l-Infinity Constrained Approximations for Image and Video Compression

Marco Dalai and Riccardo Leonardi

Universitá degli Studi di Brescia
Dipartimento di Elettronica per l'Automazione
via Branze, 38 - I-25123 Brescia, Italy
Email: {marco.dalai, riccardo.leonardi}@ing.unibs.it

Abstract. In this paper we focus on image and video coding with an l-infinity norm criterion. From a general point of view, we discuss the problem by considering segmentation and piecewise approximations in linear spaces when dealing with the l-infinity norm. We apply then the main ideas to the case of image compression showing some practical techniques which represent the first attempts to adapt second generation image coding approaches under the l-infinity norm. Based on this setting we consider the problem of video compression and we propose some frameworks to the more complex problem of spatial-temporal segmentation for second generation video coding under l-infinity norm constraints.

1 Introduction

When considering the topic of lossy image and video coding almost all the available literature refers to the case of a weighted mean square error distortion criterion, i.e. the loss in the compressed data with respect to the original one is measured by taking the average squared error. In some situations it is however important to preserve the quality of the reconstructed signal at every point and not just in a mean square sense. In these cases it is thus necessary to consider lossy compression algorithms designed so as to be optimized in an l_∞ sense. Only very few attempt have been made so far in this direction. Of notable importance is the particular case of near lossless coding, for which effective techniques have been developed which use predictive based or transform based approaches ([1, 2]) while a second generation approach has been proposed for the wider error range problem in [3, 4].

In [3], the image was recursively partitioned into rectangular regions by means of a binary tree decomposition, and bilinear surfaces were used as approximations of the original signal within each region. In [4] an improvement over the method of [3] was proposed where neighboring regions of similar content are jointly coded.

With respect to [3, 4], in this paper we aim at providing a more general framework to the problem of segmentations and piecewise approximations in linear spaces under the l_∞ norm criterion and its application to image and video coding. The previously published works of [3] and [4] are given a more general and comprehensive context.

The content of the paper is as follows. In the next section we discuss the theoretical aspects of l_∞ approximations in linear spaces and we consider the problem of segmentation within this context. In Section 3 we focus on applications to image coding and we discuss some examples of segmentations and piecewise approximations for this particular case. Finally, in Section 4 we consider the more difficult problem of segmentations and approximations for image sequences, analyzing the main difficulties and giving some hints for future directions.

2 Approximations and Segmentations

Given a discrete domain of n points $D = \{p_1, p_2, \ldots, p_n\}$, we consider a digital signal s as a function from D to the set of real numbers. Given a set $B = \{b_1, b_2, \ldots, b_m\}$ of m basis functions from the set D to $\mathbb{R}$, call V the linear space spanned by B, i.e.

$$V = \left\{ \sum_{i=1}^{m} c_i b_i,\ c_i \in \mathbb{R}\ \forall\, i \right\} \tag{1}$$

The l_∞ distortion between the signal s and a function $v \in V$ is defined as

$$d(s, v) = \max_{p \in D} |s(p) - v(p)| \tag{2}$$

Given a signal s, we are interested in finding a function $v \in V$ which is a sufficiently good approximation of s, say $d(s, v) \le \delta$, or, in some cases, we are interested in finding the best approximation of s in V. Consider this latter case where we want to find the best approximation; this problem can be reformulated as the problem of minimizing, for $v \in V$, the value of e subject to the constraints $|s(p_i) - v(p_i)| \leq e$, $i = 1, \ldots, n$. Given the linear structure of V, this corresponds to minimizing, for $(c_1, c_2, \ldots, c_m) \in \mathbb{R}^m$, the value of e under the constraints

$$\left| s(p_i) - \sum_{j=1}^{m} c_j b_j(p_i) \right| \leq e, \quad i = 1, \ldots, n. \tag{3}$$

By considering the vector of unknown $(c_1, \ldots, c_m, e) \in \mathbb{R}^{m+1}$ the problem is thus equivalent to a linear program in $(m+1)$ dimensions with $2n$ constraints (2 constraints for every point p_i in equation (3)). So, many known algorithms for linear programming can be applied in order to find the optimal approximation (see [5] for a detailed technical and historical discussion). A classic algorithm can be found in [6] while more recent randomized algorithms with $O(n)$ expected time complexity were given in [7, 8].

By considering the interpretation of the problem as a linear program, an important observation can be made on a particular characteristic of the obtained optimal approximation. It is possible to show in fact, under some rather weak hypothesis (see [5]), that the best l_∞ approximation of a signal s in a linear space of dimension m is completely determined by a particular subset of $m+1$ points of D; these m points $q_1, q_2, \ldots, q_{m+1}$, which we call *pivot points*, are such that the optimal approximation v^* gives an approximation error which takes its highest absolute value on the point q_i, $i = 1, \cdots, m+1$. This fact is of great importance when piecewise approximation are to be considered, and we will come back to this point soon when discussing segmentations.

Note that no hypothesis was made on the domain D up to now, in the sense that we have not specified if the discrete points p_i are one-dimensional or multidimensional points, i.e. D is a subset of $\mathbb{R}$ or $\mathbb{R}^d$. Indeed, the above discussion only consider the values of $s(p_i)$ and $b_j(p_i)$, for $j = 1, \ldots, m$ and $i = 1, \ldots, n$. The obtained method for finding the optimal approximation is thus very general and can be applied without changes to one-dimensional or multidimensional domains D. Anyway, for some particular problems less general but more efficient algorithms can be used. For the simple case of straight line approximations in one dimension, an efficient geometric algorithm was proposed in [9]. In this case the pivot points are identified to be three particular vertices of the convex hull of the points $s(p_i)$ as shown in Fig. 1. Two pivot points, namely A and B in Fig. 1, are extremities of a side which gives the slope of the optimal approximation (i.e. r), while the other pivot point C is such that it is vertically projected inside side AB and it determines the approximation error. A similar situation can be argued for the case of planar approximations in two-dimensional domains but without the computational benefits obtained for the one-dimensional case. In this case the four pivot points are again vertices of the convex hull of $s(p_i)$ and they again have a clear geometrical meaning, but but they can be positioned in two different configurations, see Fig 2. Finally, for the case of bilinear approximations over two dimensional rectangular domains a fast suboptimal solution was proposed in [3] which is based on a separable application of the one-dimensional linear case.

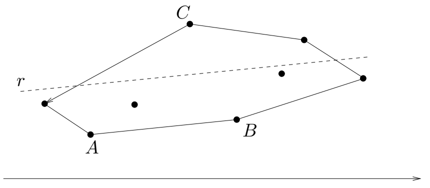
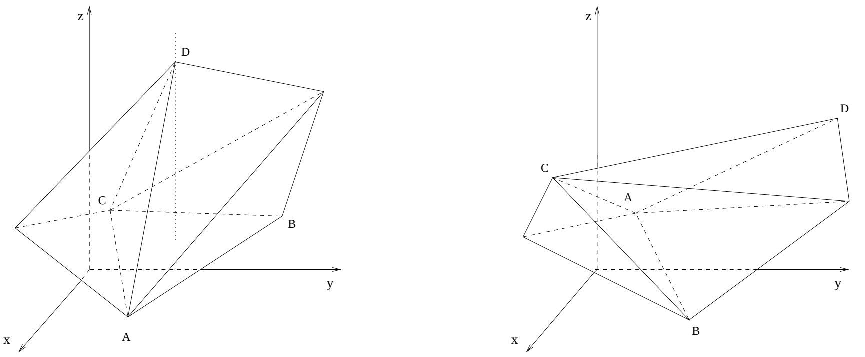


Fig. 1. Straight line approximation of one-dimensional signals. Vertices A, B and C are the pivot points in this case.

Stated in the above form, the problem of l_∞ approximation in linear spaces is easily solved. In the case of image and video processing, however, one is usually interested in using a relatively small number m of basis functions (for complexity reasons), so that the obtainable approximations cannot describe the given data with sufficient detail in all the domain points. So, it is necessary to segment the domain into subregions and try to approximate the signal within each one. As opposite to what stated for the approximation technique, here the dimensions of the domain D plays a fundamental role. The higher the dimensionality of D and the more complex the segmentation problem becomes. For the case of one-dimensional signals an exhaustive treatment of the problem of optimal segmentations for general linear spaces has been proposed in [10] while for bidimensional signals only empirically good solutions for piecewise bilinear approximations have been proposed in [3, 4].

From a high level point of view we can adopt an approach based on the important class of hierarchical segmentations; the signal is approximated first



(a) Pyramidal configuration

(b) Two edges configuration

Fig. 2. Two possible configurations for the pivot points (A, B, C, D) of a linear approximation in a bidimensional space. In the first case the pivot points are disposed in a "pyramidal" configuration, i.e point D is vertically projected into the ABC face. In the second case instead no pivot point is vertically projected into the face represented by the remaining pivot points.

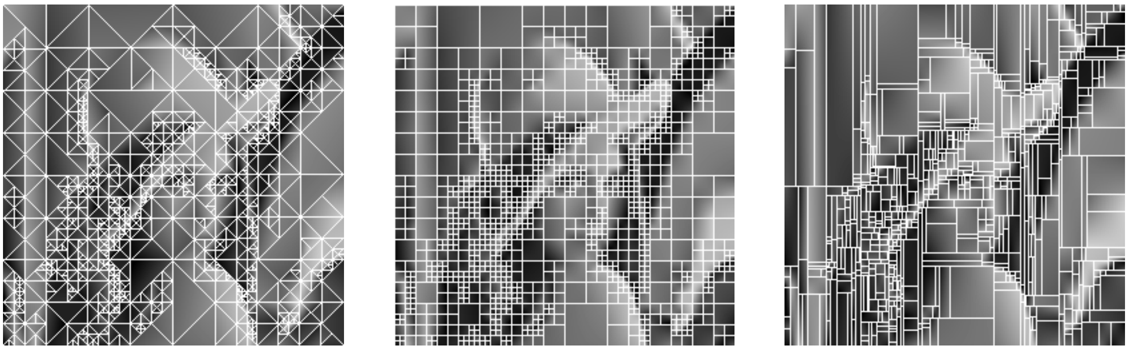
on the whole domain and, if the approximation error is higher than a given threshold, the domain is partitioned and the procedure is recursively applied to every subdomain. When using the l_∞ norm within this type of segmentations, the properties of the optimal approximations computed at every step must be taken into account in order to partition the domain in an efficient way. It is very important in fact to consider that the pivot points are the points where the maximal error occurs and that the solution is uniquely determined by those pivot points. So, in order to decrease the approximation error from one iteration to the next, it is necessary to split the domain in such a way that the pivot points are left in different subdomains. For example, for the case of linear approximations as in Fig. 1, if r is not a sufficiently good approximation, points A and B must be divided by the segmentation process, and the partition point should be placed in C.

3 Applications to Image Coding

In this section we aim at giving an overview of different l_∞ approximations and segmentation techniques used for image coding. In the case of image coding we must consider some important additional facts with respect to the theoretical discussion of the previous section. In fact, in order to use piecewise approximations for the sake of image coding, it is important to consider with care the two fundamental entities involved in the representation, namely the segmentation information and the approximation information in every region. For both of them also the quantization problem must be taken into account and, as usual in second generation image coding, the more complex the segmentation is, the smaller the number of regions. It is then usually necessary to find a trade-off between the complexity of the segmentation and the number of regions obtained, which is a trade-off between the coding rate associated to the partition and the coding rate associated to the approximations within the regions.

The problem of coding the segmentation is not different here from the case of l_2 approximations, so that we can focus here on the problem of coding the approximations found in every region. Let us consider here the problem in the case where there is no restriction on the region shape and we are working in a linear space of dimenstion m. Note that the optimal approximation v^* in every region is identified by the sequence of m coefficients c_i, which are real numbers in the general case. So, in order to code the solution by using the c_i coefficients it is necessary to quantize them. In the quantization phase an additional error is introduced and it is not easy in general to perform the quantization of the coefficients in such a way that l_∞ error is kept under control.

A different approach can instead be used, which allows to perform a lossless coding of the optimal approximation. Again, the solution is given by the pivot points. As the optimal solution is completely determined by the pivot points, and as these pivot points are samples of the image, it is possible to losslessly encode the optimal solution by simply specifying these points and their values. So, for a typical image at 8 bits per pixel, the approximation



(a) Linear approximations on triangles

(b) Bilinear approximations on QuadTree

(c) Bilinear approximations on RectTree ([3])

Fig. 3. Examples of segmentation with associated approximations.

relative to a region of n points can be coded with $m(\log_2 n + 8)$ bits, if no entropy coding is applied.

A further different approach to the problem of coding the approximations is obtained if one restricts the attention to some special cases of combinations between the approximating functions and the type of partitions to be used. For example, in the case of planar approximations, if one divides the image into triangular regions, it is possible to code the approximation in every region by quantizing its values on the three vertices of the domain. This way it is very easy to keep under control the error due to quantization. The same thing holds for the case of bilinear approximations if the used domains are of rectangular shape.

As examples of the possible choices obtained following the above discussion we show in Fig. 3 two non-adaptive segmentations and an adaptive one used in order to approximate the image Lena with an error threshold of 16. In all cases the hierarchical segmentation process is iterated until an error not exceeding the threshold is obtained or, in alternative, until the obtained regions are so small that there is no gain in further using the segmentation approach, in which case the values of the pixels are just listed one by one.

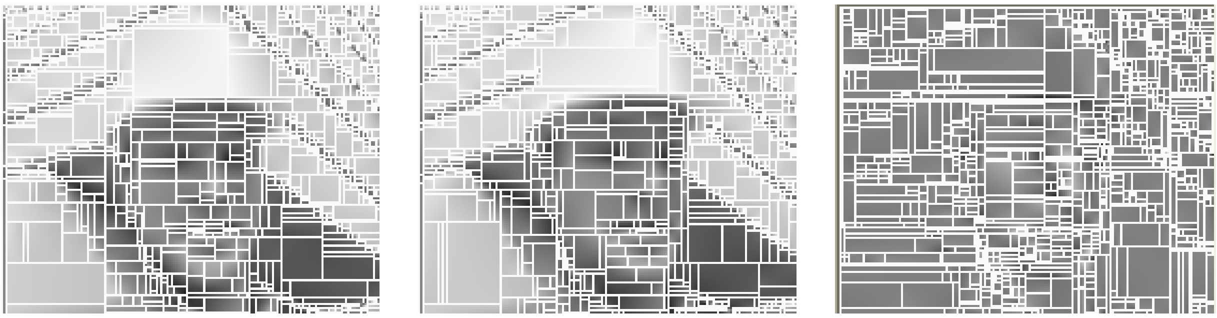
The first image shows a regular dyadic partition in triangular regions with linear approximations within each region. Here the information related to the partition is limited to a very small number of bits (1 bit for every region) and the surface within every region is encoded with 8 bits for the value in every one of the three vertices. A similar thing holds for the second type of approximation which represents quadtree decomposition with bilinear approximations. The segmentation information here has a cost of 1/3 of a bit for every region while the bilinear surface is encoded with 8 bits for every corner. The last approximation, instead, uses bilinear approximations combined with an adaptive segmentation; here the cost of the segmentation is higher (about 8 bits for every region) but the number of regions is much smaller, so that the code associated to the surfaces is lower.

4 Toward Video Coding

Building upon the previously discussed framework, we consider now the problem of video coding. Here we aim at giving a first insight to possible applications of the above discussed techniques and at proposing some possible extensions for including the temporal dimension into the coding process.

As already clarified in Section 2, first note that the problem of finding the optimal approximation in a linear space can be solved exactly in the same way in the three dimensional space of a video sequence. The main difficulty relies instead in segmenting the domain. From a theoretical point of view it is possible to operate in a similar way to what was explained for images by extending some of the partition techniques to the three dimensional case. For example, it is possible to perform an *octtree* decomposition which is an immediate extension of the quadtree. In the case of video anyway, it is even more important to adopt adaptive techniques.

With respect to this point, a possible extension of the rectangular tiling with bilinear approximations can be constructed. The separable approach used in [3] can be extended to the three dimensional space, and can thus be used in order to build a trilinear approximation. Here an outline of the procedure is



(a) Frame 1

(b) Frame 2

(c) Prediction error on Frame 2 (shifted up by 128 levels)

Fig. 4. Segmentations obtained for the first two frames of the foreman sequence (QCIF@15fps) with error threshold set to 16.

presented. Given the signal $s(x, y, t)$ defined on the domain

$$D = \{(x, y, t) : x_0 \leq x \leq x_1, y_0 \leq y \leq y_1, t_0 \leq t \leq t_1\}$$

we aim at finding a function f of the form

$$f(x, y, t) = axy + bxt + cyt + dx + ey + ft + g,$$

which is a good approximation under the l_∞ norm. Following the idea used in [3] for the two dimensional case, we first fix two values $\bar{t}$ and $\bar{y}$ and then perform linear approximation of the points $s(x, \bar{y}, \bar{t})$, i.e. along the x dimension. We thus obtain a segment whose extremities, positioned on $x = x_0$ and $x = x_1$, can be denoted by $q_0(\bar{y}, \bar{t})$ and $q_1(\bar{y}, \bar{t})$ respectively. Then, we let y move and we approximate the points $q_0(y, \bar{t})$ with a segment whose extremities are $h_{0,0}(\bar{t})$ and $h_{0,1}(\bar{t})$; afterward, we do the same with the points $q_1(y, \bar{t})$ obtaining $h_{1,0}(\bar{t})$ and $h_{1,1}(\bar{t})$. Finally, we let t move and we approximate the points $h_{0,0}(t)$ with a segment whose extremities are called $\tilde{f}_{0,0,0}$ and $\tilde{f}_{0,0,1}$; doing the same for the remaining $h_{\cdot,\cdot}(t)$ points, we obtain all the values $\tilde{f}_{i,j,k}$ for $(i, j, k) \in \{0, 1\}^3$. These are the eight values of the trilinear approximation in the eight vertices of the cuboid. At every step the approximations are constructed so as to minimize a pessimistic estimate of the actual l_∞ error, exactly in the same way as explained in [3] for the two-dimensional case. So, the obtained approximation is only sub-optimal and not optimal. However the construction is very fast and it gives important information about the internal structure of the one-dimensional behaviour of the signal, which is very useful for the sake of segmentation.

This way, the video sequence can be iteratively divided into smaller cuboids until the error of the trilinear approximation within each region is smaller than the threshold δ.

The main problem of the technique is that it is not possible to take advantage of the motion, due to the fixed directions of the cutting planes. So, as video sequences are usually highly undersampled in the temporal direction, this leads to a very high number of regions. In order to take into account the motion in the coding of the video, a first attempt can be made by working frame by frame and using motion compensation. An idea is first to performe a two-dimensional segmentation on the frames, and then use motion compensated prediction in order to encode the segmentation and the texture information of each frame given the previous one. In Fig. 4(a) and 4(b) we show the segmentations with bilinear approximations obtained for the first two frames of the foreman sequence. In Fig. 4(c) instead, we show the segmentation obtained for the approximation of the prediction error of the motion compensation between the original second frame and the approximated first one. By comparing Fig. 4(b) and 4(c), we note that the number of obtained region is not fundamentally different in the two cases, but that there is instead a difference in the position where regions get finer. In Fig 4(c) in fact, the regions get smaller where there is high motion (the face of the man), while they are wider where there is low motion (top left portion of the frame). So, a possible strategy for the encoding of the second frame is given by mixing motion compensation and segmentation in different order, depending on the level of motion of different regions. Where the motion is relevant, it is better to use segmentation on the original frame and use motion compensation in order to predict the texture of the obtained regions.

In case of low motion instead, it is better first to compensate the frame and then to apply the segmentation based coding on the residual error. These considerations are being used in an ongoing work on video coding.

5 Conclusion

In this paper we have presented an overview of the main issues involved in the problem of signal approximation and segmentation under an l_∞ norm constraint. We have considered some applications to image coding showing different examples of segmentation and approximation, thus strengthening the basis of previously presented works. Finally, we have proposed some first attempts to the problem of video coding, giving some guidelines for future directions within this context.

References

1. Karry, L., Duhamel, P., Rioul, O.: Image coding with an L^∞ norm and confidence interval criteria. IEEE Trans. Image Proc. **7** (1998) 621–631
2. Alecu, A., Munteanu, A., Schelkens, P., Cornelis, J., Dewitte, S.: Wavelet-based fixed and embedded L-infinite-constrained coding. Proc. IEEE Int. Conf. Digital Signal Processing (2002) 509–512 Santorini, Greece.
3. Dalai, M., Leonardi, R.: l^∞ norm based second generation image coding. In Proc. of ICIP'04 (2004) 3193–3196
4. Dalai, M., Leonardi, R.: Segmentation based image coding with l-infinity norm error control. Picture Coding Symposium 2004 (2004)
5. Watson, G.A.: Approximation in normed linear spaces. J. Comput. Appl. Math. **121** (2000) 1–36
6. Borrodale, I., Phillips, C.: Solution of an overdetermined system of linear equations in the Chebyshev norm [F4] (algorithm 495). ACM Trans. Math. Software **1** (1975) 264–270
7. Megiddo, N.: Linear programming in linear time when the dimension is fixed. J. ACM **12** (1984) 114–127
8. Seidel, R.: Small-dimensional linear programming and convex hulls made easy. Discrete Comput. Geom. **6** (1991) 593–613
9. Dalai, M., Leonardi, R.: Efficient (piecewise) linear minmax approximation of digital signals. Proc. of ICASSP'04 (2004) 585–588
10. Dalai, M., Leonardi, R.: Approximations of one-dimensional digital signals under the l^∞ norm. Accepted for pubblication in the Trans. on Signal Proc. (2005)