ANALYSIS OF MOJETTE TRANSFORM PROJECTIONS FOR AN EFFICIENT CODING.

Pierre Verbert, Vincent Ricordel and Jeanpierre Guédon

Image & Video Communications Team, IRCCyN UMR CNRS 6597
École polytechnique de l'Université de Nantes
La Chantrerie, BP 50609, 44306 NANTES CEDEX 3
{pierre.verbert, vincent.ricordel, jean-pierre.guedon}@polytech.univ-nantes.fr

ABSTRACT

We investigate the analysis of Mojette transform projections in an image coding context. A strategy for choosing projections elements (denoted as bins) is introduced in order to obtain compact bin-streams. In a first part the Mojette transform is presented with its direct and inverse algorithms. In a second part these basis algorithms are upgraded, to select the useful bins for image reconstruction, and to get well-defined streams for projection coding. The rules for this selection are detailed. Results of real world image coding will be presented.

1. INTRODUCTION

The Mojette transform has already been used in the context of transmission and storage of multimedia contents. Briefly speaking, this transform is based on the projection of digital data on redundant discrete hyperplanes. This allows formultiple description [1][2], joint source-channel coding techniques and storage area network. Our study aims at optimizing the streams issued from the Mojette transform of a digital image. The proposed scheme is a control tool, presenting some decision rules on the elements of the projections (denoted as bins). The purpose of these rules is to limit the amount of data to be transmitted, and to find the best layout of the bins inside the projections: compact bins-streams are expected such as the statistical distribution of the remaining bins permits an efficient entropic coding of their values.

2. THE MOJETTE TRANSFORM

2.1. Principles of the Mojette transform

The Mojette transform has been described and used for the last ten years [3][4]. It corresponds to a linear discrete exact Radon transform [5], i.e. a set of I discrete projections describing the discrete image f. Projection angles are chosen among discrete directions $\theta_i = atan(q_i/p_i)$ where $i \in I$ and such as p_i and q_i are integers prime together $(GCD(p_i, q_i) = 1)$. Thus, the equation defining the transform is given in 2D (see also fig. 4) by:

$$\mathcal{M}f(k,l) = \{M_i f(k,l) = proj(p_i, q_i, m); i \in I\}, \quad (1)$$

with

$$\mathcal{M}_i f(k,l) = \sum_k \sum_l f(k,l)\Delta(m - p_i.k + q_i.l), \quad (2)$$

where $\Delta(m)$ is the discrete Kronecker function. The transform is thus linear both in the number of projections I and in the number of pixels denoted by N. The number of bins onto a projection of direction (p_i, q_i) depends upon the shape of the support. A major difference with the Radon transform is that the bins spacing onto the projection depends on the projections directions, i.e. is different for each (p_i, q_i) pair value. The sampling step on the i-th projection is $h_i = \frac{1}{\sqrt{p_i^2+q_i^2}}$. The consequence of this specific sampling is that the number of bins onto the projection depends of the (p_i, q_i) values and on the shape of the region to be projected. In the case of a rectangular $P \times Q$ shape, the number of bins of the i-th projection is given by:

$$\#bins_i = (P-1).|q_i| + (Q-1).|p_i| + 1, \quad (3)$$

The total number of bins is then:

$$\#bins = \sum_{i=1}^{I} \#bins_i, \quad (4)$$

This linear transform generates a redundancy usually computed with the following index: $red = \frac{\#bins}{N} - 1$.

In the previous rectangular shape case, a condition for allowing the reconstruction of the image is the Katz criterion [6], for which the image can be reconstructed if:

$$\sum_{i=1}^{I} p_i \geq P \qquad \text{or} \qquad \sum_{i=1}^{I} q_i \geq Q, \quad (5)$$

2.2. Algorithms for direct and inverse Mojette transform

The direct and inverse transform algorithms are now presented. They both have a complexity order of $O(IN)$.
DIRECT ALGORITHM

Algorithme 1 Algorithm for direct Mojette transform

for $index\_projection\ [i] \leftarrow 1\ \ I$ **do**
  for $index\_pixel\ [l.P+k] \leftarrow 1\ \ N$ **do**
    $b \leftarrow p_i.l - q_i.k$
    $bin(p_i, q_i, b) \leftarrow +image(k, l)$ $(*)$
  end for
end for

INVERSE ALGORITHM

The inverse transformation can be implemented with an algorithm that has the same complexity than the direct transform, i.e. $O(IN)$ with only (integer or modulo, see 2.4) additions / subtractions operators instead of finding an inverse matrix formulation.

Algorithme 2 Algorithm for the inverse Mojette transform

Require: $image \leftarrow$ -1 //Reset of the image map
for $index\_pixel\ [l.P+k] \leftarrow 1\ \ N$ **do**
  $(n, b) \leftarrow$ Step 1
  $(k, l) \leftarrow$ Step 2 (n, b)
  $image(k, l) \leftarrow bin(p_i, q_i, b)$
  Step 3
end for

Step 1: List of one-to-one correspondences between bins and pixels. b and n represent respectively the bin index and the pixel index, and the (b, n) pair symbolises their correspondence in the list.
Step 2: Linking information between bins and pixels allows to easily retrieve pixel coordinates.
Step 3: Updating the bins values on projections (subtracting the pixel value from the matching bins on each projection).

Considering a N pixels shape and I projections, the order of complexity of the direct algorithm is quite obviously $O(IN)$ as shown above. Less obviously, the order of complexity of the inverse algorithm is also $O(IN)$. To reach this low complexity, three different kinds of information must be used : the set of projections containing the bins, the size of the image, an array managing a relationship between pixel coordinates (k, l) and bin indexes (b). This directly gives for a bin the position of the matching pixel in order to back project it onto the image.

The comparison between the Mojette algorithm and the FFT (same complexity order) shows that only additions and substractions are computed in the first case. Moreover, the reconstruction does not need the information to be sorted or ordered in the Mojette case; i.e. any one-to-one correspondence can be used at any time without modifying the complexity of the algorithm.

The inverse algorithm is iterative: at each loop, the image is progressively reconstructed by using an one-to-one bin (a bin in one-to-one correspondence with a pixel). If several one-to-one bins are candidates, a choice is randomly made. In this study we aim at updating the classical inverse Mojette transform for an efficient coding scheme, by prioritizing this one-to-one correspondence choice.

2.3. Mojette transform and correlation

Let $\mathcal{C}_f$ be the 2D autocorrelation of the image f, and $\mathcal{C}_{\mathcal{M}_i f}$ be the 1D autocorrelation of the Mojette projection $\mathcal{M}_i f$ (see eq.2). It can be shown that [7]: $\mathcal{C}_{\mathcal{M}_i f}(b) = \mathcal{M}_i \mathcal{C}_f(k, l)$.

This result highlights the fact that the correlation between image pixels remains between consecutive bins onto the projection. This result will be exploited for bins values coding purposes.

2.4. Class of Mojette transform

Mojette transform can be considered as a class of transform. Various versions of this transform can be stated depending on variations of the algorithm or on the models of pixels used (see fig. 5). For instance in the projection algorithm, classical additions (see algo. 1$(*)$) can be replaced by modulo additions so that bins values can still be coded with the same number of bits. If binary is considered (mapping of a binary stream onto an image), additions can be replaced by XOR operations between bit pixel values on the projection line.

3. SELECTION OF PROJECTIONS

The first step for the reduction of amount of data to transmit is the selection of a set of projections, so that the overall number of bins is minimal.

To minimize the number of bins we proceed as follows, q_i values are fixed to 1, so equations 3 and 4 become: If $\sum_{i=1}^{I} |p_i| = P$, the Katz criterion (see equation 5) is satisfied and the number of bins is limited. Finally, the p_i values are choosen such as $|p_i| = \lfloor \frac{P}{I} \rfloor + \epsilon$ where ϵ is an increment, $\epsilon = \{0, 1, 2, \cdots\}$. The set of projection directions will be the following:

$$S_I = \Big\{(1,0), (\lfloor \frac{P}{I} \rfloor + \epsilon, 1), (-\lfloor \frac{P}{I} \rfloor + \epsilon, 1) \quad /\epsilon = \{0, 1, \cdots, \lfloor \frac{I}{2} \rfloor - 1\}\Big\} \tag{6}$$

As a consequence of this specific choice of the directions (p_i, q_i), the projection angles are close to each others and the resulting projections are strongly inter-correlated.

4. SELECTION OF BINS

The Mojette transform introduces some inter-projections redundancy. The whole computed set of projections is necessary to reconstruct the original image, nevertheless after image reconstruction some non-used bins will remain. Our goal is to set some rules aiming at choosing precisely the useful bins for the reconstruction. Globally speaking, the purpose is not to suppress the inter-projections redundancy, but to be able to choose progressively the bins so that the projection encoding is minimal.

The bins selection is achieved by the encoder. Just after the projections computation, an upgraded inverse Mojette transform is used. A new step is added in order to select, at each loop of the pixels reconstruction process, the useful bin among the set of candidates bins.

At the end of the process we want to get a set of compact projections is obtained, *i.e.* for each projection, one or two useful bins-streams without holes and located at projection extremities. Such a compact projection is well-configured for the coding because consecutive bins are more correlated, and bins positions do not have to be supplied.

4.1. Metrics

4.1.1. Definitions and metrics

Some cost functions have to be settled. For a given iteration of the reconstruction algorithm, we call (see fig. 1):
Candidate Bin (CB): one-to-one bin that candidates for reconstruction;
Used Bin (UB): useful bin that has already been chosen for reconstruction;
Non Used Bin (NUB): unusable bin for reconstruction (all the pixel on its projection line have already been reconstructed);
Gap: blank space between 2 bins (UB and/or NUB), or between 1 bin (UB or NUB), and an extremity of the projection.

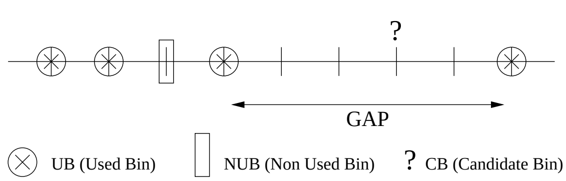

Figure 1: Bins definitions.

Let's also define the different metrics, we use:
Distance between two bins: $dist(Bin_1, Bin_2) =$ *number of possible bins between them* $+1$;
Distance between one bin and a projection extremity: $dist(Bin, EXTREM) =$ *number of possible bins between them* $+1$.

4.1.2. Score computation

We detail now the computation cost executed for each choice among the CB set.
Primary distance: if the CB is inside a gap, then we compute $dist(CB, B1)$, where $B1$ is the closest bin (UB or NUB) or the closest extremity of the projection. If there is no UB or NUB on the projection, we compute $dist(CB, EXTREM1)$ where $EXTREM1$ is the closest extremity on the projection. Indeed, choosing the CB minimizing the primary distance aims at filling a gap from its edges.
Gap score: we compute $(\alpha/dist(CB, B2))$ where B2 is the bin (UB or NUB) on the other edge of the gap or the other extremity of the projection (case with no gap) ; and where α is the number of bins onto the projection.
The second distance of the cost aims at filling in priority the smallest gap (because quicker to fill). The higher $dist(CB, B2)$, the bigger the gap.

Finally, the selected CB minimizes the global score:

$$score(CB) = primary\ distance(CB) - gap\ score(CB) \tag{7}$$

4.2. Refinements

In the previous section, rules used for selecting the best CB were explained. Still some refinements on these rules are necessary in order to improve the bins transmission and because some CB have identical scores.
Tolerance to NUB: Some NUB that split the bin-stream into two parts are kept in order to complete it and to avoid coding and sending start and stop positions for two bin-streams.
Priority to the largest projection: For higher p_i, the projection lines counts less pixels. As a consequence, the resulting projections offer more CB. Nonetheless, if many CB are selected onto this projection, more NUBs will appear on the other projections.
Pixels cluster: Clustering of image pixels during reconstruction is favored, so that the selected CB remain close to each other. This allows to promote larger bin-streams.

5. EXPERIMENTAL RESULTS

Let f be a 12×12 image. Such an example gives a typical and easily understanding representation of the process. The selected set of projections is $S = \{(1\ 0), (1\ 1), (-1\ 1),$

$(2\ 1)$, $(-2\ 1)$, $(3\ 1)$, $(-3\ 1)\}$, which gives $\#bins = 216$ and $red = 0.5$. After selection 144 bins have to be transmitted (smallest amount of bins possible). Figure 2 gives the layout of the projections. The coloured bins have to be transmitted. We observe that, except in the last projection of the set, bins are grouped into bins-streams. In the last projection $(-3, 1)$, the selected bins are spaced. This projection always contains gaps because candidate bins of the short projections are favored in case of *ex aequo* scores. In practice, this last projection will be transmitted within a single bin-stream.

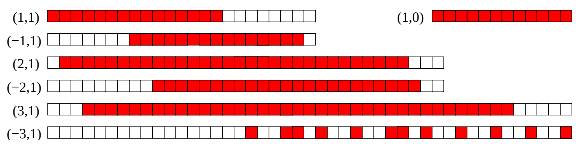

Figure 2: Projections and selected bins.

Figure 3 shows the image at different steps of the reconstruction process. We clearly see that from images corners, a pixels cluster is growing, up to the complete image reconstruction.

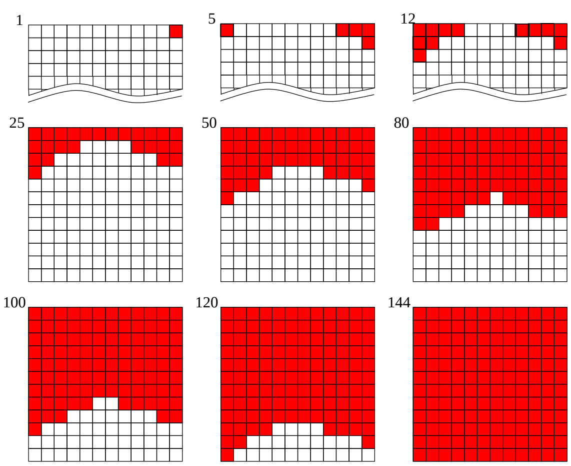

Figure 3: Different steps of the reconstruction algorithm.

6. CONCLUSION

In this paper, a specific implementation of the Mojette transform was presented for a coding scheme application. The method is based on an iterative process for the image reconstruction, associated with a bin selection according to coding criteria. A metric, with classification of bins (CB, UB, NUB), and some rules aiming at reducing gaps between selected bins, have been introduced. An example was given, representing the projections layout and the reconstruction of an image, in a typical case.

7. REFERENCES

[1] V. Goyal, J. Kovacevic, R. Arean, and M. Vetterli, "Multiple description transform coding of images.," in *ICIP 98*, October 1998, Proc.IEEE, pp. 674–678.

[2] B. Parrein, P. Verbert, N. Normand, and J. P. Guédon, "Scalable multiple descriptions on packets networks via the n-dimensional mojette transform.," in *ITCom 2001*, Denver, USA, August 2001, vol. 4524 of *Proc.SPIE*, pp. 243–252.

[3] N. Normand and J. P. Guédon, "The mojette transform : a redundant image representation.," *Comptes-rendus de l'Académie des Sciences de Paris, theorical computer science*, pp. 124–128, January 1998.

[4] J. P. Guédon and N. Normand, "The mojette transform : applications for image analysis and coding.," in *VCIP 97*, February 1997, vol. 3024 of *Proc.SPIE*, pp. 1220–1230.

[5] J. Radon, "Uber die bestimmung von functionen durch ihre integrawerte langs gewisser mannigfaltigkeiten.," *Berichte Sachiste Academie der Wissenschaften, Math-Phys*, vol. K1, 69, pp. 262–267, 1917.

[6] M. Katz, *Question of uniqueness and resolution in reconstruction from projections*, Springer Verlag, 1977.

[7] O. Philippé and J. P. Guédon, "Correlation properties of the mojette representation for non-exact image reconstruction.," in *PCS'97*, Berlin, September 1997, Proc., pp. 237–241.

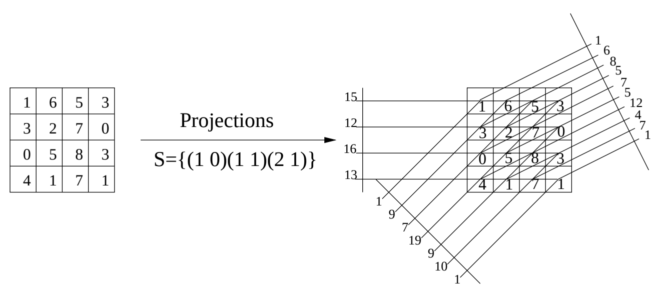

Figure 4: Example of projections set for a 4×4 image.

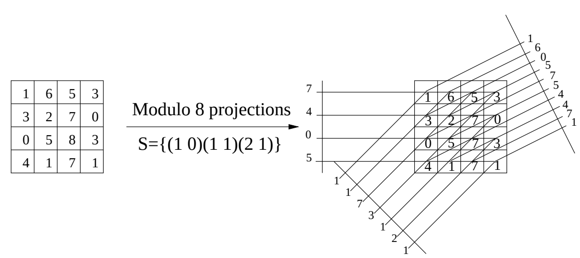

Figure 5: Exemple of modulo 8 projections.