14th European Signal Processing Conference (EUSIPCO 2006), Florence, Italy, September 4-8, 2006, copyright by EURASIP

# PDF SHARPENING FOR MULTICHANNEL PREDICTIVE CODERS

*Cihan Topal, Ömer Nezih Gerek*

Anadolu University, Department of Electrical and Electronics Engineering,
Eskişehir, 26470 Turkey
phone: + (90) 222 3213550 X 6450, fax: + (90) 222 3239501, email: ongerek@anadolu.edu.tr

## ABSTRACT

*Predictive coders that split the prediction decision into contexts depending on the local image behaviour have proved to be practically useful and successful in image coding applications. Such predictive coders can be named as multichannel. LOCO is a simple, yet successful example of such coders. Due to its success, a fair amount of attention has been paid for the improvement of multichannel predictive coders. The common task for these coders is to split the pixel layout around the pixel of interest into a list of contexts or prediction rules that specifically succeeds in predicting the value in a reasonable way. The improvement proposed in this work is due to the well known observation that the prediction error pdfs are not identically or evenly distributed for each channel output. Although several methods have been proposed for the compensation of this situation, they mostly perturb the low complexity behaviour. In this work, it is shown that a two-pass coder is a simple, yet efficient improvement that perfectly determines channel pdf bias amounts, and the adjustment produces up to 5% compression improvement over the test images.*

## 1. GENERAL INFORMATION

Classically, predictive image coders exploit the correlation between neighboring or physically close pixels inside and image to obtain a low entropy output. After the transform or prediction stages, the output signals mostly have a narrower pdf or histogram, indicating a lower energy and lower entropy. If the predictor is kept the same throughout the whole input data, the sharpness of pdf solely depends on the prediction power of the algorithm. In case of multiple predictions that are selected according to an initially set context rule, the predictions within each selected context may correspond to a sub-optimal decision. The algorithm of a 4-channel predictor corresponding to 4 hypothetical contexts can be listed as follows:

```
if      (condition_1 is true)
        target_pixel= pred_1;
else if (condition_2 is true)
        target_pixel= pred_2;
else if (condition_3 is true)
        target_pixel= pred_3;
else
        target_pixel= pred_4;
```

*Figure 1*: A 4-channel predictor algorithm.

In such algorithms, it is normal to encounter situations where pdfs of each channel output are distributed different from each other. Since each condition is rule based, the rule may inherently produce biased or skewed distributions. For example, the prediction error for condition_1 may have distribution mean $\mu_1$, and condition_2 may $\mu_2$. These different prediction channel means would be unnoticeable if the overall prediction mean is close to zero. Compensating for the bias of each channel before merging the outputs may sharpen the histogram of the "overall" output.

Although the differences in biases of channel output distributions are commonly encountered cases, the reasons of these are hardly predictable using simple image models within short templates. As an example, LOCO is a very popular multichannel predictor which uses a short 2x2 template around the pixel of interest. Although the original algorithm is very simple and fast, yet fairly efficient, the authors of LOCO already have recommendations for the compensation of bias in channel outputs utilizing complicated stochastic models, perturbing the simplicity and beauty of the algorithm. Furthermore, these methods do not totally compensate for the bias in practical images. In another work regarding LOCO, the bias is not determined. Instead, the entropy coder following the prediction stage was compensating for the possible bias differences in channels.

Exact modelling of the bias or skew in a channel would completely solve the problem of prediction error histogram sharpening. However, such a model construction requires a-priori information of the considered image which makes it practically impossible to use. More generalized models, on the other hand, are quite complicated together with high computational complexity figures. Instead of such complicated attempts, it is proposed here that simple *measurement* of each channel histogram bias and skew after one pass of the multichannel predictor would provide all the information necessary to compensate and sharpen the overall output distribution.

## 2. PREDICTION ERROR PDF ALIGNMENT

No matter how many channels a prediction algorithm may have, the entropy coding stage that follows the prediction step only depends on the overall prediction error. In that case, the sharper (with less entropy and/or less energy) the output histogram is, the better the overall compression becomes. The following illustration in Figure 2 depicts a typical situation where a wide-spread input histogram transforms into a histogram with lower entropy.

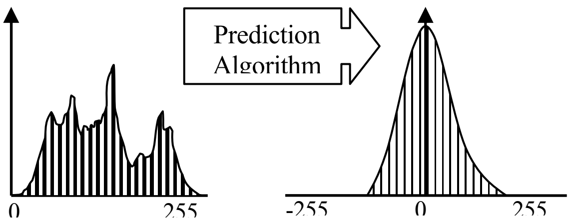


*Figure 2*: The effect of prediction on distribution

In this example, the output distribution seems to have a nice Gaussian distribution with a relatively low variance. However, since it corresponds to a straight collection of all the channel outputs, it does not carry any information about how each channel output histograms may look like. In this work, it has been observed that, even when the overall histogram exhibits nice properties, the individual histograms of separate channels may actually exhibit awkward behaviour including biased means or skews. It may, therefore, be possible that the above overall prediction error distribution may actually correspond to the system in Figure 1 with each channel distribution as illustrated in Figure 3.

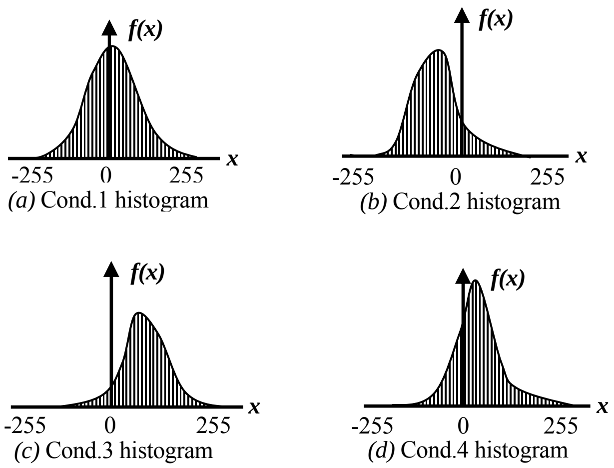


*Figure 3*: Possible error distributions of individual prediction channels

When the outputs of each channel with such distributions get collected, the overall output histogram becomes as in Figure 4. In the illustrated case, it is hard to claim that the overall histogram is optimum, although it may look so. Surprisingly, such behaviour is quite common in various standard predictive coders including CALIC or LOCO. It is, therefore, quite critical to sharpen the resulting histogram by manipulating the separate channel output histograms. As indicated in Section I, one has to also keep in mind that the manipulation should not perturb the simplicity of the original prediction algorithms by incorporating complicated stochastic models.

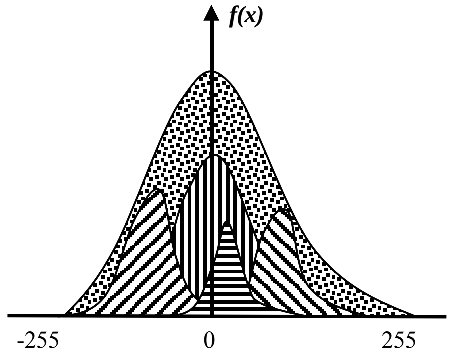


*Figure 4*: Overall histogram by combination of separate channel outputs.

### 2.1 The alignment process

The idea presented in this work is to compensate for the biases of the separate channel outputs. One actually neither wants nor needs to estimate for the bias. It has been observed that *measurement* of the bias is a simple and efficient approach. The measurement is obtained by running the algorithm once. Therefore, after the first pass, the bias and other statistical information is directly obtained. This two-pass approach technically does not increase the complexity order of the algorithm. Instead, the run-time is approximately multiplied by two. In the succeeding sections, it will be shown that the improvements obtained may be well worth the increase in the total execution time. Continuing with the above example, an improved (sharpened) histogram can be obtained as in Figure 5.

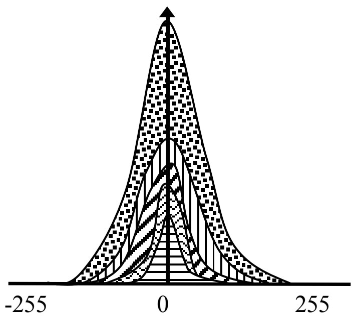


*Figure 5*: Cumulative histogram after aligning for the biases of separate channel histograms

## 3. DETERMINATION OF EFFECTIVE BIAS

Although the existence of bias is visible by observing the histograms of individual channel outputs, the effective amount of offset to be applied to each channel stays to be an issue. Depending on the minimization of the overall prediction error energy or entropy, the amount of shifts to be applied may vary.

Typically, there are three issues to be considered in the alignment process: the mean, the distribution peak, and the dynamic range. One has to be careful about the optimization goal. As an example, by simply aligning the means of each channel output distributions to zero, one may come up with a total distribution which has higher entropy.

It has been experimentally observed that the distribution peak (the mode of the sequence) gives the best entropy reduction for the LOCO multichannel predictor. The reason for this may be due to the skewness of the channel distributions. It has been observed that one of the three channels of LOCO gives a balanced distribution, whereas the other two have biases in opposite directions and skews in the negative direction of their biases. A skewed and biased distribution is illustrated in Figure 6.

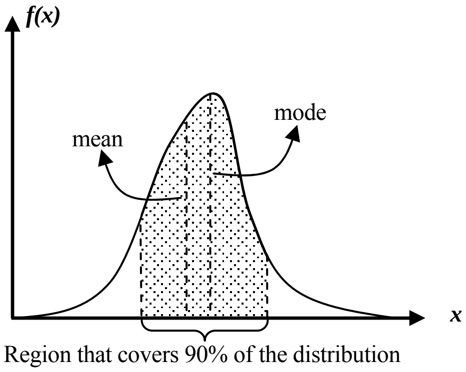


*Figure 6*: A biased and skewed distribution

In this figure, it can be seen that the majority of the data samples are not covered evenly around the mean. Instead, the mode is closer to the center of the region which covers 90% of the samples. These types of situations were commonly encountered in experiments that we have conducted using the LOCO prediction algorithm. Consequently, the alignment using the offset determined from the mode gave the best compression improvements.

At the end of such an alignment of individual channel distributions, the overall prediction error distribution was observed to be sharper with less entropy and energy. The overall scheme applied to the example in Figure 1 can be summarized as follows:

```
if      (condition_1 is true)
        target_pixel= pred_1 + offset1;

else if (condition_2 is true)
        target_pixel= pred_2 + offset2;

else if (condition_3 is true)
        target_pixel= pred_3 + offset3;

else
        target_pixel= pred_4 + offset4;
```

*Figure 7*: A 4-channel aligned predictor algorithm.

where offset1 to offset4 are first determined by running the code in Figure 1 once and looking at individual histograms.

## 4. EXPERIMENTAL RESULTS

The experiments are conducted over the original LOCO prediction algorithm that uses a simple three-channel context as follows;

Let the following 2x2 region be the template of interest that slides over the image in a raster-scan order:

| | |
|---|---|
| A | B |
| C | *X* |

*Figure 8*: A 2x2 LOCO template.

The LOCO prediction algorithm splits the prediction estimate into three channels according to the topological quantities of A, B, and C as:

```
if (max(A,B,C) = A)
      x̂ = min(B,C);
else if (min(A,B,C)=A)
      x̂ = max(B,C);
else
      x̂ = B + C – A;
```

Each channel produces its own output distribution, however, at the end, there is only one prediction for each pixel. Although the algorithm is fairly simple and intuitively justifiable, its compression results are very good, and it has even been adopted as the algorithm to be used in JPEG-LS. As a result, it is a good basis to test the proposed histogram sharpening method. Furthermore, it has been experimentally observed that LOCO, indeed, produces biased histograms in its sub-channels. The reason for this can be justified as follows. Consider the following image gradient cases where we apply the LOCO prediction as in Figures 9(a) and 9(b):


(a) $\hat{x}$ =min(B,C) (b) $\hat{x}$ =max(B,C)

*Figure 9: (a)* LOCO condition_1 *(b)* LOCO condition_2

These two cases are actually the situations that are governed by the first and second channel conditions of the LOCO algorithm; therefore they are frequently encountered in images. It is clear that, for the condition in Figure 9(a), the true value of pixel $X$ is actually less than $\hat{x}$ =min(B,C). Therefore, the difference $X - \hat{X}$ is negative. Since this situation is specific to this channel (channel 1), the channel output distribution is expected to have a negative bias. Similarly, channel 2 has a typical situation as depicted in Figure 9(b), and the prediction error $X - \hat{X}$ is, this time, mostly positive. Obviously, since the positive and negative bias situations are expected to be encountered at similar number of times through an arbitrary image, the overall prediction error exhibits no bias. However, this time, it is possible to improve the overall distribution sharpness by obtaining the biases in channel 1 and channel 2. The experiments are performed on an application with graphical user interface that shows the distributions of LOCO channels and lets the user to select the amount of offset to be applied to each channel. Since the amount of evaluated bias is put into the coded bitstream, lossless compression is guaranteed. Several energy and entropy reduction examples can be given. However, due to lack of space and to better improve flexibility, the experiment application and some test images are made available for download [6]. A screen output is illustrated in Figure 10:

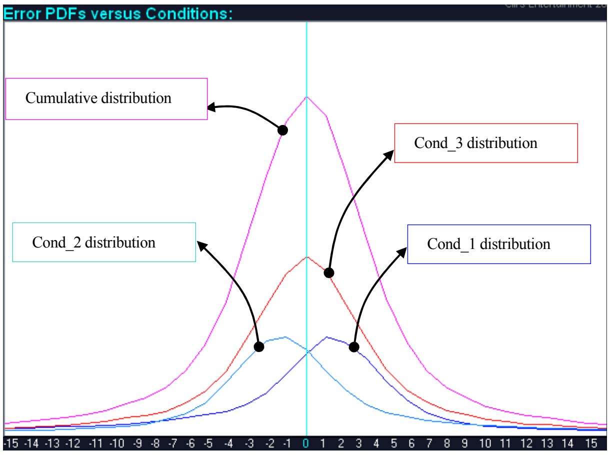


*Figure 10:* Error distributions for test image: boats.raw

Figure 10 clearly proposes that improvement is expected for this test image. Similar situations occur in all of the commonly used test images. The situation for the peppers test image is shown in Figure 11. Here, it can be noted that, after the alignment by an amount taken as the modes of distributions for condition 1 and condition 2, an improvement is visible.

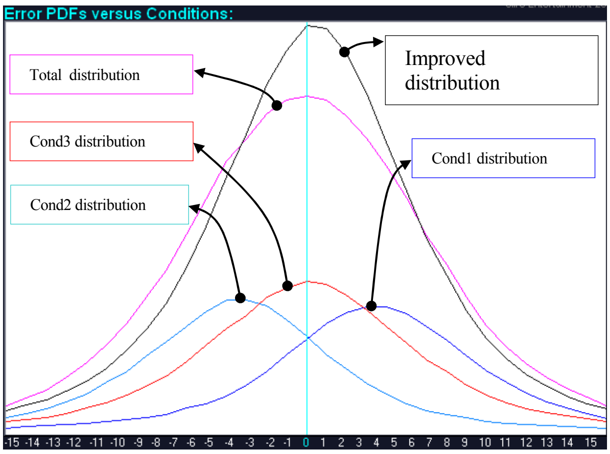


*Figure 11:* Corrected and original overall distributions for peppers test image.

The illustrated software also displays the entropies and energies of the original and improved distributions together with all the statistical properties of individual channel distributions such as mean, mode, skewness, kurtosis, etc. As a result, the improvements listed in Table I were obtained.

The improvements in the entropy directly apply to the practical coding results. To incorporate the bias amount of each channel into the coded bitstream, 12 bits are sufficient. This extra burden is very marginal, and it does not affect the overall compression ratio for the following test images with sizes 512x512. Therefore, the reduction in entropy is almost identical to the overall compression improvement.

Table I : Experimental Results.

| Image | Original LOCO Entropy | Improved Entropy | Improvement |
|---|---|---|---|
| Peppers | 4.9423 | 4.8082 | %2.71 |
| Boats | 4.4081 | 4.3821 | %0.59 |
| Lena | 4.5470 | 4.5331 | %0.30 |
| Mandrill | 6.2748 | 6.2715 | %0.21 |

The improvements are marginal for some test images, but for other images with smooth surfaces having gradients, the improvements become significant.

## 5. EXTENSIONS AND DISCUSSIONS

The histogram sharpening method described here also inspires the idea that, if one can obtain sharp (but probably biased) sub-distributions along several prediction channels, the channel distributions can always be aligned in the second pass and an overall sharp output can be obtained. In that case, smartness and efficiency of the unified or single prediction would be of little concern. Instead, the splitting of the prediction into several channels would be more important. This effect is being investigated as an extension by the authors.

In this work, however, the multichannel distribution alignment is investigated using a commonly used prediction algorithm: LOCO. It has been shown that biases of channel outputs can be easily obtained by running the algorithm once. After this determination, if these biases are added to respective channel outputs, the overall prediction error sequence has a sharper distribution with less entropy and energy. The proposed method is computationally simple, and it does not alter the complexity by incorporating complicated signal models. Furthermore, the improvement obtained this way is more than the reported improvements obtained by using more complicated bias cancellation methods in the literature.

## REFERENCES

[1] M.J. Weinberger,G. Seroussi, and G. Sapiro, " The LOCO- I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS," IEEE Trans. Image Proc., Vol.9, No.8, pp.1309-1324, Aug.2000.

[2] D. Guang, Y. Hua, L.W. Cahill, "Adaptive predictor combination and its applications in lossless image coding," International Conference on Advances in Medical Signal and Information Processing, 2000. pp.110-116, 4-6 Sept. 2000.

[3] ISO/IEC FCD 14495-2, "JPEG-LS Part 2," Apr. 2000.

[4] X. Wu and N. Memon, "Context-based, adaptive, lossless image coding," IEEE Trans. Commun., Vol. 45, No.4, pp.437-444, Apr. 1997.

[5] B. Carpentieri, M.J. Weinberger, G. Seroussi, "Lossless Compression of Continuous-Tone Images," HP Laboratories Report No: HPL-2000-163, 8 Dec. 2000.

[6] C. Topal, Ö.N. Gerek, "Modified LOCO analyzer++" online at: http://home.anadolu.edu.tr/~ongerek/mla