Local K-means Algorithm for Colour Image Quantization

Oleg Verevka
John W. Buchanan
Department of Computing Science
University of Alberta
Edmonton, AB., Canada
T6G 2H1
{oleg,juancho}@cs.ualberta.ca

Abstract

Colour image quantization is the process of representing an image with a small number of well selected colours. Most previous colour quantization techniques use a recursive pre-clustering approach. These algorithms subdivide the colour space into a set of simple geometric regions. Thus, the colour map is chosen on the basis of this approximation.

We propose a new quantization method called local K-means (LKM). It is an iterative post-clustering technique that approximates an optimal palette using multiple subsets of image points. The paper also presents ways to speedup the search of the closest colour for a dynamically changing palette. The local K-means procedure is compared with popular pre-clustering algorithms. The LKM method is able to generate a high quality palette significantly faster than other quantization techniques.

Keywords: Colour quantization, image compression, colour reduction, palette, colour map.

1 Introduction.

Colour quantization is one of the most frequently used operations in computer graphics and image processing. Traditionally, quantization is used to reproduce 24 bit images on graphics hardware with a limited number of simultaneous colours (i.e. frame buffer displays with 4 or 8 bit colourmaps). Even though 24 bit graphics hardware is becoming more common, colour quantization maintains its practical value. It lessens space requirements for storage of image data and reduces transmission bandwidth requirements in multimedia applications.

Colour quantization is usually defined as a *lossy image compression operation* that maps a full colour image to an image with a smaller palette. The mapping substitutes each original image colour by the closest colour from the reduced palette. The objective of the research in quantization is to minimize the perceived distortion in the resulting image. Mathematically this process can be formulated as an optimization problem (see [Wu92]).

Usually quantization algorithms take one of the two possible approaches: *pre-* or *post-clustering* [Dek94].

Previous colour quantization algorithms use a *pre-clustering* scheme. A colour space is partitioned into a set of clusters. The centroids of these clusters define the resulting colour map.

The median-cut algorithm [Hec82] takes a pre-clustering approach. The colour space is recursively subdivided into a set of rectangular boxes by planes parallel to the space axis. The objective of the split is to place an equal number of colours into every rectangular cluster.

The variance based method [WPW90] follows a similar scheme. At each step a box with the largest variance is selected. The partition plane is chosen to be perpendicular to the axis with the smallest sum of projected variances. The goal of such a subdivision is to minimize variance of colour within each rectangular cluster.

The octree algorithm [GP88], [CFM93], [Cri92] relies on a tree structured partitioning of the colour space. The root of the tree represents the entire space. Colours of the original image are placed into the leaves of the octree. Neighboring leaves are recursively merged together.

The algorithms described above have a common flaw. The intermediate clusters are bipartitioned one at a time independently from each other. As a result the quantization process is not able to take into account interrelationships between neighbouring colour clusters. Wu [Wu92] recently proposed the principal multilevel quantization algorithm. The performance of the pre-clustering scheme is improved by simultaneous optimization of multiple cuts.

Minimization in the pre-clustering techniques is tied to approximation of Voronoi clusters. These clusters are usually presented by simple geometric objects. Post-clustering algorithms try to find representative colours first. Voronoi tessellation of the colour space is computed using these representative colours. Post-clustering techniques have been actively studied in statistical analysis, data coding, signal processing and pattern recognition [LBG80], [Gra84], [Fri93], [MG93], [KKL90]. Until now these schemes were considered to be computationally expensive for colour quantization.

The objective of our research is to make a post-clustering technique feasible for colour image quantization. We explored the local K-means scheme [MG93]. This approach is a combination of a K-means quantization [LBG80] and a self-organizing map (or Kohonen neural network) [KKL90]. The scheme presented in this paper is significantly faster and at least as accurate as previous pre-clustering methods: median cut, variance and octree based algorithms.

2 Formulation of the Colour Quantization Problem.

Let c_i be a 3-dimensional vector in one of the colour spaces (Lu^*v^*, HSV, RGB, etc.). The set $C = \{c_i, i = 1, 2 \ldots N\}$ is the set of all colours in the full colour image I. A quantized image $\overline{I}$ is represented by a set of K colours $\overline{C} = \{\overline{c_j}, j = 1, 2 \ldots K\}, K \ll N$. The quantization process is therefore a mapping:

$$q : C \rightarrow \overline{C}. \tag{1}$$

The closest neighbour principle states that each colour c of the original image I is going to be mapped into the closest colour $\overline{c}$ from the colour palette $\overline{C}$:

$$\overline{c} = q(c): \quad ||c - \overline{c}|| = \min_{j=1,2\ldots K} ||c - \overline{c_j}||. \tag{2}$$

The quantization mapping defines a set of clusters S_k, $k = 1, 2 \ldots K$ in the image colour space C:

$$S_k = \{c \in C : q(c) = \overline{c_k}\}. \tag{3}$$

The goal of quantization is to make the perceived difference between the original image and its quantized representation as small as possible. Human vision is an extremely complicated and not yet fully understood process. It is very difficult to formulate a definite solution to the image quantization problem in terms of *perceived* image quality. In fact, there is no good objective criterion available for measuring the perceived image similarity.

In the colour quantization literature it is common to use image dependent distortion measures (see [Hec82], [WPW90], [Wu92] and others). Let an image I be an array of M pixels (x, y), then $c_{(x,y)}$ is the colour of each image pixel. The average quantization distortion per pixel can be defined as follows:

$$\epsilon_{q(C,I)} = \frac{1}{M} \sum_{(x,y) \in I} ||c_{(x,y)} - q(c_{(x,y)})||, \tag{4}$$

where $||*||$ is the Euclidean L_2-norm. Wu in [Wu92] recommends to use CIE Lu^*v^* space where the Euclidean norm can approximate a perceived colour difference.

Even though the average distortion measure $\epsilon_{q(C,I)}$ can give a reasonable estimate of a perceived image difference, it can also be very misleading (see [WPW90]). Colours of the original image are often nonuniformly distributed in the colour space. Thus, significant image information is carried by some distinct but "rare" colours (e.g. specular highlights). If a quantization algorithm approximates the more popular colours, the average distortion might be small, but the "rare" colours of the original will be lost.

Rather than using a single measure of quantization errors we propose to evaluate approximation accuracy by a combination of the average colour distortion:

$$\epsilon_{q(C)} = \frac{1}{N} \sum_{i=1}^{N} ||c_i - q(c_i)||, \tag{5}$$

and the standard deviation of distortion per pixel:

$$\sigma = \sqrt{\frac{\sum_{(x,y)\in I}(||c_{(x,y)} - q(c_{(x,y)})|| - \epsilon_{q(C,I)})^2}{M}}. \tag{6}$$

The objective of our research is to find an algorithm that minimizes both approximation measures simultaneously. Small values of $\epsilon_{q(C)}$ guarantee that a quantization process accurately represents colours of the original image. Unfortunately the human visual system is not able to determine the absolute value of a colour. It is more sensitive to colour variations. A quantization algorithm that produces small values of σ introduces almost equal colour distortion to every pixel. Therefore, the minimization of the standard deviation of distortion σ helps us to preserve variations of colours in the quantized image.

It should be noted that these error measures have a significant limitation. Even though $\epsilon_{q(C,I)}$ and σ are image dependent measures they treat each pixel independently. The spatial correlation among colours is not taken into account. Recent work [BA91] attempts to account for the colour context by a pre-quantization step. Unfortunately, the technique does not provide a mathematical tool that is useful in the quantization process.

3 K-means algorithm and its variants.

K-means algorithm [LBG80] is a post-clustering technique that is widely used in image coding and pattern recognition. A sequence of iterations starts with some initial set $\overline{C}^{(0)}$. At each iteration t all data points $c \in C$ are assigned to one of the clusters $\overline{S_k}^{(t)}$ as defined in (3). A new center $\overline{c_k}^{(t)}$ for a cluster is computed as follows:

$$\overline{c_j}^{(t+1)} = \frac{1}{l}\sum_{i=1}^{l}(c_i | c_i \in S_j^{(t)}). \tag{7}$$

The algorithm is known to converge to a local minimum.

The K-means algorithm was used to quantize images in [WPW90]. For the test images it produced smaller average errors $\epsilon_{q(C,I)}$ than the median-cut and variance-based pre-clustering algorithms. Unfortunately, high cost of computation makes K-means impractical for image quantization.

3.1 Kohonen self-organizing maps.

A self-organizing map (SOM) is a post-clustering scheme. It was introduced by Kohonen [KKL90] as a solution to a general vector quantization problem. The SOM is a neural network that imposes a one or two-dimensional topological structure over a set of clusters in a higher dimensional space. The adaptation process attempts to approximate the density function of the input.

Dekker in [Dek94] studied the use of a one-dimensional self-organizing map for image quantization. The initial palette is set to equally spaced gray values. The input values are obtained by multiple sampling of the image with large step sizes. The closest colour $\overline{c_k}^{(t)}$ of the palette is adjusted to better comply with the input $c^{(t)}$.

The network is considered to be *elastic*. Thus, when $\overline{c_k}^{(t)}$ is updated the other $\overline{c_k}^{(t)} : |k - j| \leq r$ are also moved. The parameter r is the radius of elasticity that decreases with time. The SOM adaptation process is defined as follows:

$$\overline{c_j}^{(t+1)} = \overline{c_j}^{(t)} + \alpha_t \rho_{(t,j)} ||c^{(t)} - \overline{c_j}^{(t)}||, \tag{8}$$

where the adaptation parameter $0 < \alpha_t < 1$ is exponentially decreasing. The elasticity coefficient $\rho_{(t,j)}$ ensures that only entries in the r-neighbourhood are updated.

Since the update neighborhoods often overlap the values of $\overline{c_k}$ tend to be smoothed. In order to ensure a fair representation of colour regions by the palette $\overline{C}$ Desieno (see [HN90] p. 69) proposed the use of a special bias value $b_j^{(t)}$. The input colour $c^{(t)}$ updates the palette entry $\overline{c_k}^{(t)}$ found by the following rule:

$$||c^{(t)} - \overline{c_k}^{(t)}|| - b_k^{(t)} = \min_{j=1,2...K} ||c^{(t)} - \overline{c_j}^{(t)}|| - b_j^{(t)} \tag{9}$$

The bias factor increases for less frequently chosen vectors. Thus a colour that was chosen many times before will not to be chosen later.

In the experiments in [Dek94] the self-organizing map method produced quantized images of a better quality than octree and median-cut pre-clustering schemes. Unfortunately the SOM quantization is significantly slower than other techniques. Dekker proposed to use only a part of the image as an input data to generate the palette. This approach speeds up the palette selection but reduces quantization accuracy.

3.2 Local K-means algorithm.

In this paper we argue that a *Local K-Means* algorithm (LKM) is a suitable approach to the colour quantization problem. The method can be considered a special case of a self-organizing map. Unlike the Kohonen network, the adaptation step of the LKM process updates only the closest colour:

$$\overline{c_j}^{(t)} = \begin{cases} \overline{c_j}^{(t-1)} + \alpha_t \| c^{(t)} - \overline{c_j}^{(t)} \| & j = k; \\ \overline{c_j}^{(t-1)} & otherwise \end{cases} \tag{10}$$

The LKM is similar to gradient quantization techniques used in gray scale image coding [Mat92], [MC92]. These works prove convergence of the process to a local minima. Moreover, the gradient method converges faster than the K-means algorithm.

In the case of the Kohonen network the bias factor ensures that a distinct small colour cluster is represented by a separate palette entry [Dek94]. We found that the same result can be obtained by an improved selection of the initial palette. We have chosen to construct the initial palette by incremental insertion of a colour from the original image. A new colour is added into the palette if its distance from the already inserted entries exceeds a specified threshold.

The input data sets are constructed by sampling the image in decreasing step sizes: 1009, 757, 499, 421, 307, 239, 197,... We have chosen these step sizes to be prime numbers, thus the input sets do not intersect too much. The iteration process stops when changes to the palette $\overline{C}^{(t)}$ in a complete image scan become small. In our experience the union of input sets does not include more than 10% of all image points.

Even though the proposed algorithm examines only a portion of the input image, it is able to generate good approximating palettes. This result may be explained by the fact that colours of a typical image are clustered in the colour space. Therefore, it is enough to use a few colours from the cluster to approximate all its members. Since similar colours are often close to each other on the image surface, we hope that our input sets contain representatives for most clusters.

4 Fast nearest neighbour search.

Performance of a quantization method greatly relies on the speed of the nearest neighbour search. This search is the basis of the colour mapping operation. Moreover, the described post-clustering techniques use the nearest neighbour to determine the optimal palette.

In order to speed up the search Freidman et. al. proposed the use of k-d trees [FBF77]. In his software Poskanzer implements the search using various hash functions (see [Pos91]). Unfortunately these techniques cannot be used in the framework of iterative procedures such as local K-means. Positions of representative colours $\overline{c_j}$ are constantly changing, therefore a k-d tree or a hash table must be recomputed after every iteration.

Another approach is to use less expensive distance metric. For example, the Euclidean L_2 norm can be substituted by the less expensive L_1 norm. Unfortunately, the nearest neighbour determined by L_1 norm is not necessarily the nearest neighbour in L_2 norm.

Chaudhuri et. al. [CCW92] proposed the L_α norm as an approximation of the Euclidean metric. For a vector $x \in R^n$ the L_α norm it is defined as follows:

$$\begin{aligned} \|x\|_\alpha &= (1-\alpha)\|x\|_1 + \alpha\|x\|_\infty \\ &= (1-\alpha)\sum_{i=1}^{n} |x_i| + \alpha \max_i |x_i|. \end{aligned} \tag{11}$$

To reduce the computation cost we have chosen to use $\alpha = 1/2$. We found that the application of the $L_{\alpha=1/2}$ norm significantly speeds up the search (Table 1). Moreover, the introduced misclassifications do not noticeably influence the quality of the output image.

Table 1:
Influence of different norms on a quantization error for image "Lenna'

Norm	Time	$\epsilon_{q(C,I)}$	Wrong neighbour
L_1	11.9 sec.	5.56	11%
L_2	59.9 sec.	5.46	
$L_{\alpha=1/2}$	14.7 sec.	5.47	4%

The search cost can be further reduced using the following considerations [Hod88]:

Table 2:
Influence of different stopping rules in the nearest neighbour search on the speed of the quantization

Algorithm	Execution time	
	16 colours	256 colors
Direct search	1.74	25.51
$\sum_p$	1.45	19.44
$\sum_p$ and sorting	0.89	2.93
$\sum_p$, sorting and NND	0.76	2.76
k-d trees	1.41	2.65

- *Calculation of the partial sum.*
  Before each addition in the norm calculation (11) a partial sum Σ_p is compared with the current minimum distance Σ_{min}. The norm calculation terminates if $\Sigma_p > \Sigma_{min}$
- *Sorting on one coordinate.*
  The palette colours are sorted using one of the coordinates. Suppose that the first coordinate is chosen. The search selects palette entries in the increasing first-coordinate distance order starting with the closest colour. This process terminates when the first coordinate distance between the next palette entry and the input is larger than the current minimum Σ_{min}.
- *Nearest neighbour distance (NND).*
  The search for the nearest colour should terminate when Σ_{min} is less than one half of the distance from the current palette colour to its closest palette neighbour.

These speed optimizations were tested using the image "Lenna". We mapped the 512x400 image into 16 and 256 colour palettes. Computation time can be found in Table 2. The k-d tree colour mapping is given as a reference.

According to the experiments the performance of our colour mapping algorithm is close to that of k-d trees.

We also studied the application of the described optimization techniques to the palette selection phase of the LKM algorithm. We believe that NND rule is rather hard to use in this case as positions of centers change in each iteration step. Fortunately, the adaptation phase does not greatly rearrange a sorted order of centers. We found that it is sufficient to sort centers only at the beginning of a new sampling set without any noticeable loss of quantization accuracy.

Overall the local K-means algorithm is able to select a colour map significantly faster than the other methods (Table 3).

5 Experiments.

For our experiments we have chosen a set of 24-bit images that represent various image sources: scanned photographs, computer rendered scenes, and digitized works of art.

The local K-means procedure (LKM) was compared to implementations of the popular quantization algorithms found in public domain image processing software: median-cut [Pos91], variance-based [Tho90], octree [Cri92], SOM [Dek94]. These implementations worked in RGB colour space. For a fair comparison we also used the RGB space. Note, that even though quantization in perception-based spaces can give a better visual result, it does not change the relative correspondence of numerical values of quantization accuracy. Therefore, algorithms that produce small distortions in RGB space are expected to perform as well in Lu^*v^* or HSV spaces.

Figures 1-6 are chosen to represent two typical quantization artifacts: the loss of colour information and artificial banding.

A digitized painting by Gustav Klimt "Kiss" (Figure 1) was quantized to 16 colours. The quantized image produced by the median-cut method looks significantly distorted (Figure 2). Some fine details of the image have disappeared: blue flowers on the woman's head, yellow spots on her dress, etc. The variance-based (Figure 3) and octree (Figure 4) algorithms were able to preserve most of these details, though the original colour contrast was greatly reduced. The local K-means quantization seemed to reproduce a full chromatic range of the original image (Figure 5). In fact, the numerical values of the average distortion per colour

Table 3:
Execution time in seconds for colour map selection

Algorithm	"Kiss" 612,096 pixels		"Pool Balls" 195,330 pixels	
No. of colours:	16	256	16	256
Median-cut	4.64	4.93	1.2	1.57
Octree	2.04	3.67	0.65	0.90
Kohonen SOM		101.81		32.27
Local K-means	0.65	1.74	0.20	0.38

Table 4:
Quantization errors for image "Kiss"

Method	Max	$\epsilon_{q(C)}$	$\epsilon_{q(C,I)}$	σ
Median cut	161	29.8	20.77	14.71
Variance based	146	25.4	17.63	11.57
Octree	133	26.4	19.41	13.65
Local K-means	102	20.4	26.65	9.30

Table 5:
Quantization errors for image "Pool Balls"

Method	Max	$\epsilon_{q(C)}$	$\epsilon_{q(C,I)}$	σ
Median cut	107	8.1	4.39	2.49
Variance based	45	6.6	4.27	1.72
Octree	61	6.4	2.07	2.16
Kohonen SOM	95	7.9	1.74	2.87
Local K-means	105	7.42	2.01	2.59

$\epsilon_{q(C)}$ and deviation of distortion per pixel σ are the smallest for the LKM method (Table 4).

A computer rendered image "Pool balls" was quantized to 256 colours. All the tested algorithms were able to preserve the original colour contrast. Unfortunately, pre-clustering techniques introduced significant artificial banding (Figure 6, right column). Both LKM and Kohonen map methods were able to avoid this artifact (Figure 6, left column). It is important to notice that in the case of the SOM the palette was chosen using the entire image. The input data of the LKM algorithm covered only 8% of the original image. We found that in the case of images with large areas of close colours the average distortion per pixel $\epsilon_{q(C,I)}$ carries the most information about the quantization accuracy. The values of $\epsilon_{q(C,I)}$ are the smallest for LKM and SOM algorithm (Table 5).

6 Conclusion

The objective of our research was to develop a technique which is able to produce colour maps for quantization with minimum distortion of the original image. We presented the local K-means algorithm. This technique follows the post-clustering approach. The advantage of the LKM algorithm is the ability to select a palette without making any simplifying assumptions about the boundaries of colour clusters.

The performance of the local K-means scheme was compared to the quantization results of median-cut, octree, variance-based and SOM algorithms. The resulting images were evaluated using statistical distortion parameters as well as the perceived difference with the original. We found that the LKM technique is able to produce high quality colour maps significantly faster than other tested methods.

7 Future work

The adaptive nature of the LKM algorithm can be further explored in the future. Verevka in [Ver95] shows how the LKM method can be used to improve quantization in the windowing systems. A similar approach may be applied to quantize animation sequences.

We found that quantization may lead to reduction of the perceived colour contrast and artificial banding. Unfortunately none of the proposed statistical parameters of image distortion were able to capture these artifacts. Better numerical measures still have to be defined.

Acknowledgments

We want to thank all the people who helped us in this work. Special thanks to:

Mark Green for providing summer support to the first author;
Anthony Dekker for discussing the SOM algorithm and making his source code and test images available;
Paul Ferry for taking the time to proof read this document.

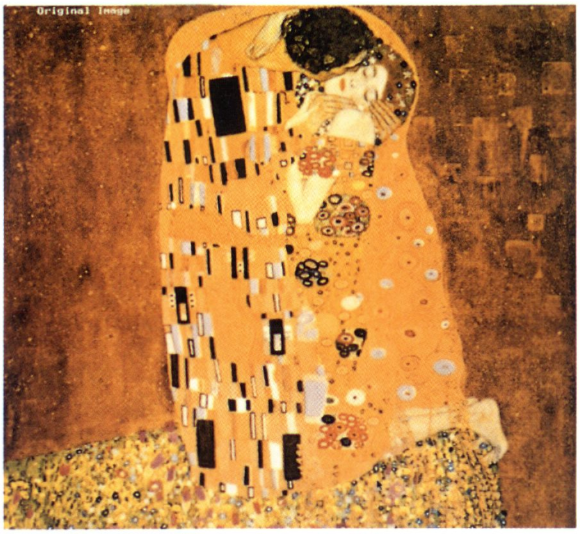

Figure 1: Kiss painting by Klint

Figure 2: Median cut

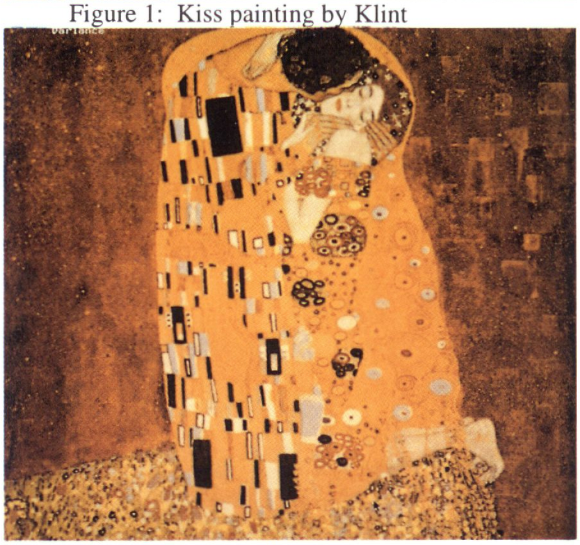

Figure 3: Variance based

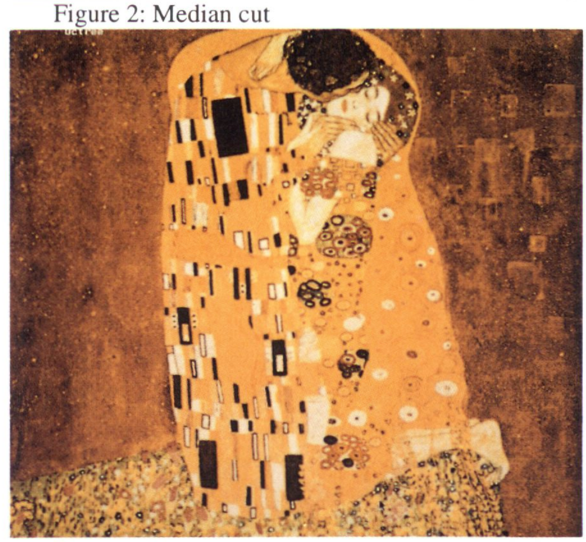

Figure 4: Octree

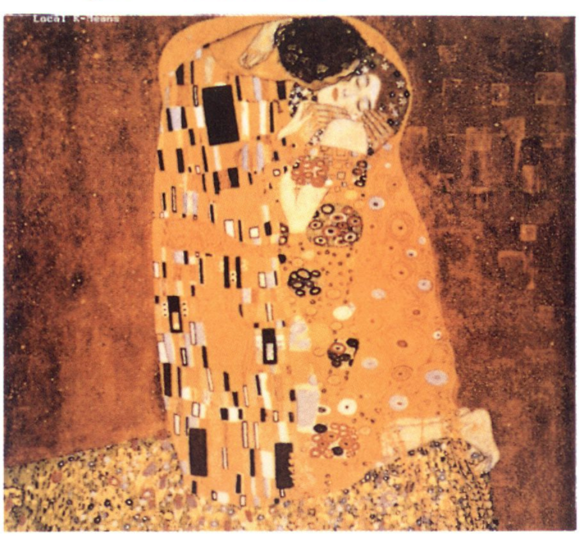

Figure 5: LKM

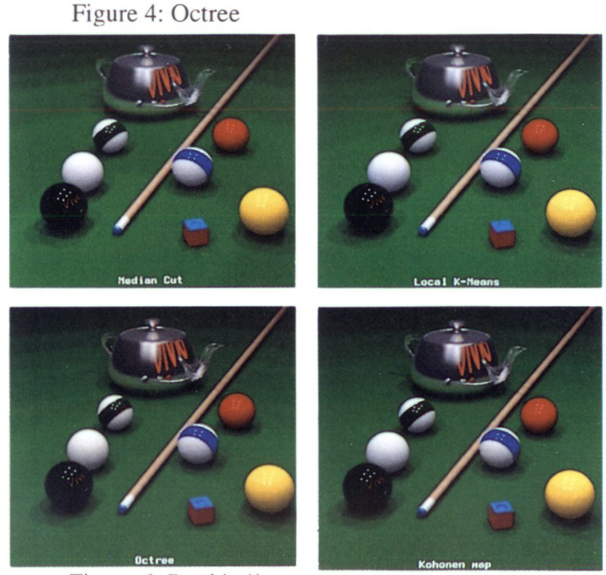

Figure 6: Pool balls

References

[BA91] R. Balasubramanian and J. Allebach. A new approach to palette selection for color images. *Imaging Technology*, 17:284–290, December 1991.

[CCW92] C. A. Chaudhuri, W. T. Chen, and J. S. Wang. A modified metric to compute distance. *Pattern Recognition*, 7(25):667–677, 1992.

[CFM93] P. Coltelli, G. Faconti, and F. Marfori. On the application of quantization and dithering techniques to the history of arts. *EUROGRAPHICS*, 12:351–362, 1993.

[Cri92] John Cristy. `quantize`, 1992. ImageMagick Image Processing Software Package.

[Dek94] Anthony Dekker. Kohonen neural networks for optimal colour quantization. *Network: Computation in Neural Systems*, 5:351–367, 1994.

[FBF77] J. H. Friedman, J. L. Bently, and R. A. Finkel. An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematical Software*, 3(3):209–226, September 1977.

[Fri93] B. Fritzke. Vector quantization with a growing and splitting elastic net. In *ICANN'93: International Conference on Artificial Neural Networks*, pages 580–585, Amsterdam, 1993. Springer.

[GP88] M. Gervautz and W. Purgathofer. A simple method for color quantization: Octree quantization. In *New Trends in Computer Graphics*, pages 219–231. Springer-Verlag, New York, NY, 1988.

[Gra84] R. M. Gray. Vector quantization. *IEEE ASSP*, 1:4–29, April 1984.

[Hec82] Paul Heckbert. Color image quantization for frame buffer display. *ACM, Computer Graphics*, pages 297–304, 1982.

[HN90] Robert Hecht-Nielsen. *Neurocomputing*. Adison-Wesley, 1990.

[Hod88] Michael H. Hodgson. Reducing the computation requirements of the minimum stance classifier. *Remote Sencing of Environment*, 25:117–128, 1988.

[KKL90] Jari A. Kangas, Teuvo A. Kohonen, and Jorma T. Laaksonen. Variants of self-organizing maps. *IEEE Transactions on Neural Networks*, 1(1):93–99, 1990.

[LBG80] Y. Linde, A. Buzo, and R. M. Gray. An algorithm for vector quantizer design. *IEEE Transactions on Communication*, 28(4):84–95, January 1980.

[Mat92] V. John Mathews. Multiplication free vector quantization using l_1 distortion measure and its variants. *IEEE Transactions on Image Processing*, 1(1):11–17, January 1992.

[MC92] V. John Mathews and S. H. Cho. Improved convergence analysis of stochastic gradient adaptive filters using the sign algorithm. *IEEE Transactions on Acustics, Speech and Signal Processing*, 35:450–454, April 1992.

[MG93] Jose L. Marroquin and Frederico Girosi. Some extensions of the K-means algorithm for image segmentation and pattern classification. Technical Report 1390, Massachusetts Institute of Technology, January 1993.

[Pos91] Jef Poskanzer. `ppmquant`, 1991. PBM+ Image Processing Software Package.

[Tho90] Spencer W. Thomas. `rlequant`, 1990. Utah Raster Kit.

[Ver95] Oleg Verevka. Color image quantization in windows systems with local K-means algorithm. *Proceedings of VI Western Computer Graphics Symposium*, pages 74–79, March 1995.

[WPW90] S. J. Wan, P Prusinkewicz, and S. K. M. Wong. Variance-based color image quantization for frame buffer display. *Color Research and Application*, 15(1):52–58, February 1990.

[Wu92] Xiofan Wu. Color quantization by dynamic programming and principal analysis. *ACM Transactions on Graphics*, 11(4):348–372, October 1992.