Reversible Interframe Compression of Medical Images: A Comparison of Decorrelation Methods

P. Roos and M. A. Viergever

Abstract—**The present paper investigates whether data representing medical image sequences can be compressed more efficiently by taking into account the temporal correlation between the sequence frames. The standard of comparison is intraframe HINT [1], which is the best known reversible decorrelation method for 2-D images. In interframe decorrelation, distinction is made on the one hand between extrapolation- and interpolation-based methods, and on the other hand between methods based on local motion estimation, block motion estimation, and unregistered decorrelation (i.e., without motion compensation). These distinctions give six classes of interframe decorrelation methods, all of which are described. The methods are evaluated by applying them to sequences of coronary X-ray angiograms, ventricle angiograms, and liver scintigrams, as well as to a (nonmedical) videoconferencing image sequence.**

The conclusions of the study are, for the medical image sequences: 1) interpolation-based methods are superior to extrapolation-based methods, 2) estimation of interframe motion is not advantageous for image compression, 3) interframe compression yields entropies comparable to intraframe HINT at higher computational costs, and 4) two methods, unregistered extrapolation and interpolation, are nonetheless possibly interesting alternatives to intraframe HINT. Unregistered interpolation gives slightly worse results than 2-D HINT at reduced (approximately 2/3) computational cost, whereas unregistered extrapolation gives significantly worse—but for many purposes still acceptable—results than 2-D HINT at significantly lower (approximately 1/5) computational cost.

I. Introduction

IMAGE compression is becoming increasingly important in efficient archiving and transmission of images. Reversibility of compression methods may seem a questionable option, since, on the one hand, many choices and compromises outweighing slight compression losses have already been made in the image acquisition, and on the other hand, the requirement that no information be lost severely limits the amount of compression. Yet in the design of picture archiving and communication systems, the compression step is often prescribed to be reversible [2], either for legal reasons or because postprocessing of the images would augment the losses introduced by a nonreversible method to an unacceptable level.

In reversible image compression distinction is made between two consecutive steps, decorrelation and coding. The image decorrelation aims to reduce the redundancy within the images. The thus obtained energy compaction is exploited by a variable length coder, e.g., arithmetic coding [3], Huffman [4], or Lempel–Ziv [5]. The coding step is not considered in this paper.

In a previous study on reversible intraframe image compression, HINT (Hierarchical INTerpolation) stood out as the most efficient decorrelation method [1]. For time series of 2-D images it is likely that a more efficient decorrelation scheme can be obtained by utilizing the temporal correlation in addition to the spatial correlation. The purpose of this paper is to investigate whether the temporal correlation can be exploited to increase the efficiency of image compression of a sequence with respect to 2-D intraframe decorrelation. Since an image sequence may contain motion artefacts, registration methods, which estimate the artefactual motion and compensate for it, may be expected to increase the temporal correlation between the images and hence the temporal decorrelation efficiency. We shall evaluate this expectation by considering both unregistered (i.e., not motion-compensated) decorrelation and motion-compensated decorrelation. Motion estimation in image sequences can be accomplished by two principally different approaches: intensity matching and feature matching. In intensity matching the pixel values (in the case of 2-D images) of the two images to be registered are compared so as to yield a displacement vector field. In feature matching the locations of well defined features in the two images are compared; the features may be either low-level features as edges, corners, etc., or high-level features like entire objects. In either case, the estimation procedure yields a relative motion between the two images which is corrected for by means of some, usually affine, transformation.

In this paper, we consider only intensity matching, since this approach has proven successful in registering medical image sequencies, in particular sequences of angiographic images, [6]–[8]. Distinction is made between local estimation methods, also known as pixel-recursive or pel-recursive methods, and block motion estimation

Manuscript received September 4, 1990; revised May 12, 1991.

P. Roos was with the Faculty of Technical Mathematics and Informatics, Delft University of Technology, Delft, The Netherlands. He is now with the Faculty of Medicine, University of Utrecht, Utrecht, The Netherlands.

M. A. Viergever is with the Faculty of Medicine, University of Utrecht, Utrecht, The Netherlands.

IEEE Log Number 9102722.

methods; see Mussman [9] and Jain [10] for a review. Furthermore, any temporal decorrelation scheme may be based on either extrapolative or interpolative prediction. The division into extrapolation- and interpolation-based decorelation on the one hand, and unregistered, locally motion-compensated, and block motion-compensated decorrelation on the other hand, gives rise to six classes of decorrelation methods. In Section II extrapolation- and interpolation-based frame coding is described, in Section III unregistered decorrelation techniques are presented. In Section IV, block motion-compensated decorrelation is described in extrapolation and interpolation form and Section V discusses local motion-compensated decorrelation in extrapolated and interpolated form. The various decorrelation techniques are applied to a number of medical image sequences and one nonmedical image sequence in Section VI. The results are somewhat surprising, in particular the fact that motion registration is detrimental for efficient image compression in most of the cases considered. This phenomenon, as well as other characteristics of the results, are discussed in Section VII. The final Section VIII gives a summary of the conclusions of this study on reversible interframe decorrelation.

II. Extrapolation- and Interpolation-Based Frame Coding

In temporal decorrelation methods, distinction can be made between extrapolation-based and interpolation-based schemes. Extrapolated frame coding produces an estimate of each frame, apart from the first one, from previous frames. In this paper, we confine ourselves to schemes which are based on the previous frame only:

$$\hat{f}(\boldsymbol{x}, t) = \mathfrak{F}_E(f(\boldsymbol{x}, t-1)) \qquad t = 1, 2, 3, \cdots, \quad (1)$$

where $f(\boldsymbol{x}, t)$, $t = 0, 1, 2, \cdots$ is a temporal sequence of 2D images, $\hat{f}(\boldsymbol{x}, t)$ is an estimate of $f(\boldsymbol{x}, t)$, and $\mathfrak{F}_E$ is some function of $f(\boldsymbol{x}, t-1)$; $\boldsymbol{x}$ is a spatial coordinate vector. In frame interpolation a number of temporal resolution levels can be distinguished. At the highest temporal resolution level the images at odd sampling times ($t = 1, 3, 5, 7, \cdots$) are estimated:

$$\hat{f}(\boldsymbol{x}, 2t+1) = \mathfrak{F}_I(f(\boldsymbol{x}, 2t), \ f(\boldsymbol{x}, 2t+2)) \qquad t = 0, 1, 2, 3, \cdots, \quad (2)$$

where $\mathfrak{F}_I$ is some interpolation function. At the second level (lower temporal resolution) the images at time $t = 2, 6, 10, 14, \cdots$ are estimated:

$$\hat{f}(\boldsymbol{x}, 4t+2) = \mathfrak{F}_I(f(\boldsymbol{x}, 4t), \ f(\boldsymbol{x}, 4t+4)) \qquad t = 0, 1, 2, 3, \cdots. \quad (3)$$

This procedure may be continued for lower resolution levels. The overall interpolation estimator can be written in hierarchical form:

$$\hat{f}(\boldsymbol{x}, 2^l(2t+1)) = \mathfrak{F}_I(f(\boldsymbol{x}, 2^{l+1}t), \ f(\boldsymbol{x}, 2^{l+1}(t+1))) \qquad l, t = 0, 1, 2, 3, \cdots \quad (4)$$

where the integer l denotes the temporal resolution level of the interpolative estimation, with $l = 0$ at the highest level. Note that (4) is in fact one-dimensional HINT applied in the temporal direction. The hierarchical prediction steps may proceed in arbitrary order, the reconstruction, however, must proceed from the lowest level of resolution level to the highest.

In both extrapolation- and interpolation-based schemes it is not possible to (temporally) decorrelate all frames without violating the causality condition which ensures reversibility. For example, in extrapolated coding the first frame cannot be estimated, while in interpolated coding at least two frames cannot be decorrelated temporally. To obtain optimal overall reversible compression of the series, these basis frames are coded intraframe using HINT.

III. Unregistered Interframe Decorrelation

If the motion between consecutive frames in the sequence is not compensated for, the decorrelation schemes become very straightforward. In extrapolation-based coding, the only reasonable (and statistically significant) estimate is

$$\hat{f}(\boldsymbol{x}, t) = f(\boldsymbol{x}, t-1) \qquad t = 1, 2, 3, \cdots. \quad (5)$$

In interpolation-based frame coding, the estimation scheme is simply

$$\hat{f}(\boldsymbol{x}, 2^l(2t+1)) = \tfrac{1}{2}(f(\boldsymbol{x}, 2^{l+1}t) + f(\boldsymbol{x}, 2^{l+1}(t+1))) \qquad l, t = 0, 1, 2, 3, \cdots. \quad (6)$$

The difference image to be coded is given by

$$E(\boldsymbol{x}, t) = f(\boldsymbol{x}, t) - \mathrm{NINT}(\hat{f}(\boldsymbol{x}, t)) \quad (7)$$

where NINT is nearest integer rounding.

The simulation results of both schemes are presented in Section VI. In Figs. 1–3 an example is given of an original frame (this frame is taken from Series I, which is used later on in Section VI), an extrapolated difference frame and an interpolated difference frame. (Note that the difference images have been shifted and scaled up for illustration purposes only).

IV. Block Motion-Compensated Decorrelation

A. Extrapolation Schemes

In block matching the image is segmented into a fixed number of rectangular (overlapping or nonoverlapping) blocks. The assumption is made that all picture elements of one block have the same distortion, which yields one displacement vector $\boldsymbol{D}$ per block. A displacement vector is obtained by minimizing

$$d(f(\boldsymbol{x} - \boldsymbol{D}, t-1), \ f(\boldsymbol{x}, t)) \quad (8)$$

where d represents some image distance measure. Examples of image distance measures are normalized root mean squared and normalized mean absolute distance measures [11], (normalized) cross-correlation (e.g., [12]) and the discrete sign change (DSC) criterion [13]. We will

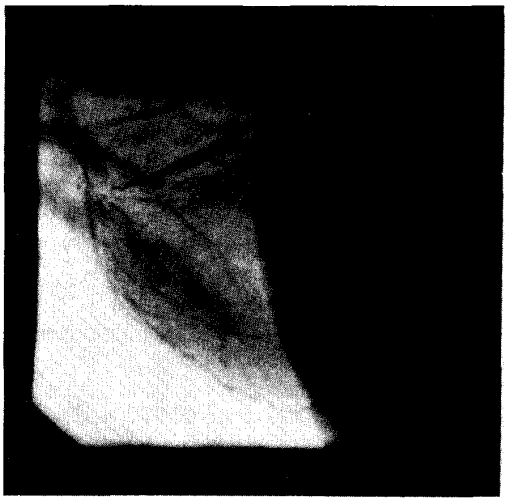

Fig. 1. One frame of a coronary angiogram time sequence (512 × 512 pixels, 512 gray values, recorded at 30 images/s; this sequence is "Series I" in Section VI).

Fig. 2. An unregistered extrapolated difference frame (Series I). The image has been shifted and scaled for display purposes.

use the DSC criterion since Venot has shown that this image measure is more robust than the other distance measures mentioned [13]. The DSC criterion calculates the number of sign changes in a block (scanned line by line or column by column) of the difference image

$$f(x_1, x_2, t) - f(x_1, D_1, x_2 - D_2, t - 1) + q(-1)^{x_1 - x_2},$$
$$t = 1, 2, 3, \cdots \tag{9}$$

where x_1, x_2 are the coordinates of $\boldsymbol{x}$ and D_1, D_2 the coordinates of $\boldsymbol{D}$, while the scalar q represents the depth of the "chessboard pattern" added to the subtracted image. The match is optimal if the DSC value reaches its maximum. Note that in this model only translation is dealt with. In order to fully extract in-plane motion artefacts, rotation (uniform) scaling and shear should also be included in the optimization procedure, but this would imply a prohibitive increase of computation time. Venot *et al.* [6] and Zuiderveld *et al.* [8] have shown that the restriction to translation produces acceptable results in acceptable computation times. Of course, any motion of the 3-D scene perpendicular to the projection plane of the time sequence is disregarded. Finally it should be noted that gray-level offsets between frames have been included in the registration procedure.

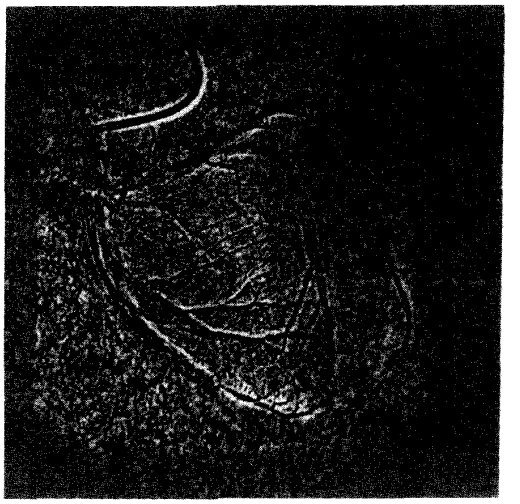

Fig. 3. An unregistered interpolated difference frame (Series I). The image has been shifted and scaled for display purposes.

A local optimum of the DSC value can be obtained by using search techniques such as Hooke and Jeeves [14], logarithmic search [15], or conjugate direction search [16]. We found that for all three methods at least 30 percent of the obtained displacement vectors did not globally optimize the DSC criterion. We therefore simply used a brute force technique by scanning all possibilities (within a sufficiently large window) to ensure globality of the optimum.

The rectangluar grid of displacement vectors is bilinearly interpolated to obtain a displacement vector per pixel. The difference image

$$E(\boldsymbol{x}, t) = f(\boldsymbol{x}, t) - \mathrm{NINT}(f(\boldsymbol{x} - \boldsymbol{D}, t - 1)) \tag{10}$$

is coded together with the block displacement vectors to obtain a fully reversible block displacement compensation method. The number of block displacement vectors is a trade-off. By increasing the number of vectors a more local but also noisier estimate is obtained, while moreover the additional storage of these vectors may undo any improvement in decorrelation efficiency.

B. Interpolation Schemes

In block motion-compensated interpolation the images $f(\boldsymbol{x}, t - 1)$ and $f(\boldsymbol{x}, t + 1)$ are subdivided into a fixed number of rectangular blocks. The assumption is made that the motion is time-independent in the interval $(t - 1,$

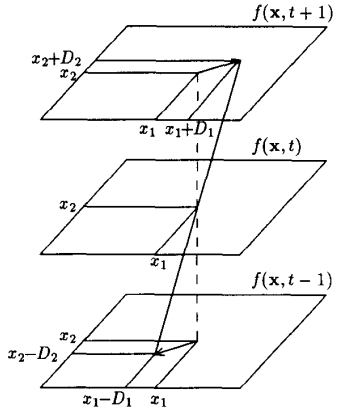

Fig. 4. In symmetrical motion-compensated interpolation a displacement vector $\boldsymbol{D}$ for the frame $f(x, t)$ at time t is estimated from the frames at time $t - 1$ and $t + 1$.

$t + 1$). This allows the calculation of a displacement vector for each block by minimizing

$$d(f(\boldsymbol{x} - \boldsymbol{D}, t - 1), \quad f(\boldsymbol{x} + \boldsymbol{D}, t + 1)) \tag{11}$$

where d is the DSC criterion, see Fig. 4. Analogously to block extrapolation we simply scanned all possibilities to ensure globality of the optimum. A complete displacement vector field is obtained by bilinear interpolation of the block displacement vectors. The difference image to be coded is given by

$$E(\boldsymbol{x}, t) = f(\boldsymbol{x}, t) - \text{NINT}\,(\tfrac{1}{2}(f(\boldsymbol{x} - \boldsymbol{D}, t - 1) + f(\boldsymbol{x} + \boldsymbol{D}, t + 1))). \tag{12}$$

The calculation of the displacement vector field is solely based on $f(\boldsymbol{x}, t - 1)$ and $f(\boldsymbol{x}, t + 1)$ and thus no additional storage of the block displacement vectors is required. In the reconstruction of $f(\boldsymbol{x}, t)$ the block displacement vectors are simply recalculated.

V. Local Motion-Compensated Decorrelation

A. Extrapolation Schemes

In local (or recursive) motion estimation a displacement vector is calculated for each pixel. Thus a vector field is obtained where all geometrical in-plane distortions are described by local translations. The method assumes that for each pixel $\boldsymbol{x}$ there is a distortion vector $\boldsymbol{D}(\boldsymbol{x})$ such that

$$f(\boldsymbol{x}, t) = f(\boldsymbol{x} - \boldsymbol{D}(\boldsymbol{x}), t - 1) \qquad t = 1, 2, 3, \cdots. \tag{13}$$

A "best" estimate of $\boldsymbol{D}(\boldsymbol{x})$ is obtaind by minimizing the displaced frame difference (DFD) in some sense. The DFD is defined by

$$\text{DFD}(\boldsymbol{x}, \hat{\boldsymbol{D}}) = f(\boldsymbol{x}, t) - f(\boldsymbol{x} - \hat{\boldsymbol{D}}, t - 1) \qquad t = 1, 2, 3, \cdots \tag{14}$$

where $\hat{\boldsymbol{D}}$ is some estimate of $\boldsymbol{D}$. The DFD can be linearized by combining (13) and (14):

$$\begin{aligned}\text{DFD}(\boldsymbol{x}, \hat{\boldsymbol{D}}) &= f(\boldsymbol{x} - \boldsymbol{D}, t - 1) - f(\boldsymbol{x} - \hat{\boldsymbol{D}}, t - 1) \\ &= -(\boldsymbol{D} - \hat{\boldsymbol{D}})^T \nabla f(\boldsymbol{x} - \hat{\boldsymbol{D}}, t - 1) \\ &\quad + \mathcal{O}(\boldsymbol{D} - \hat{\boldsymbol{D}})^2 \\ &= -\nabla^T f(\boldsymbol{x} - \hat{\boldsymbol{D}}, t - 1)\,(\boldsymbol{D} - \hat{\boldsymbol{D}}) \\ &\quad + \mathcal{O}(\boldsymbol{D} - \hat{\boldsymbol{D}})^2\end{aligned} \tag{15}$$

where $\mathcal{O}$ is the Landau order symbol.

In straightforward motion estimation the DFD is minimized at position $\boldsymbol{x}$ to find the desired estimate of the distortion $\boldsymbol{D}(\boldsymbol{x})$. For the purpose of (reversible) compression, however, this would be impractical since it would require the storage of the complete displacement vector field. By calculating the DFD at already coded pixels only, a causal estimate of $\boldsymbol{D}(\boldsymbol{x})$ can be obtained. The estimation is done using a window containing pixels neighbouring the pixel to be coded. The number of pixels at which the DFD is calculated ("observations") is a tradeoff; the displacement estimate of a (spatially) uniformly moving object will benefit from a large number of observations, whereas a small number of observations leads to a local but noisy estimate. In the reconstruction the DFD value is computed from the already decoded pixels.

From (15) it follows the N observations at pixels $\boldsymbol{x}(j)\, j = 1, 2, \cdots, N$ take the form:

$$\begin{aligned}\text{DFD}(\boldsymbol{x}(1), \hat{\boldsymbol{D}}) &= -\nabla^T f(\boldsymbol{x}(1) - \hat{\boldsymbol{D}}, t - 1)\,(\boldsymbol{D} - \hat{\boldsymbol{D}}) + \mathcal{O}(\boldsymbol{D} - \hat{\boldsymbol{D}})^2 \\ &\vdots \qquad\qquad \vdots \qquad\qquad \vdots \\ \text{DFD}(\boldsymbol{x}(N), \hat{\boldsymbol{D}}) &= -\nabla^T f(\boldsymbol{x}(N) - \hat{\boldsymbol{D}}, t - 1)\,(\boldsymbol{D} - \hat{\boldsymbol{D}}) + \mathcal{O}(\boldsymbol{D} - \hat{\boldsymbol{D}})^2.\end{aligned} \tag{16}$$

In matrix/vector notation (16) is rewritten as

$$z = H^T(\boldsymbol{D} - \hat{\boldsymbol{D}}) + \boldsymbol{v} \tag{17}$$

where the vector z represents the N observations of the DFD, H^T is the $N \times 2$ matrix of gradients, and $\boldsymbol{v}$ is a vector representing the Taylor expansion truncation error.

In order to calculate an optimum causal estimate, an iterative procedure is followed which minimizes the DFD, and hence the truncation residue $\boldsymbol{v}$, in some sense. Let $\boldsymbol{D}_k$ be the kth value of $\hat{\boldsymbol{D}}$ in the iteration process, and z_k, H_k^T the corresponding values of z and H^T. By using the calculated distortion vector as the next estimate an implicit difference equation is obtained as follows:

$$z_k = H_k^T(\boldsymbol{D}_{k+1} - \boldsymbol{D}_k). \tag{18}$$

Netravali and Robbins [17], [8] minimize the truncation residue $\boldsymbol{v}$ by means of a weighted gradient method, which

is related to the SIRT algorithm [19] used in image reconstruction from projections, see the Appendix.

The resulting iteration procedure is

$$\begin{aligned} \boldsymbol{D}_{k+1} &= \boldsymbol{D}_k - \epsilon_k \sum_j W(j)\, z_k(j) H_k(j) \\ &= \boldsymbol{D}_k - \epsilon_k \sum_j W(j)\, \mathrm{DFD}(\boldsymbol{x}(j), \boldsymbol{D}_k) \nabla f(\boldsymbol{x}(j) \\ &\quad - \boldsymbol{D}_k, t-1) \end{aligned} \tag{19}$$

where ϵ_k is a relaxation constant (which was taken to be independent of k by Netravali and Robbins), $H_k(j)$ denotes the jth column of H_k, and $W(j)$, $j = 1, 2, \cdots, N$, is a set of weight coefficients, with

$$\sum_j W(j) = 1. \tag{20}$$

An example of a three point causal weight-coefficient window is

$$\begin{pmatrix} w(1) & w(2) \\ w(3) & \cdot \end{pmatrix} \tag{21}$$

where the dot represents the position of the pixel to be coded. The choice of ϵ_k influences the rate of convergence and the stability, but not necessarily the quality of the estimation [20]. Walker and Rao [21] use a nonconstant ϵ_k, which is small if the gradient is large (e.g., at the edge of an object) and large for small gradients:

$$\epsilon_k(\boldsymbol{x}, \boldsymbol{D}) = \frac{1}{2\|\nabla f(\boldsymbol{x} - \boldsymbol{D}, t-1)\|_2^2}. \tag{22}$$

The resulting iterative displacement estimation scheme is

$$\boldsymbol{D}_{k+1} = \boldsymbol{D}_k - \sum_j W(j) \epsilon_k(\boldsymbol{x}(j), \boldsymbol{D}_k) z_k(j) H_k(j). \tag{23}$$

In the derivation of the above methods a least square estimate of the DFD has been sought. Consequently, is has been assumed implicitly that the truncation residue $\boldsymbol{v}$ in (17) can be considered as additive white zero-mean Gaussian noise. By assuming this noise behavior explicitly, Biemond *et al.* [22] derive a regularized least squares (Wiener) estimate of the update, which reads

$$\boldsymbol{D}_{k+1} = \boldsymbol{D}_k + (H_k H_k^T + \mu I)^{-1} H_k z_k \tag{24}$$

where μ is a regularization parameter, representing the ratio of the update variance and the noise variance. In Fig. 5, an example of a different frame (Series I, see Section VI) resulting from Wiener-based extrapolation is given.

Fig. 5. A Wiener-based motion compensated extrapolated difference frame (Series I). The image has been shifted and scaled for display purposes.

B. Interpolation Schemes

The assumption made in (13) is equivalent to the assumption that for each pixel $\boldsymbol{x}$ a distortion vector $\boldsymbol{D}'(\boldsymbol{x})$ exists such that

$$f(\boldsymbol{x}, t) = f(\boldsymbol{x} + \boldsymbol{D}'(\boldsymbol{x}), t+1) \quad t = 0, 1, 2, \cdots. \tag{25}$$

For objects moving at constant speed the distortion vectors $\boldsymbol{D}(\boldsymbol{x})$ and $\boldsymbol{D}'(\boldsymbol{x})$ are equal. This admits the definition of a symmetrical displaced frame difference (SDFD) by

$$\mathrm{SDFD}(\boldsymbol{x}, \hat{\boldsymbol{D}}) = f(\boldsymbol{x} + \hat{\boldsymbol{D}}, t+1) - f(\boldsymbol{x} - \hat{\boldsymbol{D}}, t-1), \tag{26}$$

where $\hat{\boldsymbol{D}}$ is an estimate of $\boldsymbol{D}$.

Linearizing the SDFD using (13) and (25) and the equality of $\boldsymbol{D}$ and $\boldsymbol{D}'$ yields:

$$\begin{aligned} \mathrm{SDFD}(\boldsymbol{x}, \hat{\boldsymbol{D}}) &= f(\boldsymbol{x} + \hat{\boldsymbol{D}}, t+1) - f(\boldsymbol{x} + \boldsymbol{D}, t-1) \\ &\quad + f(\boldsymbol{x} - \boldsymbol{D}, t-1) - f(\boldsymbol{x} - \hat{\boldsymbol{D}}, t-1) \\ &= -(\boldsymbol{D} - \hat{\boldsymbol{D}})^T (\nabla f(\boldsymbol{x} + \hat{\boldsymbol{D}}, t+1) \\ &\quad + \nabla f(\boldsymbol{x} - \hat{\boldsymbol{D}}, t-1)) + \mathcal{O}(\boldsymbol{D} - \hat{\boldsymbol{D}})^2 \\ &= -\nabla^T (f(\boldsymbol{x} + \hat{\boldsymbol{D}}, t+1) \\ &\quad + f(\boldsymbol{x} - \hat{\boldsymbol{D}}, t-1))(\boldsymbol{D} - \hat{\boldsymbol{D}}) \\ &\quad + \mathcal{O}(\boldsymbol{D} - \hat{\boldsymbol{D}})^2. \end{aligned} \tag{27}$$

Analogously to extrapolated recursive displacement estimation a number of observations are made. These observations however need not be causal, since no information of $f(\boldsymbol{x}, t)$ is used. We have

$$z = G^T(\boldsymbol{D} - \hat{\boldsymbol{D}}) + \boldsymbol{w} \tag{28}$$

where the vector z represents N observations of the SDFD, G^T is the $N \times 2$ matrix of gradients, and $\boldsymbol{w}$ is the vector of truncation errors.

As in Section V-A an iterative procedure is followed: let $\boldsymbol{D}_k$ be the kth value of $\hat{\boldsymbol{D}}$ in the iteration process, and z_k, H_k^T the corresponding values of z and H^T. The difference equation then becomes

$$z_k = G_k^T(\boldsymbol{D}_{k+1} - \boldsymbol{D}_k), \tag{29}$$

and the weighted gradient minimization takes the form

$$\boldsymbol{D}_{k+1} = \boldsymbol{D}_k - \sum_j \epsilon_k W(j) z_k(j) G_k(j) \tag{30}$$

Fig. 6. A difference frame resulting from a Wiener-based motion compensated interpolation (Series I). The image has been shifted and scaled for display purposes.

where ϵ_k is constant for the Netravali and Robbins method and defined by (22) for the Walker and Rao method.

Wiener-based pel-recursive interpolation yields a regularized least squares estimate of the update of the symmetrical displacement vector

$$\boldsymbol{D}_{k+1} = \boldsymbol{D}_k + (G_k G_k^T + \mu I)^{-1} G_k z_k. \tag{31}$$

In Fig. 6, an example is given of a difference frame (Series I), resulting from Wiener-based interpolation.

VI. Results

We have evaluated the above described methods on several medical image sequences and one nonmedical image sequence. Series I, II, and III consist of coronary X-ray angiograms. Series I is a set of 15 frames of 512 × 512 pixels, 9 b (512 gray values), recorded at 30 images per second. Series II consists of 14 152 × 1512 frames, 8 b, recorded at 25 images/s. Series III consists of 18 frames of 256 × 256, 8 b, recording speed 25 images/s. Series IV is a set of 18 ventricle X-ray angiograms, 256 × 256, 8 b recorded at 12.5 images/s. Series V is a liver scintigram image sequence for the detection of Meckel's diverticle. The series contains 40 frames of 128 × 128, 8 b, recording speed 1 image/min. Series VI is a video conferencing time series of 20 frames of 256 × 256, 8 b, recorded at 60 images/s. Although reversible compression is not very appropriate for the latter type, the series is included for the sake of comparison.

In Table I, the average entropies (in bits/pixel) resulting from the various interframe decorrelation schemes and from straightforward intraframe HINT are given for the six image sequences. In Figs. 7 and 8, the results of the extrapolation-based methods are presented in pictorial form. The differences between the average entropies produced by the interpolation-based methods are too small to make a pictorial presentation suitable.

In temporal interpolation the number of hierarchical steps used depends on both the spatial and the temporal correlation. For the situation of low-spatial and high-temporal correlation the number of hierarchical steps should be large. In this situation it is advantageous to interpolate images which are (temporally) far apart. For the five medical image sequencies the optimum value of the parameter l in (4) was found to be 0; only half of the images are temporally decorrelated, the other half are 2-D HINT decorelated. For the video conferencing series, which is the only series with a ratio of temporal to spatial correlation larger than 1 (see Fig. 9), $l = 2$ is optimum; every 8th frame is 2-D HINT decorrelated.[1]

In block motion estimation, described in Sections IV-A and B, an equidistant grid of block displacement vectors is calculated. The DSC distance is calculated over an area of 31 × 31 pixels, each grid point represents an area of 16 × 16 pixels. The overlapping blocks will produce a global motion estimate. The block displacement vectors are stored at 24 b/vector, thus creating an overload of $24/16^2 \approx 0.09$ b/pixel. This overload has already been taken into account in Table I. The q parameter in the DSC criterion depends on the signal-to-noise ratio of the images. The optimal q is a tradeoff: a small q leads to a displacement sensitive but noisy estimate, a large q yields a noise- and motion-insensitive estimate. In the simulation results the q is tuned per image sequence, the optimal values are 4, 4, 8, 8, 6, 12 for image sequences I through VI.

The three extrapolated pixel-recursive motion estimation methods described in Section V-A use a causal window from which the displacement information is obtained. The optimal window size depends on the correlation of the displacement vectors. Global motion is detected for large windows whereas a small window leads to a local yet noisy motion estimate. We found that a window of three pixels, configurated as shown in (21) is optimal for the medical image sequences. For the video sequence a slight improvement can be obtained by using a 5-point window. Conventionally, the obtained displacement vector is used as a starting vector for the recursive estimation of the next displacement vector. We found, however, that for the medical image sequences this leads to a less efficient scheme than the memoryless model where the starting vector is set equal to zero. The optimal ϵ in the Netravali and Robbins extrapolated method turned out to be 0.005 for all images series, for the interpolation variant the optimal value was found to be 0.001. The parameter μ in the Wiener-based pel-recursive method is a damping parameter. For the extrapolated scheme the optimum values for μ are 10, 1, 5, 10, 10, 100 for image sequences I through VI. For interpolated Wiener-based motion compensation the optimum μm is equal to 90 for all sequences.

In reversible interframe compression the temporal decorrelation can be combined with spatial (reversible) decorrelation without any restriction. From the methods de-

[1]Note that entropy numbers in Table I represent the average entropy for the complete image sequences; thus they are based on the entropy of both temporally and spatially decorrelated frames.

TABLE I
AVERAGE ENTROPY (b/pixel) OF SIX IMAGE SERIES AFTER INTRAFRAME HINT DECORRELATION AND VARIOUS INTERFRAME DECORRELATION TECHNIQUE

	I Coronary Angiogram 512 × 512 30 im/s	II Coronary Angiogram 512 × 512 25 im/s	III Coronary Angiogram 256 × 256 25 im/s	IV Ventricle Angiogram 256 × 256 12.5 im/s	V Liver Scintigram 128 × 128 1 im/mm	VI Video Conferencing 256 × 256 60 im/s
Quantization Level (b/pixel)	9	8	8	8	8	8
Average Entropy before Decorrelation	7.57	6.26	6.95	7.21	3.58	6.72
Intraframe HINT	2.68	2.52	3.45	3.59	2.93	4.45
Extrapolation						
Unregistered	*3.24*	*3.20*	*3.75*	*4.15*	*3.21*	*4.03*
	2.59	2.71	3.53	3.65	3.27	3.90
Block-based	*3.23*	*3.30*	*3.78*	*4.14*	*3.30*	*4.07*
	2.74	2.86	3.65	3.81	3.36	3.97
Netravali & Robbins	*2.90*	*3.02*	*3.50*	*3.80*	*3.11*	*3.87*
	2.69	2.70	3.50	3.63	3.14	3.88
Wiener	*2.69*	*2.85*	*3.44*	*3.68*	*3.11*	*3.87*
	2.61	2.74	3.53	3.65	3.18	3.93
Interpolation						
Unregistered	*2.78*	*2.71*	*3.43*	*3.71*	*3.01*	*3.98*
	2.54	2.53	3.39	3.69	3.05	3.90
Block-based	*2.80*	*2.73*	*3.44*	*3.76*	*3.01*	*3.96*
	2.58	2.56	3.40	3.58	3.05	*3.88*
Netravali and Robbins	*2.84*	*2.73*	*3.47*	*3.74*	*2.99*	*4.18*
	2.66	2.58	3.45	3.62	3.02	4.16
Wiener	*2.77*	*2.70*	*3.41*	*3.68*	*2.99*	*3.96*
	2.55	2.53	3.38	3.53	3.02	3.89

The figures in italic font denote the entropy after temporal decorrelation only, those in roman font denote the entropy after temporal decorrelation followed by intraframe HINT applied to the individual frames. In all intraframe HINT decorrelations we used an initial blocksize of 8 × 8, resulting in a 7-level pyramid. The Walker and Rao method described in Section V-A; B has been omitted, since the results were always in between the Netravali and Robbins and Wiener-based pel-recursive methods both for interpolation and extrapolation.

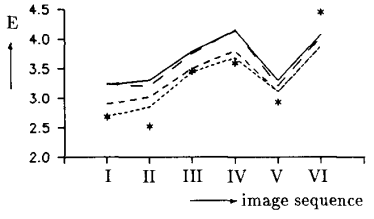

Fig. 7. The average entropy (bits/pixel) of the six image Series I–VI after extrapolation-based temporal decorrelation.[2] ——Block-based extrapolation; — —unregistered extrapolation; ----Netravali and Robbins; - - - - Wiener-based motion compensation. The results of straightforward intraframe HINT (*) are also shown for the six series.

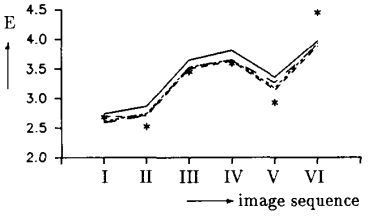

Fig. 8. The average entropy (bits/pixel) of the six image Series I–VI after extrapolation-based temporal decorrelation and consecutive intraframe HINT.[2] —— Block-based extrapolation; — —unregistered extrapolation; ----Netravali and Robbins; - - - - Wiener-based motion compensation. The results of straightforward intraframe HINT (*) are also shown for the six series.

scribed in [1] HINT turned out to be the best post spatial decorrelator. See Table I for the post-HINT results (roman font). Reversing the order of temporal and spatial decorrelation, which was found to be useful in irreversible compression [23], has also been considered. We found that for the medical image sequences a spatial-temporal scheme is better than temporal decorrelation, but less efficient than the above outlined temporal-spatial scheme; for details we refer to [24].

Finally, we have calculated the correlation coefficient (CC) in the two spatial dimensions and in the temporal dimensions for the six image sequencies considered. The CC [12] is defined by

$$\rho_{klm} = \frac{N_{\text{tot}} \sum_{x_1,x_2,t} hg - \left(\sum_{x_1,x_2,t} h\right)\left(\sum_{x_1,x_2,t} g\right)}{\sqrt{\left(N_{\text{tot}} \sum_{x_1,x_2,t} h^2 - \left(\sum_{x_1,x_2,t} h\right)^2\right)\left(N_{\text{tot}} \sum_{x_1,x_2,t} g^2 - \left(\sum_{x_1,x_2,t} g\right)^2\right)}} \tag{32}$$

[2]The frames which are not temporally decorrelated have been spatially decorrelated using 2-D HINT.

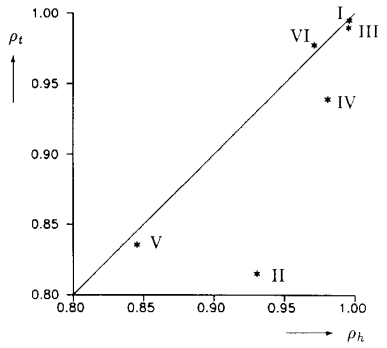

Fig. 9. The temporal correlation coefficient versus the horizontal spatial correlation coefficient for the six medical image sequences considered

where $h = f(x_1, x_2, t)$ and $g = f(x_1 + k, x_2 + l, t + m)$, and where N_{tot} is the total number of elements over which the summation is calculated. In Fig. 9 the temporal CC (ρ_{001}) is set out against the horizontal spatial CC (ρ_{100}). (The vertical CC (ρ_{010}) was found to be almost equal to the horizontal CC.) As is clear from the figure, the videoconferencing sequence has the highest ratio of temporal to spatial correlation.

VII. Discussion of the Results

From Table I we conclude that the Wiener-based pel-recursive method is the most efficient method for motion compensation both in extrapolated and in interpolated form. It outperforms block-based motion compensation, the Netravali and Robbins method of pel-recursive motion compensation, and the Walker and Roa method. The spatial postdecorrelator, however, destroys the optimality of the Wiener approach. Motion compensation destroys the spatial correlation at least partly. In some cases this effect is seen to be so large that the entropy even increases if a spatial decorrelator is applied to the temporally decorrelated images.

For the medical image sequences I through V the interpolation scheme is generally more efficient than the corresponding extrapolation scheme, both for motion-compensated and nonmotion-compensated decorrelation. This can be explained by the averaging character of the estimation in interpolative decorrelation which suppresses the temporal noise. This noise suppression effect is stronger than the disadvantage of interpolation frames that not all frames can be decorrelated temporally.

The above discussed methods are judged solely on their decorrelation performance (resulting average entropy). Another aspect however is the complexity of the various methods. The methods discussed can be divided into two classes. The first class contains the unregistered methods and intraframe 2-D HINT. These methods have in common that the execution time is independent of the image contents. In contrast, for the motion compensation methods, the execution time depends on the amount of iterations needed to obtain a suitable displacement vector field.

For the medical image sequences one of the interframe decorrelation methods performs significantly better than straightforward intraframe HINT. However, unregistered interpolation approximates 2-D HINT within a few percent as regards compression ratio. We timed both methods on an HP 375, 50 MHz, 68030 processor. For unregistered interpolation we made the assumption that each frame needs to be read only once. Since the method accesses the frames in a nonsequential order, a sufficiently large internal memory is required. The minimal size of the internal memory depends on l in (4); the number of frames needed to be stored is $2^{(l+1)} + 1$. The 2-D HINT method required 0.80 sec per frame (512 × 512 pixels). Unregistered interpolation takes 0.32 s/frame. Since only half the series is temporally decorrelated in interpolation methods with $l = 0$, while the other half is spatially decorrelated using 2-D HINT, the computational cost of unregistered interpolation is approximately 2/3 of that of intraframe HINT. Consequently, it is an interesting alternative for intraframe HINT.

If computational speed is more important than the number of bits of coded, unregistered extrapolation could be the method of choice. It approximates the average entropy produced by 2-D HINT not as well as unregistered interpolation, but for many applications the resulting bit rate might still be acceptable. Unregistered extrapolation takes 0.14 s/frame. Since all image frames but the first are temporally decorrelated, the computational cost of unregistered extrapolation is approximately 1/5 of that of intraframe HINT.

The performance of a decorrelation method depends on the correlation of the signal. In intraframe HINT the two spatial correlations are utilized while the temporal decorrelation step of nonmotion compensated extrapolation and interpolation methods solely utilizes the interframe correlation. The lack of success of the motion compensated methods may be due to the low ratio of temporal correlation to spatial correlation, or, more probably, to the poor displacement estimates provided by the motion estimation (effects not included are, inter-alia, illumination changes, motion perpendicular to the frames, in-plane motion other than linear translational motion).

The influence of the spatial sampling rate can be observed by spatially downsampling the images. For images sequence I we downsampled the images by a factor of two: the images were low-pass filtered (half the bandwidth) before downsampling so as to avoid aliasing. The performance of the optimal temporal decorrelation method (Wiener), intraframe HINT, and Wiener + HINT are given in Table II. For the other image sequences the results are similar. The entropy figures represent the average entropy for the complete series, they are an average of both interframe and intraframe decorrelated frames. The figures in *italic* font denote the entropy after temporal decorrelation only.

From Table II it appears that downsampling by a factor of two already makes interframe decorrelation quite efficient. Although it is obvious that the performance of intraframe HINT relative to the temporal decorrelation methods will decrease with decreasing spatial sampling rate, the degree of this decrease was surprisingly high to

TABLE II
INFLUENCE OF SPATIAL DOWNSAMPLING ON THE PERFORMANCE OF INTRAFRAME HINT, UNREGISTERED AND WIENER-BASED INTER- AND EXTRAPOLATION FOR IMAGE SEQUENCE I

	512 × 512	256 × 256
Intraframe HINT	2.68	3.42
Unregistered extrapolation	*3.24*/2.59	*3.17*/2.91
Unregistered interpolation	*2.78*/2.54	*3.11*/3.05
Wiener-based extrapolation	*2.69*/2.61	*2.94*/3.08
Wiener-based interpolation	*2.77*/2.55	*3.09*/3.04

The figures in italic font denote the entropy after temporal decorrelation only, those in roman font denote the entropy after temporal decorrelation followed by intraframe HINT (8 × 8 initial blocksize).

TABLE III
INFLUENCE OF TEMPORAL SMOOTHING ON THE PERFORMANCE OF INTRAFRAME HINT, UNREGISTERED AND WIENER-BASED INTER- AND EXTRAPOLATION FOR IMAGE SEQUENCE I

	Original	3-Point Moving Average
Intraframe HINT	2.68	2.43
Unregistered extrapolation	*3.24*/2.59	*2.29*/1.51
Unregistered interpolation	*2.78*/2.54	*2.09*/1.92
Wiener-based extrapolation	*2.69*/2.61	*1.93*/1.90
Wiener-based interpolation	*2.77*/2.55	*2.11*/1.95

The figures in italic font denote the entropy after temporal decorrelation only, those in roman font denote the entropy after temporal decorrelation followed by intraframe HINT (8 × 8 initial blocksize).

us. Furthermore, the spatial postdecorrelation is less efficient or even worse for the spatially downsampled sequence.

We examined the influence of temporal noise by smoothing the image sequences in the temporal direction. We used a 3-point moving average filter. In Table III the results of 2-D HINT, nonmotion compensated extrapolation and interpolation, and Wiener-based motion compensated interpolation and extrapolation are given for image sequence I. Again the figures in italic font denote the entropy after temporal decorrelation only. From Table III we conclude that for the temporally averaged sequences all temporal decorrelation methods considered outperform intraframe HINT significantly. The optimal method now is unregistered extrapolation followed by intraframe HINT, just as for the spatially downsampled version of the sequence (Table II).

The results of Table II and III as well as the results for the videoconferencing series in Table I show that the conclusions of the present study should be interpreted with care. While the conclusions drawn above (and summarized below) are fairly consistent for all medical image sequences considered, they are not generally valid for any type of image sequence. In particular may preprocessing of the sequence, e.g., by temporal averaging, significantly alter the efficiency of the compression methods.

VIII. SUMMARY OF CONCLUSIONS

Interpolated temporal decorrelation is generally better than extrapolated temporal decorrelation for the medical image sequences considered.

Wiener-based pel-recursive motion estimation is the most efficient method for motion compensation. However, Wiener-based decorrelation is hardly (if at all) more efficient than nonmotion compensated interframe decorrelation.

For the medical image sequences, none of the interframe decorrelation methods performs signficantly better than straightforward intraframe HINT. This is ascribed to the temporal noise present in the medical image sequence. Both spatial downsampling and temporal smoothing of the series reverse the above conclusion.

Unregistered interpolation approximates the performance of intraframe HINT within a few percent. The lower complexity makes the method an interesting alternative to intraframe HINT. Unregistered extrapolation has a significantly lower complexity still, but the decorrelation results are not as good as for the interpolation scheme. The requirements concerning compression ratio and computational speed will determine which method is most appropriate.

APPENDIX

In this Appendix it is derived how the iteration procedure (19) for determining the distortion vector $D(\boldsymbol{x})$ follows from minimization of the truncation of the Taylor series expansion of the DFD. Furthermore, it is shown that minimizing the DFD with a steepest descent technique is equivalent to applying the simultaneous iterative reconstruction technique (SIRT) to the nonlinear system of equations (17).

A. Steepest Descent

Let $f(x)$ be the function of be minimized. The steepest descent technique has the form

$$x_{k+1} = x_k - \lambda_k \nabla f(x_k) \tag{33}$$

where λ_k is a relaxation parameter. Note that the steepest descent technique is a least squares minimization method. Applying this technique to the $\mathrm{DFD}^2(\boldsymbol{x}, \hat{\boldsymbol{D}})$, that is, minimizing the DFD^2 as a function of $\hat{\boldsymbol{D}}$ results in

$$\begin{aligned} \boldsymbol{D}_{k+1} &= \boldsymbol{D}_k - \lambda_k \nabla_{\boldsymbol{D}_k} DFD^2(\boldsymbol{x}, \boldsymbol{D}_k) \\ &= \boldsymbol{D}_k - 2\lambda_k DFD(\boldsymbol{x}, \boldsymbol{D}_k) \nabla_{\boldsymbol{D}_k} (f(\boldsymbol{x}, t) \\ &\quad - f(\boldsymbol{x}, \boldsymbol{D}_k, t-1)) \\ &= \boldsymbol{D}_k - 2\lambda_k DFD(\boldsymbol{x}, \boldsymbol{D}_k) \nabla_{\boldsymbol{x}} f(\boldsymbol{x} - \boldsymbol{D}_k, t-1), \end{aligned} \tag{34}$$

for one observation at position $\boldsymbol{x}$. In order to make the prediction scheme causal, Netravali and Robbins replaced the update by a weighted average of updates, using a number of observations in a neighborhood W of the pixel to be coded

$$\begin{aligned} \boldsymbol{D}_{k+1} = \boldsymbol{D}_k - \epsilon_k \sum_{j \in W} W(j) DFD(\boldsymbol{x}(j), \boldsymbol{D}_k) \\ \cdot \nabla_{\boldsymbol{x}} f(\boldsymbol{x}(j) - \boldsymbol{D}_k, t-1) \end{aligned} \tag{35}$$

where $\epsilon_k = 2\lambda_k$.

B. Simultaneous Iterative Reconstruction Technique (SIRT)

Consider the system of linear algebraic equations:

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \tag{36}$$

where A is an $N \times 2$ matrix, $\boldsymbol{x} = 2 \times 1$ and $\boldsymbol{b} = N \times 1$. Applying SIRT to this set of linear equations results in [25]

$$\begin{aligned}\boldsymbol{x}_{k+1} &= (I - A^T\Lambda_k A)\boldsymbol{x}_k + \Lambda_k A^T\boldsymbol{b} \\ &= \boldsymbol{x}_k + \sum_{j=1}^{N} \lambda_{jk}(\boldsymbol{b}_j - A(j)\boldsymbol{x}_k)A(j)^T \end{aligned} \tag{37}$$

where $\Lambda_k = \mathrm{diag}(\lambda_{jk})$ is a diagonal matrix providing for relaxation and scaling, and $A(j)$ denotes the jth row of A. However, (17) which we wish to solve, is nonlinear; both A and $\boldsymbol{b}$ are dependent on $\boldsymbol{x}$.

We thus have the set of equations

$$A(\boldsymbol{x})\boldsymbol{x} = \boldsymbol{b}(\boldsymbol{x}). \tag{38}$$

A natural way to solve the nonlinear system using SIRT is to apply SIRT to the original system making one iteration step with zero starting vector, to update the system matrix and the data vector, and next to apply SIRT to the new system by again making one iteration step initialized by zero. This process is repeated until the solution is satisfactory. The reason that zero is a good starting vector for every new step is that the iterate x_k should approach zero as $k \to \infty$ in order to give a good estimate $\hat{\boldsymbol{D}}$ for $\boldsymbol{D}$. We thus obtain the iteration scheme:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \sum_{j=1}^{N} \lambda_{jk}\boldsymbol{b}_k(j)A_k^T(j). \tag{39}$$

By setting: $A = H^T$, $\boldsymbol{x} = \boldsymbol{D} - \hat{\boldsymbol{D}}$, $\boldsymbol{x}_k = \boldsymbol{D} - \boldsymbol{D}_k$, $\boldsymbol{b} = \boldsymbol{z}$, and $\lambda_{jk} = \epsilon_k W(j)$ in order to make (36) correspond with (17), we obtain as a solution to (17):

$$\boldsymbol{D}_{k+1} = \boldsymbol{D}_k - \epsilon_k \sum_{j=1}^{N} W(j)\boldsymbol{z}_k(j)H_k(j) \tag{40}$$

where ϵ_k is a relaxation parameter and $W(j)$ is a weight coefficient window providing for row scaling.

Acknowledgment

Sequences I, III, IV were provided by Dr. H. P. A. Haas and Dr. C. H. Slump of Philips Medical Systems, Best, The Netherlands. Sequence II was provided by Dr. J. H. D. Reider of the Thoraxcenter of the Erasmus University and the Academic Hospital Dijkzigt, Rotterdam, The Netherlands. Sequence V was provided by C. N. de Graaf of the Computer Vision Research group, University of Utrecht and University Hospital Utrecht, The Netherlands. The video times series was provided by Dr. J. Biemond of the Information Theory group at Delft University of Technology, The Netherlands. The series originated from British Telecom Laboratories, England.

References

[1] P. Roos, M. A. Viergever, M. C. A. van Dijke, and J. H. Peters, "Reversible intraframe compression of medical images," *IEEE Trans. Med. Imaging*, vol. 7, pp. 328-336, 1988.

[2] G. Q. Macquire, Jr., M. E. Noz, A. Bakker, K. Bijl, H. Didden, and J. P. J. de Valk, "Introduction to PACS for those interested in image processing," in *Information Processing in Medical Imaging*, C. N. de Graaf and M. A. Viergever, Eds. New York: Plenum, 1988, pp. 315-323.

[3] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Commun. ACM*, vol. 30, pp. 520-540, 1987.

[4] D. A. Huffman, "A method for the construction of minimum redundancy codes," *Proc. IRE*, vol. 40, pp. 1098-1101, 1952.

[5] J. Ziv and A. Lempel, "Compression of individual sequences via variable-rate coding," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 530-536, 1978.

[6] A. Venot and V. Leclerc, "Automated correction of patient motion and gray values prior to subtraction in digitized angiography," *IEEE Trans. Med. Imaging*, vol. MI-3, pp. 179-186, 1984.

[7] J. M. Fitzpatrick, J. J. Grefenstette, D. R. Pickens, M. Mazer, and J. M. Perry, "A system for image registration in digital subtraction angiography," in *Information Processing in Medical Imaging*, C. N. de Graaf and M. A. Viergever, Eds. New York: Plenum, 1988, pp. 415-436.

[8] K. J. Zuiderveld, B. M. ter Haar Romeny, and M. A. Viergever, "Fast rubber sheet masking for digital subtraction angiography," in *Science and Engineering of Medical Imaging*, M. A. Viergever, Ed., *Proc. SPIE*, vol. 1137, SPIE, Bellingham, pp. 22-30.

[9] H. G. Mussman, P. Pirsch, and H. J. Grallert, "Advances in picture coding," *Proc. IEEE*, vol. 73, pp. 523-547, 1985.

[10] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[11] G. T. Herman, *Image Reconstruction from Projections, The Fundamentals of Computerized Tomography*. New York: Academic, 1980.

[12] A. Rosenfeld and A. C. Kak, *Digital Picture Processing*. New York: Academic, 1982.

[13] A. Venot, J. F. Lebruchec, and J. C. Roucayrol, "A new class of similarity measures for robust image registration," *Comput. Vision. Graph. Imaging Proc.*, vol. 28, 1984, pp. 176-184.

[14] R. Hooke and T. A. Jeeves, "Dirct search solution of numerical and statistical problems," *JACM*, vol. 8, pp. 535-552, 1961.

[15] J. R. Jain and A. K. Jain, "Displacement measurements and its application in the interframe image coding," *IEEE Trans. Commun.*, vol. COM-29, pp. 1799-1806, 1981.

[16] R. Srinivasan and K. R. Rao, "Predictive coding based on efficient motion estimation," in *ICC'84 Conf. Rec.*, 1984, pp. 521-526.

[17] A. N. Netravali and J. D. Robbins, "Motion-compensated television coding; Part 1," *Bell Syst. Tech. J.*, vol. 58, pp. 631-670, 1979.

[18] —, "Motion-compensated coding: some new results," *Bell Syst. Tech. J.*, vol. 59, pp. 1735-1745, 1980.

[19] A. V. Lakshminarayanan and A. Lent, "Methods of least squares and SIRT in reconstruction," *J. Theory Biol.*, vol. 76, pp. 267-295, 1979.

[20] R. J. Moorhead II and S. A. Rajala, "Motion-compensated interframe coding," in *IEEE Proc. ICASSP'85*, pp. 347-351.

[21] D. R. Walker and K. R. Rao, "Improved pel-recursive motion-estimation," *IEEE Trans. Commun.*, vol. COM-32, pp. 1128-1134, 1984.

[22] J. Biemond, L. Looijenga, D. E. Boekee, and R. H. J. M. Plompen, "A pel-recursive Wiener-based displacement estimation algorithm," *Signal Process.*, vol. 13, pp. 399-412, 1987.

[23] J. H. Moon, J. S. Koh, S. D. Kim, and J. Kim, "Spatial-temporal prediction algorithm with pel recursive motion compensation," in *Conf. Rec. GLOBECOM'87*, 1987, pp. 70-72.

[24] P. Roos and M. A. Viergever, "Registration and reversible compression of angiographic image sequences," in *Medical Imaging III*, S. J. Dwyer, R. G. Jost, R. H. Schneider, Eds., *Proc. SPIE*, vol. 1092, SPIE, Bellingham, 1989, pp. 383-391.

[25] M. C. A. van Dijke, H. A. van der Vorst, and M. A. Viergever, "Iterative methods for sparse, linear systems arising from tomographic image reconstruction: A comparison between ART, block-ART, and SIRT," to appear in BIT.