DTU Library

Context based Coding of Quantized Alpha Planes for Video Objects

Aghito, Shankar Manuel; Forchhammer, Søren

[Link back to DTU Orbit](#)

Context Based Coding of Quantized Alpha Planes for Video Objects

Shankar Manuel Aghito and Søren Forchhammer
Research Center COM, B.371, Technical University of Denmark, DK-2800 Lyngby, Denmark
Telephone: (+45) 4525-3641, (+45) 4525-3622
Email: sma@com.dtu.dk, sf@com.dtu.dk

Abstract— In object based video, each frame is a composition of objects that are coded separately. The composition is performed through the *alpha plane*, that represents the transparency of the object. We present an alternative to MPEG-4 for coding of alpha planes that considers their specific properties. Comparisons in terms of rate and distortion are provided, showing that the proposed coding scheme for still alpha planes is better than the algorithms for I-frames used in MPEG-4.

I. INTRODUCTION

The object based video standard MPEG-4 facilitates interactivity. Each frame of the video sequence is obtained by a composition of objects that are encoded separately. In order to compose the final frame the following information must be provided for each object: texture, shape, transparency, and position with respect to the background. Shape and transparency are contained in a gray level alpha plane.

In MPEG-4 gray level alpha planes are coded in the same way as textures, which is optimized for natural images. Alpha planes are very different from natural images: they may be computer generated, hence not affected by noise, they often contain large uniform areas, and in most cases the full range of 256 values is not used. For overall rate-distortion optimization, the fairly simple alpha planes may be coded at high quality even using a small number of bits: rates as low as 0.02 bpp are achieved in lossless coding (Table II).

A coding scheme which considers the specific properties of alpha planes was proposed by Piron and Kunt [1], [2] (Fig. 1).

Given the alpha plane $\alpha(i,j)$, the texture of the object $T(i,j)$ and the texture of the background $B(i,j)$, the composed scene $I(i,j)$ is given by

$$I(i,j) = \frac{[255-\alpha(i,j)] \times B(i,j) + \alpha(i,j) \times T(i,j)}{255}, \quad (1)$$

$$\forall (i,j) \in \{1, \ldots, N_{ROWS}\} \times \{1, \ldots, N_{COL}\}.$$

In a lossy coding scheme it is a good idea to preserve at least the values 0 and 255, because they carry important semantic information (0 defines the binary shape).

Fig. 1. General alpha plane coding scheme.

In Piron's scheme, quantization is performed directly in the spatial domain. In this way it is easy to preserve the values 0 and 255, as well as the simplicity of the alpha plane. The lossless coder used prediction of the pixel value followed by adaptive arithmetic coding of the residuals. Piron showed that the performance of this simple scheme is comparable to MPEG-4 [1], [2].

Section II describes how both the quantization and the entropy coding of Piron's scheme for still alpha planes can be improved. In Section III, our scheme is extended to include the case of sequences of alpha planes. In Section IV, a comparison of the proposed scheme for still alpha planes with Piron's scheme and MPEG-4 is given. In Section V the complexity of the proposed algorithms is discussed.

II. CODING OF STILL ALPHA PLANES

This section considers coding the frames separately, which may be desired e.g. in the line of production of video.

A. Context based lossless coders

The coder used by Piron was very simple. Experiments showed that more sophisticated algorithms like Piecewise-Constant image model (PWC) [3], Prediction with Partial Match (2D PPM) [4], Minimum Code Length Context Quantization (MCLCQ) [5], [6] are very effective on alpha planes. These schemes were designed to efficiently code palette images or digital maps, which share many characteristics with (quantized) alpha planes: absence of noise, small palette and large uniform areas. These schemes use different variations of the basic elements *binary decomposition, context quantization*, and *context based arithmetic encoding*. Table I shows that PWC outperforms Context based Adaptive Lossless Image Codec (CALIC) [7], confirming that it is not optimal to use an algorithm for natural images on alpha planes.

B. pdf-optimized scalar quantization

Piron used a uniform quantizer, with small modifications in order to preserve 0 and 255. Our tests showed that rate-distortion performances are improved using a quantizer that minimizes the distortion given the number of reconstruction levels N_{rl} (Figs. 2 and 3). The optimization algorithm proposed by Lloyd [8] finds a local minimum dependent on the initial set of reconstruction values. Hence, we used a *merge − split* algorithm in order to get a good initial set.

sequence, frames	weather, 2-300	logo, 2-300	rain, 2-169
PWC	1018	825	3756
2D PPM [5]	1233	523	4062
CALIC	1125	2555	3732
Quant.[2] + PWC	433	695	1308
Quant.[2] + CALIC	1041	2555	3710

[1] Rates expressed in [KByte] = 1024 [Byte].
[2] Quantization as described in Section II-B, $N_{rl} = 16$.

Let $\{y_i\}_{i=1}^{N_{rl}}$ be the set of recontruction levels and $\{b_i\}_{i=0}^{N_{rl}}$ the decision boundaries. Modelling the alpha values as a source with probability density function $f_X(x)$, the well known conditions for a minimum are

$$y_j = \frac{\int_{b_{j-1}}^{b_j} x f_X(x) dx}{\int_{b_{j-1}}^{b_j} f_X(x) dx} \quad (2)$$

$$b_j = \frac{y_{j+1} + y_j}{2} \quad (3)$$

MERGE-SPLIT INITIALIZATION

1) Set $y_1 = 0$, $b_0 = 0$. Distribute $\{y_i\}_{i=2}^{N_{rl}}$ uniformly in $[1, M]$, M is the maximum value in the original alpha plane. Update the decision boundaries $\{b_i\}_{i=1}^{N_{rl}}$ with (3).
2) Compute the distortion associated with each decision interval $D^{(i)} = \int_{b_{i-1}}^{b_i} (x - y_i)^2 f_X(x) dx$, $\forall i \in \{2, \ldots, N_{rl}\}$, the global distortion $D_A = \sum_{i=1}^{M} D^{(i)}$.
3) Delete the interval associated with the minimum $D^{(i)}$. Split the interval associated with the maximum $D^{(i)}$ in two parts. Update the reconstruction levels associated with the new intervals using (2).
4) Recompute the global distortion D_B.

    if $D_B < D_A$
    then Go back to Step 2
    else Undo the modifications made in Step 3 and exit.

LLOYD ALGORITHM

1) Start from an initial set $\{y_i^{(0)}\}_{i=1}^{N_{rl}}$. Set $k = 0$, $D^{(0)} = 0$, $b_0 = 0$, and select the threshold ϵ.
2) Find all the decision boundaries $\{b_i^{(k)}\}_{i=1}^{N_{rl}}$ using (3)
3) Compute the distortion $D^{(k)}$.

    if $|D^{(k)} - D^{(k-1)}| < \epsilon$
    then $\{b_i^{(k)}\}_{i=0}^{N_{rl}}$ and $\{y_i^{(k)}\}_{i=1}^{N_{rl}}$ define the desired optimal quantizer
    else $\{b_i^{(k)}\}_{i=1}^{N_{rl}} \rightarrow \{y_i^{(k+1)}\}_{i=2}^{N_{rl}}$ using (3)
            Set $k = k + 1$
            Go back to Step 2.

Hereafter we refer to the combination of the two algorithms described above as Lloyd. Figs. 2 and 3 show a comparison
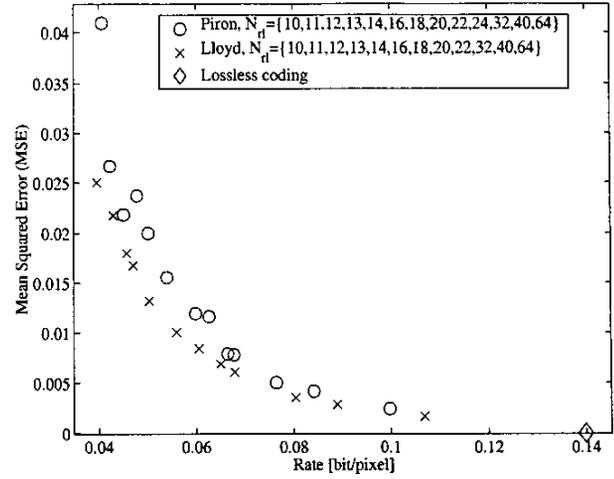

Fig. 2. Lloyd and Piron quantization on the first frame of the sequence "Akiyo". MSE was evaluated on the alpha plane. Rates were evaluated using PWC.
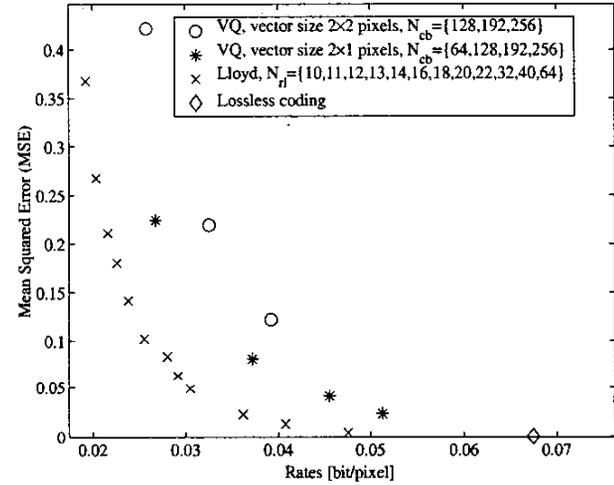

Fig. 3. Lloyd and vector quantization (VQ) on the first frame of the sequence "Weather". MSE was evaluated on the composed image. Rates were evaluated using PWC. N_{cb} is the size of the codebook.

between Lloyd quantization, Piron quantization, and vector quantization (codebook optimized with Pairwise Nearest Neighbor [9] followed by LBG [10], ordered according to the average value of the codewords). Entropy coding was evaluated using PWC.

III. CODING OF SEQUENCES OF ALPHA PLANES

As a first step towards utilizing the temporal redundancy in a sequence of alpha planes, we present some techniques based on the statistics of the previous frames, not on actual pixel values. The algorithms in Sections III-A and III-B were implemented and tested, while Sections III-C and III-D present some possible improvements.

A. Adaptive grouping of frames for the quantization

Both the *statistic initialization* (Section III-B) and the *split binary decomposition* (Section III-D) reduce the rate if the quantized coding frame and the previous quantized frame use the same $\{y_i\}_{i=1}^{N_{rl}}$. Hence, in order to minimize the rate, the same quantizer should be used for consecutive frames. On the other hand, in order to minimize the distortion of the quantized sequence, the quantizer should be optimized separately for each frame. Hence the best solution is given by a trade-off between these two features. We propose an algorithm that adaptively groups consecutive frames to be quantized together, in order to reduce the rate while limiting the increase of distortion.

ADAPTIVE GROUPING

1) Set $i = 1$, $f = 1$, choose the tolerance factor β. Quantize frame 1 optimizing the quantizer on frame 1 itself.
2) Set $i = i + 1$. Compute $D_{i,f}$, the distortion that we get if we quantize frame i with the quantizer optimized for frame f. Compute $D_{i,i}$, the distortion that we get if we quantize frame i optimizing the quantizer on the frame itself.
3) if $\quad D_{i,f} > (1 + \beta/100)D_{i,i}$

    then the increase of distortion is too high, so quantize frame i itself, as the first frame of a new group (set $f = i$)

    else quantize frame i with the quantizer optimized for frame f.

Go back to Step 2.

Using an adaptive algorithm is of course a good choice in terms of robustness, e.g. at scene cuts. Even when tested on video sequences whithout scene cuts, adaptive grouping of frames gave a slightly improved rate-distortion performance compared to the nonadaptive version of the algorithm (fixed length of the groups).

Figure 4 shows the increased coding efficiency of adaptive grouping followed by statistic initialization compared to intraframe coding. It also shows that when there is the freedom of doing it, it is always convenient to use large groups (large β), because the rate is reduced while the increase of the distortion is neglectable.

B. Statistic initialization

A common problem with all adaptive context based arithmetic coders is that the coding is not efficient in the beginning of the process, when little statistics have been gathered. Coding a sequence of alpha planes, a solution is to start coding each frame using the statistics gathered in the previous frame. An approximation of the performance achievable with this method can be obtained without implementing a new coder, concatenating the previous frame A on the top of the coding frame B. If $R(AB)$ is the rate obtained coding the image composed in this way, and $R(A)$ is the rate obtained coding A, then $R_{init}(B) = R(AB) - R(A)$ is an approximation of
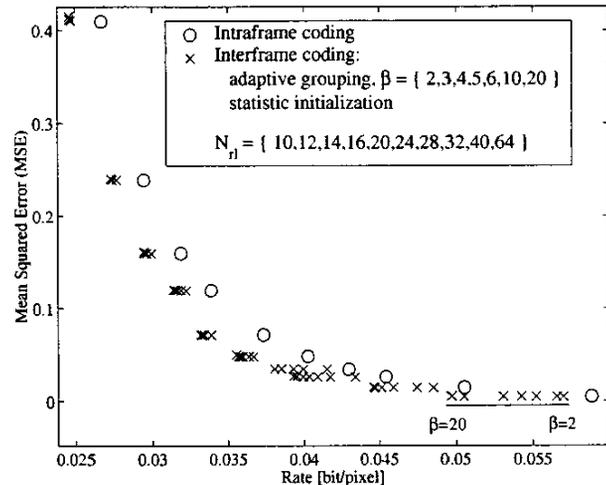

Fig. 4. Rate-distortion curve of Lloyd+PWC on the sequence "weather". Lossless coding is performed at 0.080 bpp in intra mode, 0.060 in inter mode.

TABLE II
LOSSLESS RATES[1] OBTAINED WITH STATISTIC INITIALIZATION.
PERCENTAGES IN COMPARISON WITH INTRAFRAME CODING (TABLE I).

sequence, frames	weather, 2-300	logo, 2-300	rain, 2-169
PWC	768 (75%)	774 (94%)	3593 (96%)
2D PPM [5]	894 (73%)	329 (63%)	3350 (82%)
MCLCQ [5]	760	268	3200

[1] Rates expressed in [KByte] = 1024 [Byte].

the code length of B initializing the statistic on A. Table II shows the results obtained using this procedure for PWC and 2D PPM, while MCLCQ implicitly uses the statistics of the previous frame.

C. Interframe coding using split template

A better coding context for each pixel P in the current frame can be obtained if we split the template in two parts. In addition to the typical group of causal neighbors of P, the template may also include the neighbors of P', the pixel in the previous frame aligned with P according to the motion vectors. The neighbors of P' are not restricted to be causal, because the previous frame is known by the decoder. This technique is successfully used by CAE, the context based arithmetic coder used in MPEG-4 for *binary* alpha planes. It may be extended to the coding of gray level alpha planes.

D. Split binary decomposition

PWC and MCLCQ evaluate the coding pixel value by a succession of binary questions like: "is P equal to the pixel W on its west?" If the answer is yes, the pixel can be encoded very efficiently, and this happens very often in alpha planes because of the the large uniform areas. It is straight forward to extend this method including a question like "is P equal to P'?"
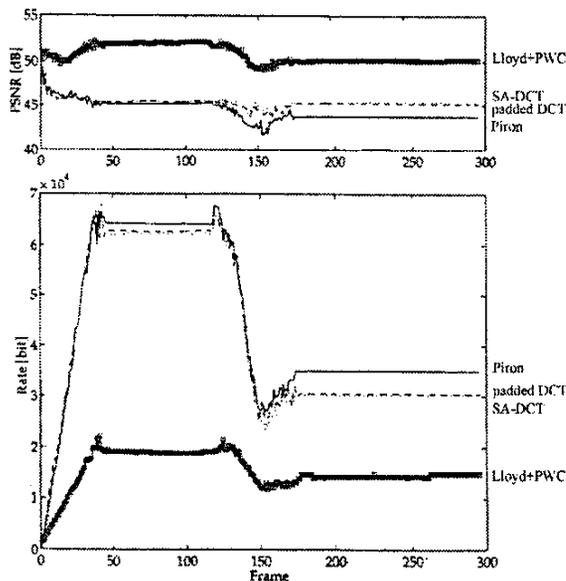
103

Fig. 5. Sequence "logo" (352 × 288 pixels): rate-distortion performances.

Fig. 6. Sequence "rain" (352 × 288 pixels): rate-distortion performances.

IV. COMPARISON WITH PIRON AND MPEG-4

We present a performance comparison of the proposed algorithm (Lloyd+PWC) for still alpha planes with the method proposed by Piron, and with two algorithms for I-frames included in the video verification model of MPEG-4: shape adaptive DCT (SA-DCT) and padded DCT [1], [2]. The Peak-Signal-to-Noise-Ratio is computed by

$$PSNR = \frac{255^2 \sum_{i=1}^{352} \sum_{j=1}^{288} S(i,j)}{\sum_{i=1}^{352} \sum_{j=1}^{288} S(i,j)(I_{or}(i,j) - I_q(i,j))^2} \quad (4)$$

where $S(i,j) = 0$ if $\alpha(i,j) = 0$, and 1 otherwise. $I_{or}(i,j)$ and $I_q(i,j)$ are obtained with (1) using the original alpha plane and the quantized one (Lloyd, $N_{rl} = 12$, optimized separately for each frame) respectively. Some results are shown in Figs. 5 and 6. CIF sequences downsampled from the full size sequences as described in [11] were used to evaluate Lloyd+PWC, while results for the other algorithms were taken from [1].

On the sequence "logo" ("children", layer 2) Lloyd+PWC outperforms the other coding schemes: it gives a higher PSNR, by about 5 dB, for less than half the bitrate. Lloyd+PWC appeared to be the best also on the sequences "weather" and "explosion" ("destruction", layer 6). SA-DCT is better than Lloyd+PWC only on "rain" ("destruction", layer 8), because this sequence is a "less typical" alpha plane, having more similarity with a natural image. Still the proposed scheme is better than padded DCT and Piron's scheme (PSNR is more than 10 dB higher for approximately the same bitrate).

V. SPEED OPTIMIZATION

Speed optimization is not a primary objective in this paper, but it has been shown that PWC is a quite fast algorithm, with encoding time comparable to GIF and PNG [3]. A fast implementation of the adaptive grouping algorithm presented
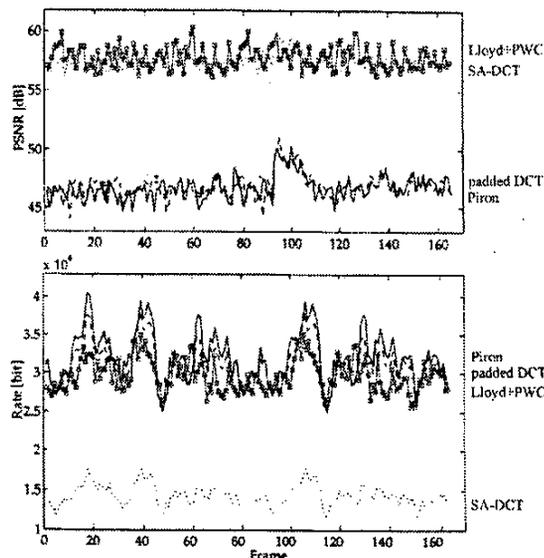
in Section III-A can be obtained computing the distortions $D_{i,i}$ and $D_{i,f}$ directly from the histograms of frame i and f.

VI. CONCLUSIONS

The coding scheme for still alpha planes proposed by Piron was considerably improved. Our scheme is also more efficient than the algorithms for I-frames included in MPEG-4. PWC was preferred as the lossless coder for practical reasons, although the more complex PPM and MCLCQ give smaller rates, so the performance can be further improved. Some ideas for utilization of temporal redundancy in sequences of alpha planes were proposed.

REFERENCES

[1] L. Piron, *Representing Transparency and Shape in Alpha Planes: Coding and Image Authentication.* Lausanne: Ph.D. thesis, Department of Communication Systems, EPFL, 1999.

[2] L. Piron and M. Kunt, "Differential coding of alpha planes with adaptive quantization," *ITG-Fachberichte*, vol. 143, pp. 713–18, 1997.

[3] J. Ausbeck, P.J., "The piecewise-constant image model," *Proceedings of the IEEE*, vol. 88, no. 11, pp. 1779–89, 2000.

[4] S. Forchhammer and J. Salinas, "Progressive coding of palette images and digital maps," *Proc. of Data Compression Conference*, pp. 362–371, 2002.

[5] S. Forchhammer, X. Wu, and J. Andersen, "Lossless compression of image data sequences using optimal context quantization," *unpublished*.

[6] ——, "Lossless image data sequence compression using optimal context quantization," *Proc. of Data Compression Conference*, pp. 53–62, 2001.

[7] X. Wu and N. Memon, "CALIC - a context based adaptive lossless image codec," *Conference Proceedings ICASSP-96*, vol. 4, pp. 1890–1893, 1996.

[8] S. Lloyd, "Least squares quantization in pcm," *IEEE Trans. on Inform. Theory*, vol. 28, pp. 127–135, 1982.

[9] W. Equitz, "A new vector quantization clustering algorithm," *IEEE Trans. on Acoust., Speech, Signal Processing*, vol. 37, no. 10, pp. 1568–75, 1989.

[10] Y. Linde, A. Buzo, and R. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. 28, no. 1, pp. 84–95, 1980.

[11] MPEG-4. N3908, "MPEG-4 video verification model version 18.0," *ISO/IEC JTC1/SC29/WG11*, 2001.

104