ISSN : 2231-0711

IJCSET | June 2011 | Vol 1, Issue 5,218-221

Color Image Quantization Techniques Based On Image Decomposition For Power Consumption For Embedded Systems

S.Subramanian, V.Shivamurugan
Dept of IT, SSN College of Engineering, Chennai

Abstract -- Data transmission over the Internet is prevalent and the development of efficient algorithms for compressing such data in order to achieve reduced bandwidth has been an active research. With increased demand for exchanges of datas over the Internet, research for data_compression is more intense than ever before. Computing techniques that would considerably reduce the number of colours in an image that occupies less space and bandwidth for transmission over networks form an active research. The less space and less bandwidth will also reduce the memory access for displaying image and this will lead to saving considerable amount of power in a resource constrained battery operated embedded system. In this project a new colour quantisation (CQ) technique is introduced. The CQ technique is based on 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. In addition, for the estimation of the proper number of dominant image colours, a new algorithm based on the projection of the image colours into the first two principal components is proposed. Applying a systematic design methodology to the developed CQ algorithm, an efficient embedded architecture based on the ARM7 processor achieving high-speed processing and less energy consumption, is derived.

KEY WORDS:
Colour quantisation technique, Artificial neural network, Kohonen Self-organised neural network classifiers.

1 INTRODUCTION

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.

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.

Step 2. The final mapping step, where all image pixels are mapped on the vectors of the created palette, derives the final quantised image.

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.

2 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.*

$$X_k = \begin{bmatrix} x_{k,1} & x_{k,2} & \dots & x_{k,D} \end{bmatrix}^T , \; k = 1,\dots,n$$

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 prformed. 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.

3 IMAGE QUANTISATION AND FEATURE VECTOR EXTRACTION USING NEURAL NETWORK:

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 neighboring 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 neighboring classes. This class is removed if its importance is below a threshold value.

3.1 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:

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.

Step 2 For a given input vector *Xk* , the first and the second winner neurons *Neuronw1* , *Neuronw2* are obtained.

Step 3 The local variables AE1 and AE2 change their values according to the relations:

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

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 neighborhood 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 neighboring 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.

Step 5 In accordance with the Kohonen SOFM, the weight vector of the winner neuron *Neuronw1* and the weight vectors of its neighboring neurons are adapted.

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.

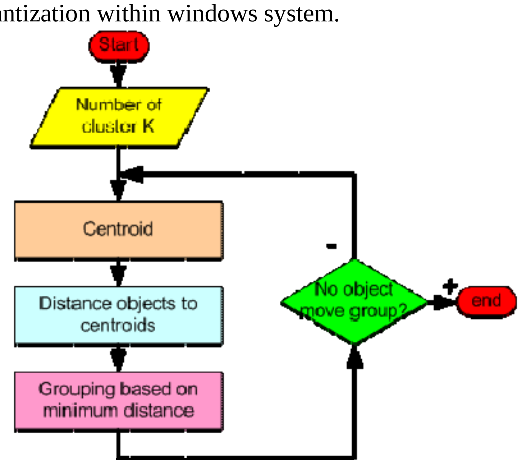
Step 7 At the end of each epoch it is examined if all neurons are in *idle state*, or equivalently, if all the local counters *Ni* , $\forall i \in [1,c]$ are greater than the predefined value *Nidle* and the neurons are well trained. In this case, the training procedure stops, and the convergence of 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 neighborhood 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. Thesecriteria are applied in the following order:

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

4 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.

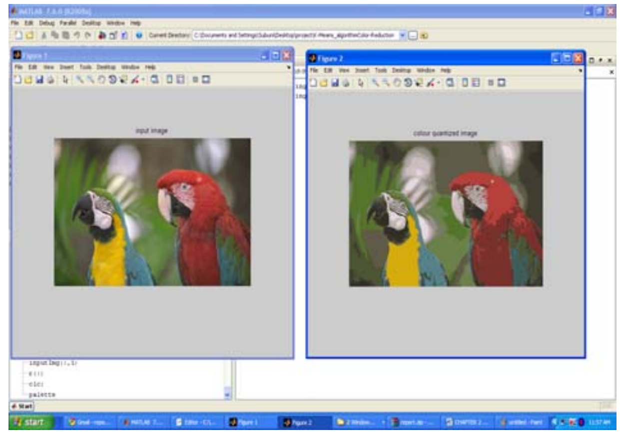


Step 1. Begin with a decision on the value of k = number of clusters.

Step 2. Put any initial partition that classifies the data into k clusters. You may assign the training samples randomly, or systematically.

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.

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.



APPLICATION OF LOCAL CQ:

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

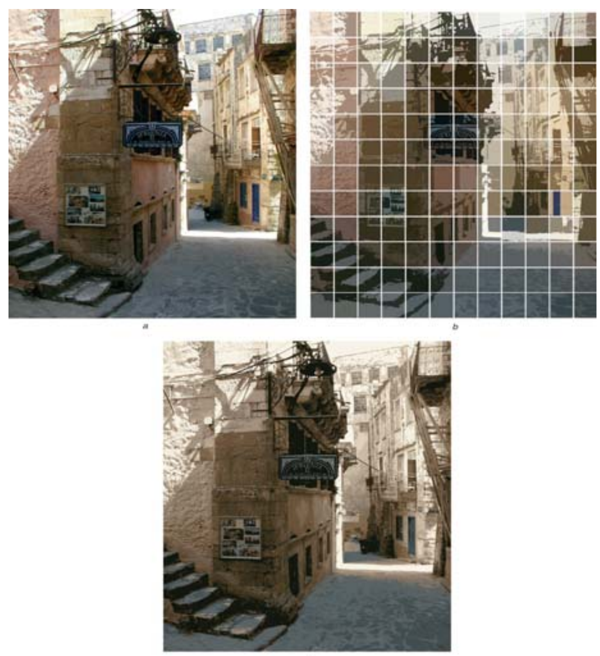


Fig.2 Proposed CQ technique applied to image with 204 760 colours

a Initial image with 204 760 unique colours

b Initial image is split into 12 _ 12 ¼ 144 sub-images, each sub-image quantised to two colours

c Final image quantisation to only 16 unique colours

5 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 subimages and quantization of an image will be done using Kohonen self organizing neural network.

IJCSET | June 2011 | Vol 1, Issue 5,218-221

6 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., and Wong, S.K.M.: 'Variance-based colour image quantization for frame buffer display', Colour. Res. Appl., 1990, 15, pp. 52–58

5 Balasubramanian, R., and 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., and 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., and Andreadis, I.: 'On estimation of the number of image principal colours and colour reduction through self-organized neural networks', Int. J. Imaging Syst. Technol., to be Published.

8 Papamarkos, N., Atsalakis, A., and Strouthopoulos, C.: 'Adaptive colour reduction', IEEE Trans. Syst. Man Cybern. B, 2002, 32, (1), pp. 44–56.