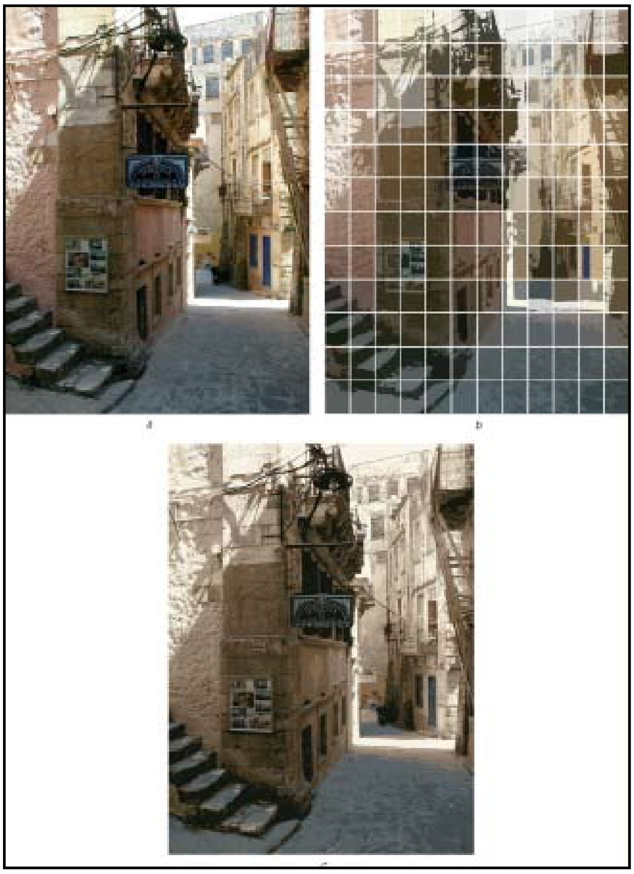

Fig. 3: Proposed CQ Technique Applied to Image With 204 760 Colours

1. Initial image with 204 760 unique colors
2. Initial image is split into 12 _ 12 ¼ 144 sub-images, each sub-image quantised to two colors.
3. Final image quantisation to only 16 unique colors.

## VI. Conclusion

In this work analysis between basic colour quantization of an image and colour quantization using K-Mean Algorithm has been compared and studied. My future works includes division of an image into number of sub images and quantization of an image will be done using Kohonen self organizing neural network.

## References

[1] Papamarkos, N.,"Colour reduction using local features and a SOFM neural network", Int. J. Imaging Syst. Technol., 1999, 10, (5), pp. 404–409.

[2] Heckbert, P.,"Colour image quantisation for frame buffer display", Comput. Graph., 1982, 16, (3), pp. 297–307.

[3] Ashdown, I.,"Octree colour quantisation in ‘Radiosity-a programmer’s perspective", John Wiley, 1994.

[4] Wan, S.J., Prusinkiewicz, P., Wong, S.K.M.,"Variance-based colour image quantization for frame buffer display", Colour. Res. Appl., 1990, 15, pp. 52–58.

[5] Balasubramanian, R., Allebach, J.P.,"A new approach to palette selection for colour images", J. Imaging Technol., 1990, 17, pp. 284–290.

[6] Balasubramanian, R., Allebach, J., Bouman, C.,"Colour-image quantization with use of a fast binary splitting technique", J. Opt. Soc. Am., 1994, 11, pp. 2777–2786.

[7] Atsalakis, A., Papamarkos, N., Andreadis,"On estimation of the number of image principal colors and color reduction through self-organized neural networks", Int. J. Imaging Syst. Technol., to be Published.

[8] Papamarkos, N., Atsalakis, A., Strouthopoulos, C.,"Adaptive color reduction", IEEE Trans. Syst. Man Cybern. B, 2002, 32, (1), pp. 44–56.

Dr. M. Ramakrishnan is the Professor and Head of Information Technology. He completed his under graduation and post graduation from Madurai Kamaraj University and Doctorate from Anna University Chennai. His research areas of expertise are parallel computing, neural networks and network security. He has 20 years of teaching experience and he has published many research papers in international/national journals and conferences. He is also a life member of ISTE and IACSIT. He has organized FDPs, Workshops and Conferences. He is a reviewer of international journals such as scientific journal of computer science and international journal of computer science and emerging technology.

Mr. S.Ahamed Ali is working as a Assistant Professor Grade III in the Department of IT in Velammal Engineering College, Chennai. He obtained his B.E and M.E Degree in Computer Science and Engineering and currently pursuing his Ph.D degree in Anna University of Technology, Chennai. His research areas of interest include Cloud Computing, Network Security Image processing and Web Services. He has 9 years of teaching experience. He has published many research papers in international/national conferences. He has written a book on Object-Oriented Programming. He has also organized FDPs, Workshops and Conferences.

Mr. K.R Mohan Raj is working as a Assistant Professor Grade I in the Department of IT in Velammal Engineering College, Chennai. He obtained his B.Tech in IT and currently pursuing his M.Tech degree in Anna University, Chennai. His research areas of interest include Image Processing, and Network security. He has a total 7 years of teaching experience.

Mr. S. Subramanian working as a Assistant Professor Grade I in the Department of IT in Velammal Engineering College, Chennai. He obtained his B.Tech and M.Tech Degree in IT from Anna University, Chennai. His current area of research includes Cloud Computing, Image processing and Neural Networks. He has published many research papers in international/national conferences.