INTERNATIONAL COMPUTER SCIENCE INSTITUTE

1947 Center Street ● Suite 600 ● Berkeley, California 94704 ● 1-510-642-4274 ● FAX 1-510-643-7684

Distortion Accumulation in Image Transform Coding/Decoding Cascades

Michael Gilge †

TR-91-064

December 1991

Abstract

With an increasing number of applications that employ transform coding algorithms for data reduction, the effect of distortion accumulation caused by multiple coding needs to be investigated. Multiple coding occurs when more than one coding system is connected in a cascade. From the second stage on, the coding algorithm operates on data that has been previously coded/decoded. First a generic image communication system is modelled and situations that can lead to distortion accumulation are analyzed. These results show two main causes for distortion accumulation, which are then separately and jointly investigated using a JPEG-type compression algorithm. The first situation involves geometric operations between the decoding and next coding step. Measurements show, however, that the spatial manipulations are the main contributors to distortion accumulation. The second reason for distortion accumulation is a misalignment of the block segmentation reference point in subsequent transform operations. A block raster detection algorithm is derived that can find the position of the block raster, which was introduced in a previous coding step. If this information is used for the block segmentation of the following coding step, distortion accumulation can be avoided. Simulation results are given for an extended algorithm that registers regions of homogeneous block raster in images consisting of several subimages.

† Michael Gilge is currently with Philips Kommunikations Industrie AG, Public Communication Systems, Bldg. 98b Room 208b, Thurn-und-Taxis-Str. 10, D-8500 Nuernberg 10, Germany, Phone: 011 49 911 526 4164, Fax: 011 49 911 526 2278, email: ntae_mgi@pki-nbg.philips.de

I. Introduction

In the past, the aspect of distortion accumulation caused by image communication system cascades was not that important since digital image communication systems did not employ data reduction techniques. Digital images were either converted back to the analog domain for storage, e.g. VCR, or were not compressed, e.g. using pulse code modulation (PCM). Regarding recent efforts inside ISO and CCITT concerning a standardization of image data compression algorithms [1], this situation is definitely going to change. Employing data compression techniques will lead to many new applications and also a more widespread and networked usage of image communication systems. Given this situation, the effects of multiple coding will have to be considered. This may be surprising at first glance, since digital signals are generally considered immune from distortion accumulation by processing cascades. However, even visually lossless data reduction techniques alter the digital signal in a way that may lead to distortion accumulation if the same coding algorithm is applied a second time.

Recording studios have always been concerned with the notion of *generations*. TV-studios, for example, label each recording with the generation count, reflecting the number of copies that have been drawn from the original recording (generation zero). Editing, dubbing and copying account for increasing generation numbers. Only recordings up to a certain generation number retain broadcast quality. Considering the proposals for the new High Definition Television (HDTV) standard, which are all digital and based on data reduction techniques [2], the importance of investigations into generation loss of these signals under multiple coding becomes clear.

With the more widespread use of data compression algorithms, situations will arise in which digital images are fed through several coding systems. These coding systems may be located anywhere in the chain from image capture until the image reaches the final display or is permanently stored. In addition to data reduction algorithms, other types of operations are likely to occur interspersed between decoding/coding operations: a stored or transmitted image will be decoded and then filtered or edited, for example, before it is again coded for storage and/or transmission. Not only data reduction methods but any other image manipulation procedure may lead to a degradation of image quality. In particular, a cascade of multiple operations may cause severe distortion accumulation and thereby render the image useless for a specific application. In this paper, the causes for such distortion accumulation effects in image processing chains are investigated. This knowledge leads to countermeasures that can avoid image quality degradation.

Throughout this paper only the case of still images or intraframe coding for image sequences is considered. The scope is furthermore limited to transform coding algorithms using the discrete cosine transform (DCT). Incidentally, this approach is very popular as documented by the recommendation of the JPEG committee, which led to the ISO standard ISO/IEC CD 10918-1.

The related issue of *transcoding* has already been discussed in the literature, mainly in connection with speech processing: the number of different standards for speech signals necessitates transcoding between representations. Many transcoding schemes, however, are

known to cause distortion accumulation. For the case of transcoding between a vocoder and other audio signal representations, the effect of distortion accumulation has been investigated in [3], using subjective tests and objective measures. The papers [4] and [5] give solutions that avoid distortion accumulation for the case of $PCM \leftrightarrow ADPCM$ transcoding cascades of audio signals.

In order to estimate the total amount of distortion introduced by a signal processing system consisting of several components, the knowledge of the distortion characteristic of each single component is generally not sufficient. The single error contributions cannot be added because the errors of different components are correlated. For an efficient improvement of overall system performance, an identification of the components that contribute strongest to the total signal distortion is necessary. The work by [6] describes an alternative definition of an error measure, based on an estimation of the original signal after each stage, and has the property to yield the total error as the sum of the errors of each stage.

The paper is organized as follows: In section II a generic image communication system is modelled and situations leading to distortion accumulation are analyzed. To isolate the effect of geometric image manipulations, used in editing for example, distortion introduced by these operations is investigated in section III. Section IV explains situations that lead to distortion accumulation even though no interpolation of pixel amplitudes occurs between decoding and the following coding procedure. A solution to this problem is presented by the introduction of a block raster detection algorithm in section V.

II. Analysis of an Image Communication System

A generic *image communication system* consists of a picture source, e.g. camera, scanner, or VCR, a transmission channel, and an output device, often a monitor. For a digital transmission, the image has to be digitized at the source by an analog-to-digital converter (ADC). Similarly, the digital signal is converted back to analog (DAC) at the output. *Encoder* and *Decoder* exploit signal characteristics (*source coding*) and condition the signal with respect to the particular transmission channel (*channel coding*). In the following, the channel is assumed to be noiseless and band-limited, and therefore only source coding is considered.

A typical transmission system is depicted in **Fig. 1**. The output of an analog source (x, y) is digitized and fed into an image coder. For the type of image coders considered in this paper, this block would contain a transform unit (DCT) yielding the spectral coefficients, followed by a quantizer (Q). After inverse transform (IDCT) of the quantized spectral coefficients $s(u, v)$, a second quantizer constrains the pixel amplitudes to valid PCM values in the range [0..255]. These values $\hat{f}(x, y)$ approximate the original image values, with the closeness of the approximation determined by the quantization of the spectral coefficients.

The combination of coder and decoder, *Codec* for short, constitutes a single coding/decoding process. Multiple coding occurs if the output of a decoder is input into another coding system, e.g. by connecting two or more codecs in cascade. The reconstructed images after each stage are denoted by a generation superscript, $\hat{f}^1(m, n), \hat{f}^2(m, n), \hat{f}^3(m, n) \ldots$,

Fig. 2 : Explanation of the quantization process as a projection: all input signals x, which are represented by the same codeword i from the code space N, form an equivalence class $E_Q(i)$.

The principal operation of a quantizer is depicted in **Fig. 2**. The elements x belonging to the signal space M are projected into a code space N. All elements of a so called equivalence

Fig. 3 : Schematic explanation of the coding/decoding process: quantization in the spectral domain results in a representation of the coefficient u by u'. The subsequent inverse transformation may cause the reconstructed pixel amplitude x' to be in a different equivalence class $E_Q(n')$, thereby representing a different PCM luminance value.

This principal operation of a quantizer is now applied to the case of an image transform coding system, **Fig. 3**. Assuming a picture representation with 8 bit, the space of all possible luminance values consists of 256 elements. The transform (DCT) is defined over all real numbers, therefore the integers n are first converted to real-valued amplitudes x, which are transformed into real-valued coefficients u. Both real-valued number spaces are of equal dimensions, i.e. N picture elements are transformed into the same number of spectral coefficients. Note that in reality the transform does not operate on single values but rather on an ensemble of values, usually square image blocks of size 8×8. All pixel amplitudes inside the same ensemble (block) determine the values of the corresponding coefficients. However, only a single pixel x from the ensemble and a single coefficient u are depicted in Fig. 3 to facilitate the discussion.

The quantization of the spectral coefficients exactly parallels the previous discussion of the quantizer shown in Fig. 2: All real-valued coefficients inside the equivalence class $E_Q(k)$ are projected onto the code word k, which is recorded or transmitted, for example. The code word corresponds to a quantizer representation value u', which is converted back into the real-valued pixel amplitude x' by inverse transform. The equivalence classes of the reconstructed pixel amplitudes x' will differ partly from the equivalence classes of the original pixels x, caused by the quantization of the spectral coefficients. This is depicted in Fig. 3 by the second equivalence class $E_Q(n')$ which does not coincide with the original equivalence class $E_Q(n)$. A second quantization converts the real-valued amplitudes back to integer values in the range of [0..255]. In this case also a different reconstructed pixel value n' is assigned.

If the spectral coefficients are quantized with higher precision, i.e. a smaller quantizer stepsize, the size of the equivalences classes $E_Q(k)$ decreases. This results in an increased number of reconstructed pixel amplitudes which fall into the original equivalence classes – and therefore remain unchanged in the image – or an adjacent class. In the end, a lossless reconstruction is obtained if all amplitudes fall into their respective original equivalence classes.

The above discussion is easily extended to the case of multiple coding: Only if a second coding procedure of the reconstructed data leads to a representation by the same codewords k, no accumulation of distortion occurs. The reconstructed image quality is then solely determined by the first coding procedure. In other words, any changes to the spectral representation of the image will not change the reconstructed result as long as these changes keep all spectral coefficients inside their respective equivalence classes. If the reconstructed amplitudes n' are subjected to a second transform (which has to be based on the same subdivision of the image into ensembles (blocks)), the same spectral representation results, due to the unique mapping property of the used transform. If, on the contrary, a spectral coefficient is assigned to a different equivalence class as compared to the first transform, the reconstructed pixel amplitude is affected, resulting in an increased reconstruction error $\Rightarrow$ **distortion accumulation**. Two reasons can be isolated that would lead to a changed spectral representation:

1) **Changes to pixel amplitudes between coding steps**, i.e. after decoding and before new coding. This can be caused by affine operations in the spatial domain. Examples are filters or geometric manipulations.
2) **Changes in the subdivision into ensembles to be transformed**, i.e. the segmentation into independently transformed blocks is changed. This could be caused by changes in the image size, resulting in a loss of the reference point used in the previous subdivision.

Note that both reasons are likely to occur together. In order to evaluate their respective effects on the accumulation of distortion in cascaded coding systems, and due to the different causes of distortion accumulation, both cases are investigated separately. The first case is difficult to evaluate, because the spatial manipulations themselves will in general impair the image. To estimate the pure influence of spatial operations on the picture quality, these filters are investigated first without coding operations.

Fig. 4 : To calculate the influence of spatial operation on the picture quality, the forward and inverse operation has been applied in series, comparing the resulting picture to the original one.

Due to the spatially discrete nature of digital pictures, all geometric transforms necessitate a gray-level interpolation of pixel values. Pixel values are only defined at integer coordinates. The values for non-integer coordinates are determined by an interpolation of surrounding pixel amplitudes. Ideal interpolation calls for a lowpass filter with rectangular frequency response or, equivalently, a convolution with a *sinc*-function ($\frac{sinx}{x}$) in the spatial domain. A window function is commonly used to limit the infinite impulse response of the sinc-function. The window size determines the precision and the computational complexity of the filter. Simpler interpolation methods approximate the sinc-function using very narrow windows, e.g. bilinear interpolation and zero-order interpolation. The results for different interpolation schemes, as well as for different types of geometric manipulations, are depicted in the following figures.

Only horizontal translations with non-integer displacements, i.e. subpixel accuracy, necessitate gray-value interpolation. **Fig. 5** shows the results for horizontal displacements in the range 0.1 pixel to 0.5 pixel. Larger displacements are obtained in combination with integer displacements and therefore do not exhibit a different characteristic. The sinc-interpolation (length 10) gives the best result, compared to the simpler methods. The result for image zoom is shown in **Fig. 6**. While the sinc-interpolation gives good results that are almost independent from the zoom factor, the performance of simple methods varies greatly. Image rotation has been realized by an application of a skew transform along the image rows and

Fig. 5 : Quality impairment for subpixel image translation in horizontal direction and different interpolation methods; single affine operation.

Fig. 7 : Quality impairment for image rotation and different interpolation methods; single affine operation.

Fig. 8 : Quality impairment for multiple affine operations and different interpolation methods; each step consists of a forward/backward vertical shift by 0.1 pel.

In a second experiment, the same geometric operation was repeatedly executed. The result of up to ten affine chains, each of which was a vertical back and forth translation by 0.1 pixel, shows a continuous accumulation of distortion, **Fig. 8**. The different interpolation methods show the same behavior. After 5 chains, image quality already degrades by more than 10 dB. These results are also important for interframe coding of image sequences using motion compensation techniques: Motion information is derived from successive frames (motion estimation). According to the measured motion, parts of the previous image are shifted to form a good approximation of the current frame. Distortion accumulation from the shifting operation thereby limits the number of frames that can be described with sufficient quality alone by shifting image parts. Therefore, update information will become necessary for these image parts, even if the motion in the scene is perfectly in accordance with the measured motion vectors.

Further investigations of the original problem, a concatenation of transform coding and geometric manipulation systems, have shown that picture quality is mainly determined by the type of affine transformation and the choice of interpolation method. If transform coding systems are included in the above experiments, the absolute SNR values are lower. However, all curves show the same behavior as in the previous experiments without data compression. Therefore, the distortion accumulation behavior, depicted in Fig. 8, also holds for the case of interspersed codec stages. Transform coding may even slightly improve the objective signal to noise ratio at times: Due to the quantization of spectral coefficients, some corresponding frequencies are effectively filtered that might otherwise be responsible for artifacts caused by the geometric operations. This is especially true if the quantizer weighting scheme from the

Fig. 9 : After each transform coding/decoding operation, the reference point of the block raster is shifted by a one of the vectors (all vectors have integer pel accuracy), to arrive at the original position after 10 operations in order to facilitate comparison with the original image.

By investigating the consequences of a changed grouping of pixels into blocks with regard to distortion accumulation, a different reference point is selected for each coding/decoding

Fig. 10 : Quality impairment for multiple coding/decoding of the same image: Without geometric alterations of the image after decoding and before the subsequent next coding, picture quality remains stable, with interspersed geometric operations, picture quality continues to degrade.

The thus obtained image blocks were coded using the DCT in the context of the JPEG coding standard. The image processing cascade consisted of ten identical codecs, all using the same quantization parameters. The output of each codec was compared to the original image and the signal to noise ratio was computed. The results are shown in **Fig. 10**: no

accumulation of distortion occurs if the block raster is left intact after each codec, i.e. the image is not shifted and fed unaltered into the next codec. If the block raster, however, is changed by shifting the reference point in accordance to the vectors depicted in Fig. 9, the image quality is clearly degraded by each codec. A subjective impression of the respective picture quality as the result of the first and the last codec in the chain is given by **Fig. 11** and **Fig. 12**. Although the chosen compression ratio of 6 is rather moderate, the image quality after 2 coding/decoding procedures may be too poor for many applications. Fig. 12 shows the devastating influence of the changed blocksegmentation.

The reason for the accumulation of distortion is explained by the following example: **Fig. 13** show the spectral representation of an artificial image block. The pixel values correspond to a single DCT basis image – consequently the spectral domain after transforming shows only a single nonzero coefficient. Now the original image block, i.e. the DCT basis function, is shifted by one pixel distance to the right and down. In order to not introduce any edge artifacts, the block boundary pixel values are mirrored to the outside at all four sides. A second transform of the now shifted block yields the spectral representation plotted in **Fig. 14**. The pixel values do not coincide with a DCT basis function and the energy is spread over several coefficients.

Fig. 11 : Reconstructed image after a single coding/decoding operation.

Fig. 12 : Effect of multiple coding: reconstructed image after ten coding/decoding operations, using the same coding procedure as in Fig. 11. The block raster was shifted after each coding/decoding step using the vectors depicted in Fig. 9.

Employing the normalization followed by quantization in the spectral domain effectively cuts down on the number of significant coefficients. All coefficient amplitudes that are too small after normalization are quantized to zero. The quantized spectrum is shown in **Fig. 15**. Inverse transformation of the quantized coefficients yields an approximation of the original, but shifted, image block. If this approximation is now shifted back to the old position, the changed pixel amplitudes only resemble the original basis function. The difference to the block depicted in Fig. 13 can be seen after a second forward transform. The spectrum shown in **Fig. 16** shows many nonzero coefficients representing the perceived degradation in image content. Note also that the amplitude of the dominant coefficient has decreased in comparison to the original value.

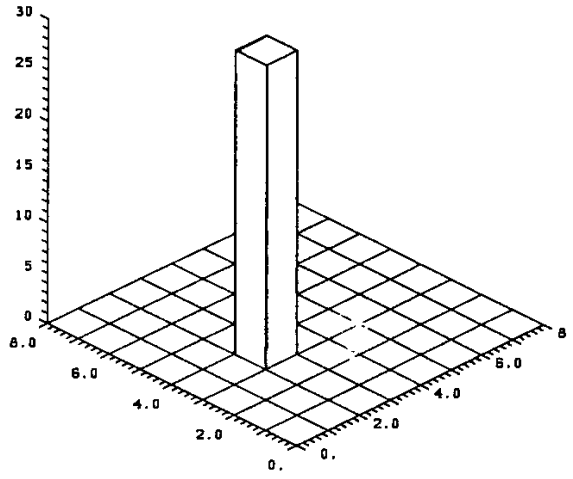

Fig. 13 : Spectral domain of an artificial image block: the grey levels represent a single DCT basis image, consequently the spectral domain after transforming the block shows only a single nonzero coefficient.

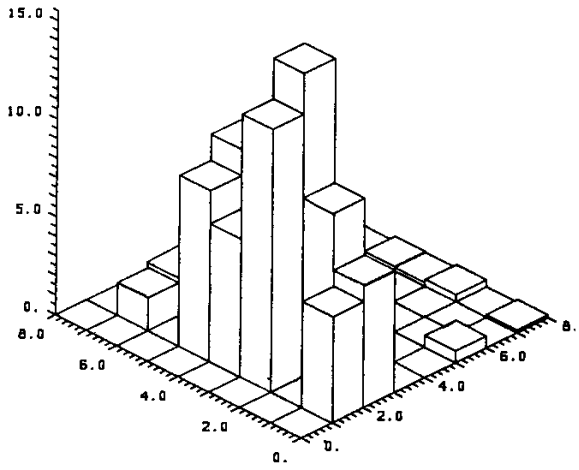

Fig. 14 : Spectral representation of the same basis image as in Fig. 13, but shifted in the original domain by one pixel to the left and one pixel position down (1,1).

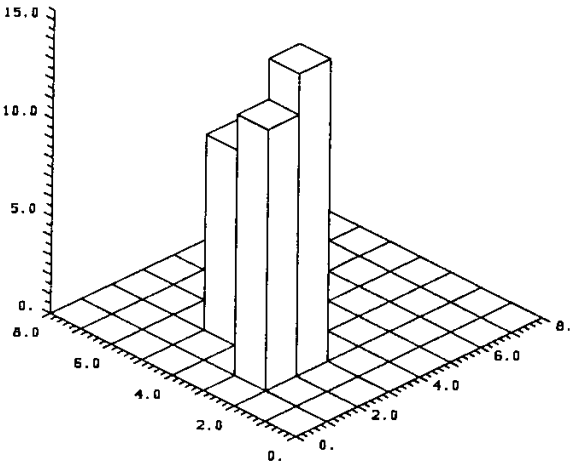

Fig. 15 : Spectral representation of the shifted basis image as in Fig. 14, but after normalization and quantization

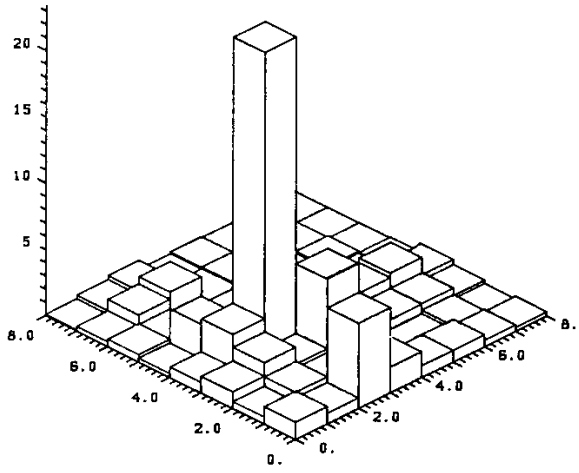

Fig. 16 : Spectral representation after the block has been shifted back to the original position; comparison with Fig. 13 shows nonzero coefficients which are responsible for the perceived image degradation.

Fig. 17 : Accumulation of error can be attenuated through reduced normalization array values (Q-factor).

Fig. 18 : Effect of the reduction in normalization values on the resulting compression ratio.

The achieved compression ratios are plotted in **Fig. 18** for both cases. Not only does image quality degrade rapidly under multiple coding. At the same time the obtained compression ratio decreases also, assuming unchanged coding parameters. Accumulating coding artifacts (see Fig. 12) are interpreted as image structures to be coded by the subsequent coding algorithms. To accommodate the error, an increased number of coefficients, and subsequently a higher data rate, is necessary. Reducing the compression ratio limits coding distortions, resulting in a flatter behavior of the compression factor.

To summarize, distortion accumulation can only be avoided if the block raster is taken into account in subsequent coding steps. Only if the same image subdivision is maintained in all coding steps of a data compression cascade, always the same spectral representation is reproduced. For visually lossless coding it may even be very difficult to correctly distinguish between a reconstructed image and an original that has never been coded before. In the following a block raster detection algorithm is introduced that determines whether an image has been previously coded. In that case, also the location of the reference point used for the block segmentation is found.

Fig. 19 : Definition of the block raster, constructed by segmentation into square, non-overlapping image blocks; all other blocks, not adhering tio the block raster are called intermediate blocks.

Before the algorithm is introduced some terms need to be defined. **Fig. 19** shows a schematic subsection of an image from the viewpoint of the coding algorithm. The segmentation of the image into nonoverlapping square blocks introduces the so-called *block raster*. In general, the block raster is not visible. Due to the independent transformation of the blocks, and especially in connection with high compression factors, the block raster may become visible in parts of the image, producing the infamous *blocking artifact*. Not knowing the position of the block raster, there are 64 different possibilities to group pixels into blocks, assuming a blocksize of 8×8 pixels. One of these positions coincides with the block raster and this block is called a raster block. The other 63 blocks are called intermediate blocks. It is usually sufficient for the detection of the block raster to find a single raster block. There is, however, the possibility that images are edited electronically and an image is combined out of several subimages. In these cases the different block rasters of the subimages will, in general, not be compatible.

Fig. 20 : Block diagram of the coding/decoding process and associated signals; please note the second quantization of pixel amplitudes after inverse transformation.

Considering the amplitude distribution of the spectral coefficients, it can be observed that the real-valued coefficients after the forward transform correspond to a continuous probability density function (PDF). A typical PDF of a coefficient (with the exception of the DC-coefficient) is plotted in **Fig. 21**. The quantization process maps all coefficient amplitudes to their respective representation values. This leads to a discrete probability density function, **Fig. 22**. The distance between the nonzero values of the PDF is determined by the quantizer stepsize. In the example, a stepsize of 30 was used.

Fig. 22 : Quantization causes the amplitude distribution of the coefficient to become discrete; quantizer step size assumed to be equal to 30.

The discrete nature of the PDF after quantization can be exploited for a detection algorithm. If the forward transform is applied on an image that has been coded before, the PDF of the coefficients is already discrete. For an arbitrarily selected image block the chosen block position must also coincides with the block raster in order to get a discrete PDF of the coefficients. On the contrary, if the PDF of the coefficients after the forward transform is continuous, the image either has never been coded before, or the chosen image block position does not agree with the block raster of earlier coding operations. All intermediate blocks give continuous coefficient PDFs after the forward transform. The algorithm would exhaustively try all 64 possible block positions and check for discrete coefficient PDFs.

Without the second quantization of the pixel amplitudes in the spatial domain after inverse transform, this detection would work perfectly. The rounding of pixel amplitudes after reconstruction causes the probability density function of the coefficients after the following forward transform to deviate from being strictly discrete. The rounding errors from the spatial domain propagate into the spectral domain. The errors of the pixels are assumed to be uniformly distributed and uncorrelated. This leads to a Gaussian distribution around the quantizer representation levels. Before the detection algorithm is presented, a slightly altered probability function is introduced which simplifies the understanding of the algorithm.

B. MODULUS-probability density function

To decide whether the probability density function is discrete or almost discrete, we would have to compare a given amplitude with the closest quantization level. If the probability density function is computed modulus the quantizer step size, all quantization levels are effectively mapped onto each other. This function is called *MODULUS-probability density function (MPDF)*. While the PDF gives the probability for the occurrence of a certain amplitude, the MPDF can be viewed as providing the probability of the deviation from the modulus.

Fig. 23 gives an example of 3 different MPDFs plotted in the same diagram for comparison: For a perfectly discrete PDF with all values coinciding with the quantization levels, the MPDF is a single impulse at the position zero and 100% amplitude. This curve is labelled *ideal* as it holds only without integer rounding of pixel amplitudes after the inverse transform. With the rounding operation the already mentioned Gaussian distribution around zero is obtained. About 10% of the coefficient amplitudes differ in amplitude from quantization levels. This curve corresponds to the case that the position of the transformed block agrees with the block raster used by any previous coding. If the image has never been coded before or the regarded block is an intermediate block, the MPDF is uniformly distributed as shown by the solid line. By thresholding the MPDF at the position zero the decision whether the block raster has been found can be made.

Fig. 23 : Introduction of the MODULUS probability density function: PDF is computed modulus quantizer stepsize, causing all amplitudes that coincide with quantizer representation levels to be mapped to zero.

C. Detection algorithm

The curves shown in Fig. 23 are based on all coefficients and the complete image. The result shows a significant difference of the MPDF amplitude at the position zero for the case of a coded raster block and the case of an uncoded or intermediate block. Taking all coefficients and the complete image into account is not only computationally expensive but is also not applicable to an image consisting out of several subimages. The detection therefore should be based on an image subregion or measurement window. However, in general it is not possible to identify any region size as sufficient for a secure detection of the block raster as this also depends on the image information of the region.

Using an adaptive algorithm the problem can be solved: starting with a small image region (seed) the measurement window is successively increased until sufficient evidence for the detection of the block raster is accumulated or the algorithm unsuccessfully terminates. Four neighboring blocks are used as the seed, **Fig. 24**. The evidence is computed from the MDPF as the confidence ratio between the number of coefficients that are observed to lie on quantization levels and the total number of coefficients used.

Fig. 24 : Four neighboring image blocks are clustered. The figure shows all possible 64 block raster reference positions, assuming a blocksize of 8×8.

Block Raster Detection Algorithm:

1) Computation of the confidence ratio based on four neighboring image blocks
2) Repeated computation of the confidence ratio at all 64 possible block raster positions (Fig. 24).
3) Best estimate for the position of the block raster corresponds to the highest confidence ratio.
4) If the confidence ratio exceeds a threshold, the position found in 3) determines the block raster.
5) If the confidence ratio is lower than the threshold, neighboring blocks are included in the search window and the confidence ratio is updated.
6) Step 5 continues until the confidence ratio either exceeds the threshold $\Rightarrow$ 4) or a predetermined maximum number of blocks is reached. Then the search is terminated and the picture is labelled **"not coded"**.

Investigations have shown that a search window of 24×24 pixels (4 blocks plus search area) is sufficient for a detection of the block raster.

D. Block raster detection for subimages

The introduced algorithm is now extended to images that contain subimages. **Fig. 25** shows an example of a combined image: Two subimages have been separately coded/decoded. A diamond-shaped part of subimage A has been inserted in the center. Although the block rasters were known, the images have been combined in a way that the block rasters of the

two images are not compatible. The algorithm should now not only find the positions of the respective block rasters, but also determine the shape of the regions that belong to either raster position. Note that no spatial image information, e.g. edge detection, is to be used. A certain intermediate area between the two subimages belongs to neither raster. The size and shape of this area necessitates special treatment, for example using region-oriented approaches [9] [10] or pixel-oriented methods like DPCM.

Fig. 25 : Example of an image being composed out of two different images. The editing process disregarded any respective block rasters used in a previous coding operation of either of the two images.

The algorithm starts at an arbitrary position in the image and begins searching for the block raster. Once the raster has been found, the algorithm checks the surrounding blocks for consistency with the raster position in a spiralling fashion around the seed, **Fig. 26**. If an inconsistent block is encountered, this block is marked and the consistency tests proceeds at a different position adjacent to the already clustered area. The search continues until all blocks surrounding the area of consistent block raster have been labelled inconsistent. This borderline of inconsistent blocks is followed until a closed contour or the image border is reached, **Fig. 27**.

Fig. 27 : Outline of the image region showing consistent block raster through following the borderline of inconsistent image blocks.

Fig. 28 : Continuation of the block raster registration process in an uncharted image region.

Fig. 29 : Simulation result using the image depicted in Fig. 25: All blocks with inconsistent block raster have been highlighted in white.

VI. Conclusion

If transform image coding systems are connected to cascades such that the output of one decoder is fed into the following encoder, the problem of distortion accumulation has to be considered. For the case of transform image coding investigated in this paper, distortion accumulation can only be avoided if the image remains unchanged between the decoder and the following encoder. In particular, any change in the grouping of pixels into blocks, e.g. caused by image cropping and the loss of the block raster reference point, leads to significant accumulation of distortion under multiple coding.

For practical purposes, the images are likely to be filtered or edited. The distortion accumulation of multiple coding in combination with affine transformations is mostly attributable to the spatial interpolation of pixel amplitudes. The introduced block raster detection algorithm enables the avoidance of distortion accumulation in cases where the reference point has been lost. Even for images that consist of several subimages, the respective block rasters are found and the shape of the regions is traced.

Recently region-oriented transform coding has been developed [9][10]. The subdivision of the image into regions is the result of a segmentation algorithm based on texture or

homogeneity. In this case the used grouping of pixels into ensembles that are separately transform-coded can always be recovered.

References

[1] Gregory K. Wallace: "The JPEG Still Picture Compression Standard", Communications of the ACM, Vol. 34, No. 4, April 1991, pp. 30-44

[2] Advanced Television Research Consortium: "Advanced Digital Television System, Technical Description", All-Digital HDTV Seminar, Columbia University, April 5, 1991

[3] D.J. Goodman, C. Scagliola, R.E. Crochiere, L.R. Rabiner, J. Goodman: "Objective and Subjective Performance of Tandem Connections of Waveform Coders With an LPC Vocoder", The Bell System Technical Journal, Vol. 58, No. 3, March 1979, pp. 601-629

[4] T. Nishitani: "Tandem Transcoding Without Distortion Accumulation", IEEE Trans. COM, Vol. 34, No. 3, March 1986, pp. 278-284

[5] M. Bonnet, O. Macchi: "A Robust Adaptive Quantizer With a Smoothed Speed Control Parameter", European Signal Processing Conference EUSIPCO-88, Grenoble, September 1988, pp. 1529-1532

[6] D.G. Messerschmitt: "Accumulation of Distortion in Tandem Communication Links", IEEE Trans. IT, Vol. 25, No. 6, November 1979, pp. 692-698

[7] J.E. Maisel, R.E. Morden: "Rotation of a Two-Dimensional Sampling Set Using One-Dimensional Resampling", IEEE Trans. ASSP, Vol. 29, No. 6, December 1981, pp. 1218-1222

[8] A. Tanaka, M. Kameyama, S. Kazama, O. Watanabe: "A Rotation Method for Raster Image Using Skew Transformation", Proceedings of the IEEE, 1986, pp. 272-277

[9] M. Gilge, T. Engelhardt, R. Mehlan: "Coding of Arbitrarily Shaped Image Segments Based on a Generalized Orthogonal Transform", Visual Communication, Special Issue of Signal Processing on 64 kbit/s Coding of Moving Video, Vol. 1, No. 2, October 1989, pp. 153-180

[10] M. Gilge: "Region Oriented Transform Coding (ROTC) of Images", International Conference on Acoustics, Speech, and Signal Processing ICASSP-90, Albuquerque, USA, April 1990