IX. Reducing the Block-Effect

The block-effect owes its existence to the independent processing of the sub-images, of a segmented image. In the segmentation process the statistical dependencies beyond the border of a block is not taken into account in DCT transform coding. Quantisation of the blocks introduces independent errors among the blocks that causes the discontinuities at the borders between blocks which is observed as the block effect. The block-effect origin is demonstrated graphically in figure 28. The discontinuities are inherent to the basis functions used in the DCT. A graph of the normalised discontinuity at the edges of the DCT is shown in figure 29. It can be seen that the lower "frequency" functions contain large discontinuities and would thus contribute more towards the block-effect.

The energy compactation should be viewed in conjunction with the "discontinuity size" to predict the influence of the quantisation of different coefficients on the block-effect. An experiment was conducted in which the low "frequency" components were first coarsely quantised and then the higher "frequency" components. The results together with the quantisation matrixes are shown in figures 30 to 33. From these results it is clear that quantisation of different coefficients will have different kinds of visual distortion. Low frequency quantisation led to a more pronounced block-effect, while high frequency quantisation led to a more grainy type of distortion.

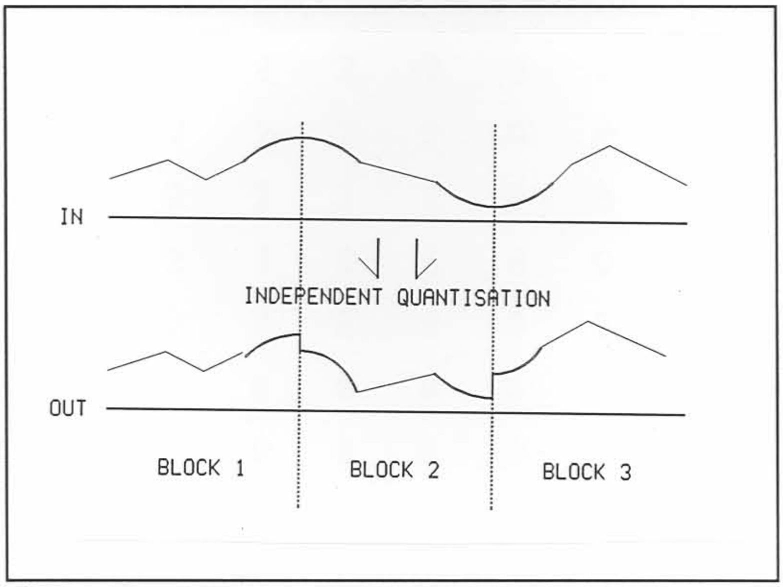

Figure 28 Block-effect: Result of independent processing and quantisation of blocks.

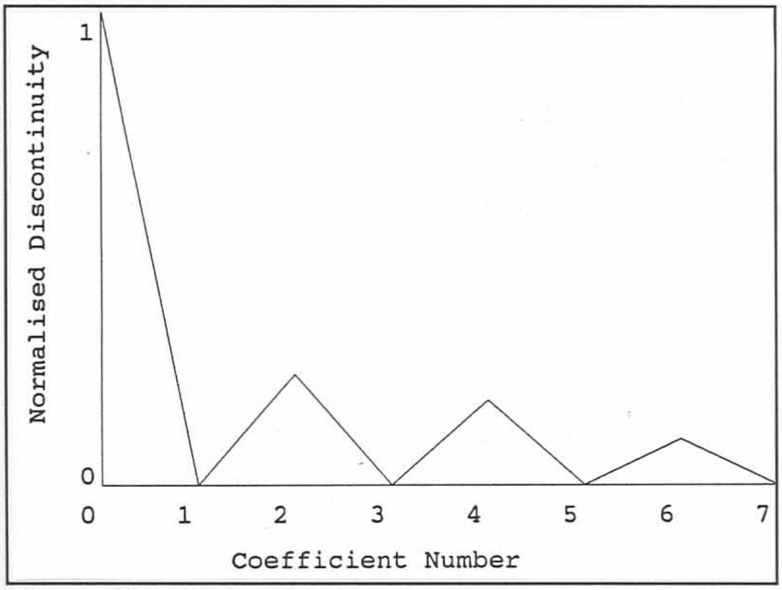

Figure 29 Normalised Discontinuities associated with the DCT Basis Functions.

9	2	2	2	2	8	8	8
2	2	2	2	2	8	8	8
2	2	2	2	8	8	8	8
2	2	2	8	8	8	8	8
2	2	8	8	8	8	8	8
8	8	8	8	8	8	8	8
8	8	8	8	8	8	8	8
8	8	8	8	8	8	8	8

Figure 30 Low frequency bitmap matrix for quantisation experiment.

Figure 31 Low Frequency Quantisation Experiment: GIRL

9	8	8	8	2	2	2	2
8	8	8	8	2	2	2	2
8	8	8	2	2	2	2	2
8	8	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2

Figure 32 High frequency quantisation experiment- bitmap matrix.

Figure 33 High Frequency DCT Quantisation Experiment: GIRL

The block-effect is a highly visible type of distortion, to a human observer, as a result of the periodic nature thereof and the effect it has on features crossing block-boundaries. At low bit rates the effect becomes very pronounced and may render the transform coder useless.

In this section four techniques will be discussed that attempt to reduce the block-effect. The techniques are filtering, overlapping, using a visual error criterion, and lapped transforms. The first two techniques will only be discussed briefly since they contain inefficiencies in solving the problem. The last two techniques, using the HVS and lapped transforms, each lead to usable improvements in the quality of the reconstructed image. The lapped transforms have been found to decrease the blocking effect considerably as well as resulting in higher coding gains than the corresponding DCT.

A. Filtering

The first and most obvious solution to the problem, is the use of a space variant filter along the boundaries of the blocks, called a post-filter. The post filter performs a smoothing action from one block to the next, with the result that features crossing boundaries tend to be blurred. This blurring can be reduced by using a pre-filter that enhances high frequency detail along the borders of the blocks to offset the low-pass filtering effect of the smoothing filter. Since high frequency detail is enhanced by the pre-filter, the image will require more bits to quantise to achieve equal fidelity. A nonlinear space-variant postprocessing technique which smooths jagged edges without blurring them, and also smooths out abrupt intensity changes in monotone areas has been proposed by Arch [66]. The disadvantage of filtering methods is that they invariably lead to a reduction in the coding gain. Results from using the spatially variant filter are given in figure 34 and 35. Figure 34 contains the original coded image and figure 35 contains the filtered image. The block-effect seems less but blurring has occurred along the edges of the blocks. The filter that was used simply replaced the border pels with their average value.

Figure 34 DCT coded image: GIRL - block size = 8x8, bit rate = 0.5 bits/pel.

Figure 35 Filtered Image: GIRL

B. Overlapping

A second method is that of overlapping of adjacent blocks, similar to the overlap and add methods used in speech processing. The overlapping of blocks is illustrated in figure 36. This method works well, since it takes the statistical dependencies into account. In the overlapping method, the blocks overlap slightly, so that redundant information is transmitted for samples at the block boundaries. The receiver averages the reconstructed samples from the neighbouring blocks, in the overlapping areas. The disadvantage of this approach is the added redundancy that causes an increase in the bit rate. No results are given in this section since it is similar to that of the LOT, that will be discussed in detail in a following section.

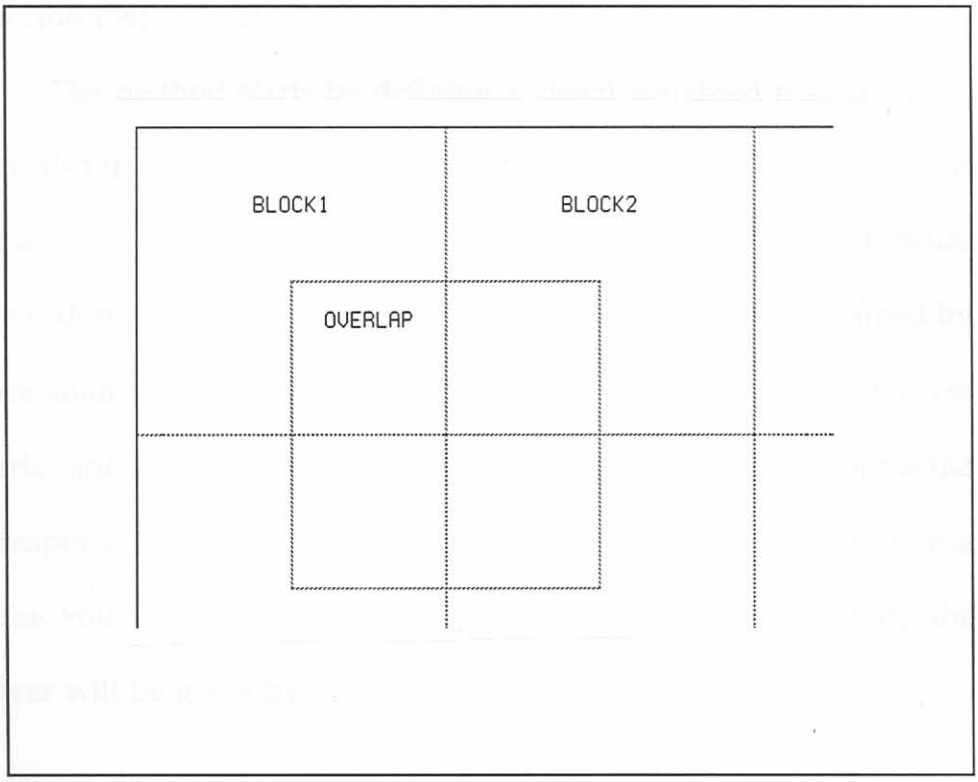

Figure 36 Overlapping of blocks in the transform domain.

C. Visual Error Criterion

A third method for reducing the block-effect is to optimise the bit allocation for quantisation. For example, the use of the human visual system (HVS) in the bit allocation algorithm will result in a more optimal allocation, that takes the user of the information into account. Using the HVS allows one to minimise the visual error, rather than the mean square error (mse) criterium normally used. This means that the visibility of the errors, such as the block-effect, will be taken into account when the coefficients of the transform are quantised. What needs to be done to implement this technique is to determine a HVS weighing matrix that determines the relative importance of the different coefficients. A method that achieves this has been introduced by Eggerton [56].

The method starts by defining a visual weighted mse criterion. They also use the HVS model by Mannos [16], but since that model is in the frequency domain it has to be transformed to the DCT domain. This is done as follows: Let x be a one dimensional vector obtained by concatenating the image rows. Let F and C represent the discrete Fourier and the discrete Cosine transforms and let X_F and X_C denote the corresponding vector of transforms. If H represents the HVS response in the Fourier domain, the DFT of the image as perceived by the viewer will be given by

$$Y_F = HX_F = HFx \tag{55}$$

The corresponding spatial function is

$$y_F = F^{-1}HFx \tag{56}$$

so that the DCT equivalent of the response Y_F will be

$$\begin{aligned} Y_C &= Cy_F = CF^{-1}HFx \\ &= CF^{-1}H(CF^{-1})^{-1}Cx \\ &= AHA^{-1}X_C \\ \textit{where} \quad A &= CF^{-1} \end{aligned} \tag{57}$$

If X_C is set to a constant value, the elements of Y_C provide the relative importance of the DCT coefficients to the human observer.

The first application of this result, using a 16x16 DCT and 16x16 DFT, gave a weighing matrix that actually worsened the block-effect. This was because the discontinuities at the block edges were not taken into account. The contrast inside a block appeared to be sharper than that of the normal mse codec, however the block effect made the technique useless. A method that took the block effect into account was given by Eggerton [56], in which the DCT is done on a block, say 16x16, and the DFT on the whole image. A flow diagram of this method is given in figure 37. The weighing matrix obtained for a block size of 8x8 is shown in figure 38, and in figure 39 the image Girl coded to 1.0 bits. It was found that the weighting matrix has little effect on image quality for rates above one bit per pel.

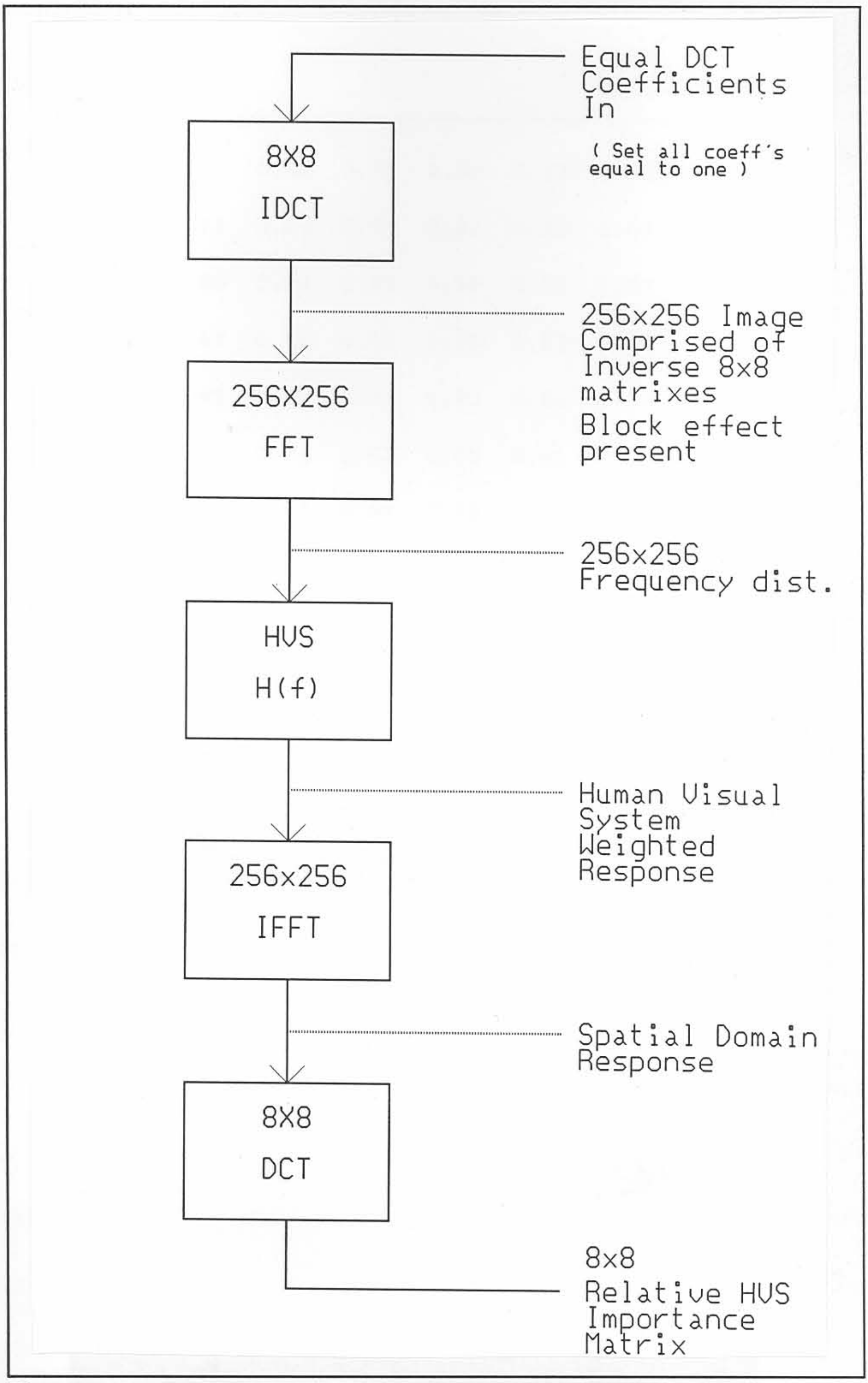

Figure 37 Processing steps to obtain the DCT domain weighting matrix from the Fourier domain function, while keeping the block boundary changes in mind.

0.05	0.85	0.82	0.89	1.00	0.77	0.90	0.61
0.84	0.71	0.86	0.67	0.81	0.58	0.69	0.47
0.82	0.86	0.94	0.85	0.99	0.73	0.87	0.59
0.89	0.67	0.85	0.62	0.76	0.53	0.64	0.43
1.00	0.82	1.00	0.77	0.92	0.65	0.79	0.52
0.78	0.58	0.74	0.53	0.65	0.45	0.54	0.37
0.90	0.69	0.87	0.64	0.79	0.54	0.66	0.44
0.61	0.47	0.59	0.43	0.53	0.37	0.44	0.30

Figure 38 Weighing matrix for 8x8 DCT with maximum display frequency of 6 cycles/degree.

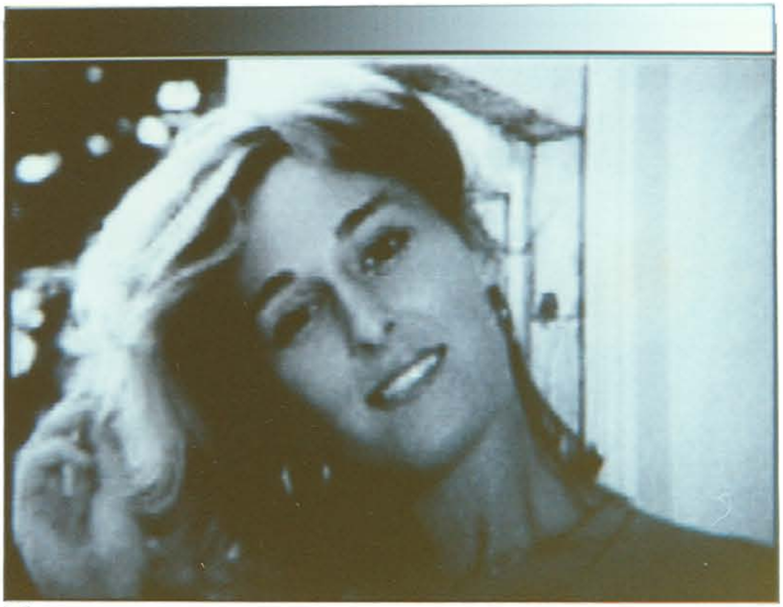

Figure 39 Human Visual System weighed DCT coded image GIRL, 1.0 bits/pel.

D. Lapped Transforms

The first of these transforms called the lapped orthogonal transform (LOT), was introduced by Cassereau [61]. Subsequently, Malvar and Staelin [62] introduced a new algorithm for generating LOT, that used the DCT as basis. By using the DCT as basis for the LOT, this algorithm is very attractive for practical implementation since it can use existing DCT algorithms and chips. Cassereau in conjunction with Staelin and de Jager recently published another paper [61], wherein an augmented Lagrangian method has been used to derive the LOT basis functions. Recognising the LOT as a special case of perfect reconstruction filter banks, Malvar [63] gave a new version of the lapped orthogonal transform, which can be efficiently computed for any transform length. He also introduced the modulated lapped transform (MLT), which is based on a modulated quadrature mirror filter (QMF). He found that the MLT can be efficiently computed by means of a type-IV discrete sine transform.

Malvar [63] found that the LOT and MLT are both asymptotically optimal lapped transforms for coding an AR(1) process with high inter-sample correlation. The coding gains of the LOT and the MLT are higher than that of the DCT for similar size, and were found to be close to that of a DCT of twice the size.

The DCT based LOT has been chosen for the simulations, based on the advantages of using the DCT as is, as well as the numerical stability of the LOT basis function when derived from the DCT. The derivation for the optimal LOT is given in Appendix A. The final form of the optimal LOT was found to be

$$P_o = \frac{1}{2}\begin{bmatrix} D_e - D_o & D_e - D_o \\ J(D_e - D_o) & -J(D_e - D_o) \end{bmatrix}\begin{bmatrix} I & 0 \\ 0 & Z \end{bmatrix} \tag{58}$$

where D_e is the even cosine basis functions, D_o is the odd cosine basis functions, J is the counter identity matrix, and Z is the optimising matrix. The fast implementation of the LOT is also given in Appendix A. A brief summary of the characteristics of the LOT is given:

1. The LOT requires approximately 30%-100% more computations than DCT, if the DCT is integrated into the algorithm,
2. The LOT is asymptotically optimal for first order Markov processes as the correlation coefficient approaches unity,
3. The LOT can be optimised to the statistics of the image by choosing the correlation coefficient and using it in the computation of the optimisation matrix,

4. The LOT decreases the block effect considerably as a result of the overlapping nature of its basis functions, and that they decay smoothly towards zero,
5. The LOT has a higher coding gain (0.3dB) than a DCT of the same size.

To compare the results achieved with the LOT to those of the DCT the test images were coded using the simple non-adaptive algorithm with Lloyd-Max quantisation for bit rates 2.0, 1.5, 1.0, and 0.5 bits/pel. The LOT fared better in most cases with a significant reduction in the block-effect. The block-effect was still visible at the lower bit rates but was less than that of the DCT. Graphical results are plotted in figure 40 and pictures for 1.0 and 0.5 bits/pel are shown for comparison purposes in figures 41-44. From figure 40 it can be seen that the LOT achieved a better signal to noise ratio than the DCT. This improvement was in accordance with what was expected.

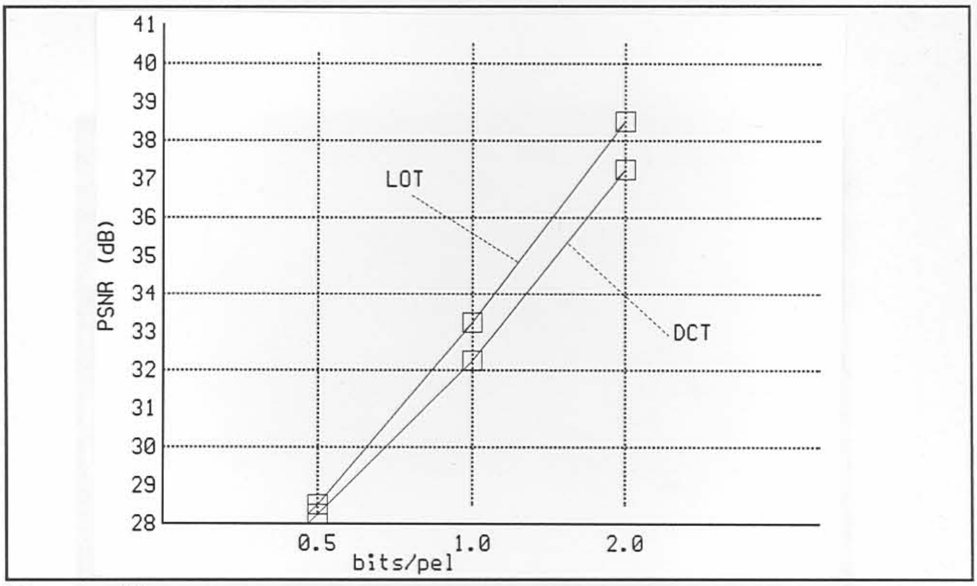

Figure 40 Comparison of LOT and DCT transform coding for images GIRL and ROAD

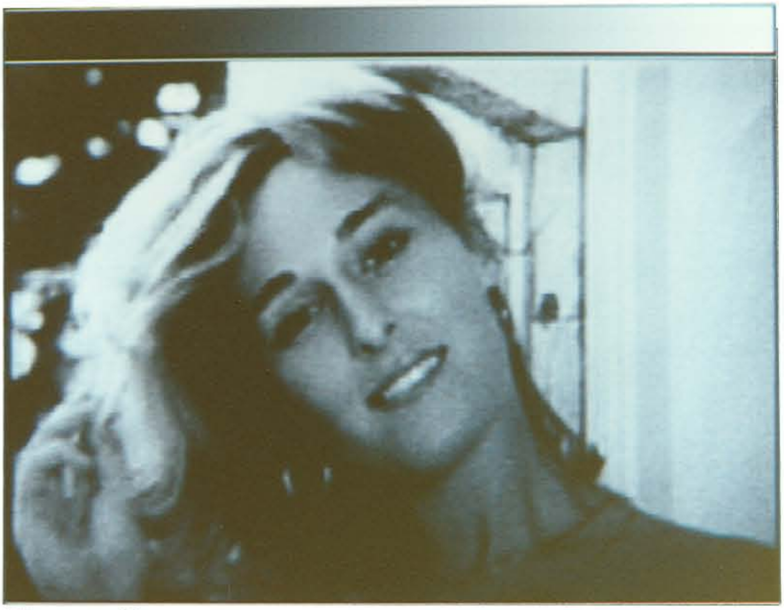

Figure 41 DCT Coded Image: GIRL - 1.0 bit/pel, 8x8 block size.

Figure 42 LOT coded Image: GIRL - 1.0 bit/pel, 8x8 block size.

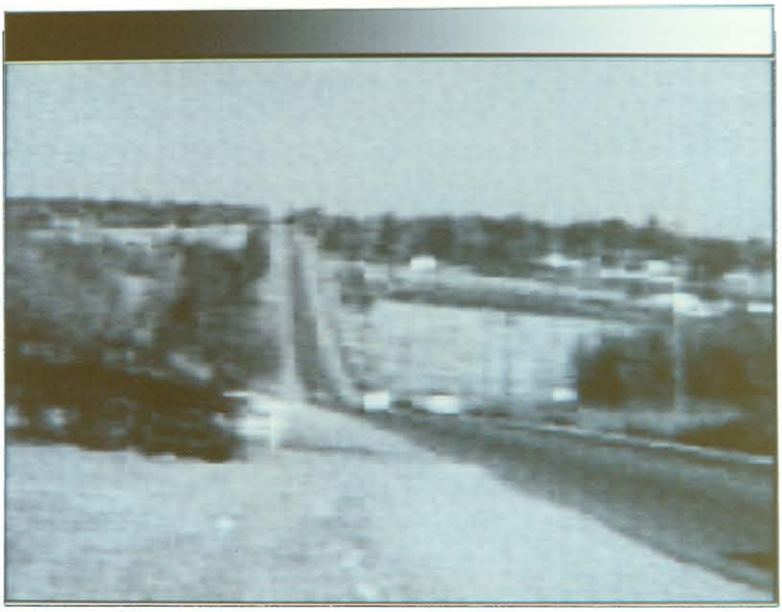

Figure 43 DCT Coded Image: ROAD 0.5 bit/pixel, 8x8 block size.

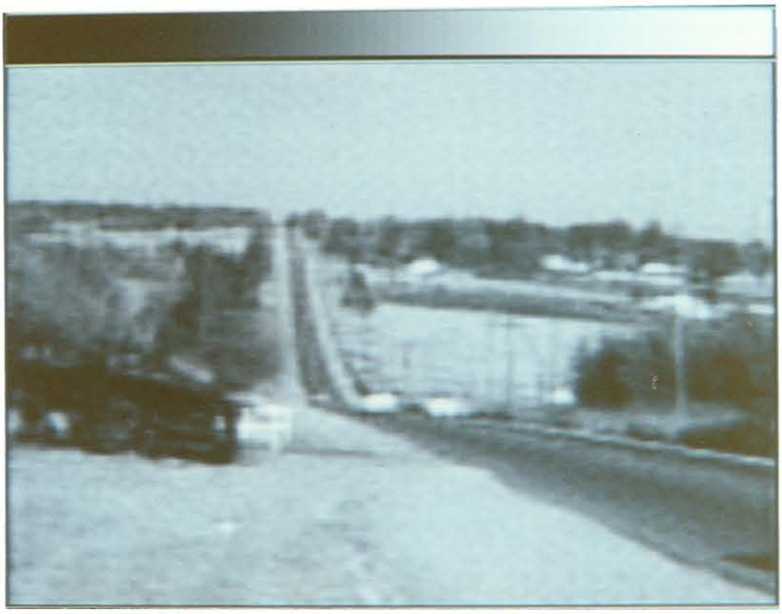

Figure 44 LOT Coded Image: ROAD 0.5 bits/pel, 8x8 Block size.