TELEVISION NOISE REDUCTION IC

Philips Semiconductors Eindhoven, The Netherlands

ABSTRACT
A noise reduction IC for consumer television has been designed. The IC contains a spatial filter for Gaussian noise and a temporal filter for clamp noise. To reduce clamp noise, the average value of the pixels in a line-segment are filtered rather than individual pixels. This reduces the cost of the temporal filter significantly, enabling the use of embedded memory. Analog interfaces are provided, as well as a line-locked clock generator. The chip includes a noise level estimator for optimal filtering under varying reception conditions. The average gain of the spatial filter is around 3 dB, whereas the temporal filter yields up to 8 dB improvement on clamp noise.

1. INTRODUCTION
Noise reduction of image data is achieved through some form of linear or non-linear operation on correlated picture elements. The operation often involves (weighted) averaging in the case of additive white Gaussian noise, or order statistical filtering in the case of shot noise. The correlation may be due to temporal or spatial proximity, which leads to the choice of a certain filter support although, usually, additional criteria are applied to select a sub-set of pixels from this support. Known selection processes are “K-nearest neighbours”, “symmetric nearest neighbours”, and “sigma nearest neighbours”. Generally, the difference between the examined neighbour and the current pixel plays an important role in the selection process.

Consequently, the basic choices in image data noise filtering concern choice of operation, choice of filter support, and choice of selection criterion. For consumer applications the cost of the filter is of primary importance. This is influenced mainly by the choice of the filter support. We can distinguish between:

- 1-D filters: either spatial, temporal [1-4], or along the estimated motion trajectory (MC) [5,6]
- 2-D filters: either spatial [7-10] or (MC-) spatio-temporal
- 3-D filters: (MC-) spatio-temporal [9].

1. The IC is available commercially as SAA4985

As a consequence of the cost constraint, horizontal extensions of the support are the most attractive, with vertical extensions somewhat less and temporal being the least attractive. Motion compensation further decreases the attractiveness in this aspect. Nevertheless, temporal filters of the type shown in Figure 1, were the first to be introduced in flicker-free, or 100 Hz, television sets [1,2]. The reason is that the most costly delay element (the field memory) is already available in these television sets for the scan rate conversion. Using this available memory, a recursive (motion adaptive) temporal filter can be added for very little additional cost.

![Fig.1 Motion adaptive first-order temporal recursive noise filter.](image)

For standard television sets the field memory is not available, and therefore the noise reduction feature cannot be introduced cost effectively. It has been shown in previous papers that a spatial filter with a reliable noise estimation circuit provides an alternative [9,10]. This alternative was shown to be rather effective on white Gaussian noise (we reported subjective gains around 3 dB). We found, however, that for the special category of low-frequency noise, e.g. clamp noise, a temporal filter performed significantly better. This paper reports on a breakthrough in combining the reduction of these classes of noise through a very cost-effective filter approach. Focusing on clamp noise, this could be realized by filtering
a characteristic of groups of pixels (in this case, their average value), rather than their individual values.

The paper is organized as follows: Section 2 summarizes the earlier published algorithms of the spatial filter and the noise estimator which are implemented again on this new IC. Section 3 contains the basics of the novel temporal filter algorithm, which is further elaborated in Section 4, while the automatic adaptation of the temporal filter is discussed in Section 5. Section 6 discusses the combined architecture of the two filters, the application, and some key data of the new IC. Section 7 discusses the result of an evaluation of this IC. And lastly, conclusions are drawn in Section 8.

2. THE SPATIAL NOISE FILTER

The spatial filter realised on this chip is based on a variant of the sigma nearest neighbours selection process [7], which in our experiments showed a good subjective performance. The combination process involves weighted averaging. For every pixel position \( x \), image number \( n \), and input luminance signal \( F(x, n) \), the filter output \( F_F(x, n) \) can be written as:

\[
F_F(x, n) = G(x, n) \left( \sum_{k \in N_1} W_1(x, k, n) F(x + k, n) + \sum_{k \in N_2} W_2(x, k, n) F_F(x + k, n) \right)
\]

(1)

where \( N_1 \) and \( N_2 \) are sets of vectors defining a neighbourhood. Furthermore, \( W_1(x, k, n) \) and \( W_2(x, k, n) \) are weights, related to the absolute difference between the weighted pixel and the current input pixel:

\[
W_1(x, k, n) = f \left( |F(x, n) - F(x + k, n)| \right)
\]

(2)

\[
W_2(x, k, n) = f \left( |F_F(x, n) - F_F(x + k, n)| \right)
\]

(3)

where \( f(a) \) is a gain or normalization factor defined as:

\[
\frac{1}{G(x, n)} = \sum_{k \in N_1} W_1(x, k, n) + \sum_{k \in N_2} W_2(x, k, n)
\]

(4)

Using only one line memory, the following neighbourhoods have been selected:

\[
N_1 = \{ (2, 0), (0, 0), (-2, 0) \}
\]

(5)

and:

\[
N_2 = \{ (-8 + w, -4 + w), (0 + w, 4 + w), (8 + w) \}
\]

(6)

where \( w \) varies cyclically on line basis following the sequence 1, 0, -1, 0. The background of this unusual aperture is explained in [9]. Figure 2 shows a block diagram of the spatial noise filter as implemented on the IC.

The noise is estimated from the minimum absolute difference between two groups of neighbouring pixels on different positions in the same field. In other words, the correlation in flat areas is used to estimate the noise variance. The reliability of this measure is increased by searching for the interval, closest to zero, in which a pre-defined number of estimates \( NE \) fall. A circuit to find this interval can be realized with simple hardware as shown in Figure 3. In a feedback loop the upper and lower boundaries of a measurement interval are adjusted until the interval contains the pre-defined number of absolute differences. More details are available in [9].
3. BASIC CONCEPT FOR CLAMP NOISE FILTERING

Clamp noise reduction of image data can effectively be realized using recursive temporal filters. In the classical case [1,2], the filter output $F_F(x, n)$ is defined as:

$$F_F(x, n) = kF_F(x, n-1) + (1-k)F(x, n-1)$$  \hspace{1cm} (7)

where $k$ is a control parameter, defining the filter characteristics.

As can be seen from eq. (7), for each pixel in the field, the filtered luminance value from the previous field has to be stored in a field memory. However, field memories are expensive. Therefore, instead of temporally filtering individual pixels on a line, we propose temporal filtering of the average value of all (or at least a large portion of the) pixels in that line. The block diagram of this basic idea is given in Figure 4.

The average luminance value $F_A(y, n)$ for a line at position $y$ with $N$ pixels, is defined as:

$$F_A(y, n) = \frac{1}{N} \sum_{x=1}^{N} F(x, n)$$  \hspace{1cm} (8)

we now calculate a filtered average luminance value $F_{AF}(y, n)$ as:

$$F_{AF}(y, n) = kF_A(y, n) + (1-k)F_A(y, n-1)$$  \hspace{1cm} (9)

and we propose an output of the temporal filter $F_F(x, n)$ for every pixel at position $x$ with an input luminance value $F(x, n)$:

$$F_F(x, n) = F(x, n) - F_A(y, n) + F_{AF}(y, n)$$  \hspace{1cm} (10)

Without interlace and assuming stationary, i.e. not moving, pictures the DC-level of a line should be equal to that of the corresponding line in the previous picture. Strong (recursive) filtering of the average value of the pixels (DC-level) therefore effectively eliminates clamping errors.

For pictures of a CCIR601 format, the memory required in this temporal filter is reduced to 1/720 of a field memory.

4. ELABORATION OF THE BASIC IDEA

In principle, with interlace a frame delay (of one value per line) is required, but much more important is that motion – and particularly vertical motion – can drastically change the average pixel value of a line. A change detector (analogous to a motion detector in temporal filtering on pixel basis) can be applied to adapt the filter. The value of the filter coefficient $k$ becomes a monotonously decreasing function of the absolute difference $D(y, n)$ defined as:

$$D(y, n) = |F_A(y, n) - F_{AF}(y, n-1)|$$  \hspace{1cm} (11)

Experiments show that this does not provide an adequate solution, since in order to prevent artifacts in moving sequences the change detector has to be so sensitive that clamp errors and other low-frequency noises are detected as motion too. A similar situation occurs with motion adaptive temporal noise filters. The problem there is reduced by applying 2-dimensional filtering in the motion detector. Since the difference signal is only two-dimensional in the currently proposed temporal filter, the analogy of the two-dimensional spatial filtering is a 1-dimensional (vertical) filtering of the difference signal $D(y, n)$ prior to determining $k$. However, we found that in the proposed temporal filter an alternative and more effective solution is possible if each line is divided into segments. The background of this segmentation is the unlikelihood of all segments being influenced by motion simultaneously.
For each segment \( S_i(y) \), consisting of \( N_i \) pixels from the line at position \( y \), an average luminance value \( F_A(S_i(y), n) \) is defined according to (this replaces eq. (8)):

\[
F_A(S_i(y), n) = \frac{1}{N_i} \sum_{x \in S_i(y)} F(x, n)
\]

(12)

where \( i \in \{1, \ldots, S\} \) and \( S \) is the number of segments (\( S \) is assumed to be odd), so:

\[
N = \sum_{i=1}^{S} N_i
\]

(13)

In order to keep \( N_i \) constant for all segments it is possible that \( N \) (eq. (13)) is not equal to the number of pixels per line (\( N \) in eq. (8)).

Furthermore, for each segment the average luminance \( F_A(S_i(y), n) \) is independently filtered and \( F_{AF}(S_i(y), n) \) is obtained using the filtered information from the corresponding segment from the previous field (this replaces eq. (9)):

\[
F_{AF}(S_i(y), n) = k_i F_A(S_i(y), n) + (1 - k_i) F_{AF}(S_i(y), n - 1)
\]

(14)

where, at this stage \( F_{AF} = F_A \) (we return to this in Section 5), \( k_i \) is the filter coefficient for segment \( i \), according to our optimization calculated as:

\[
k_i = 1 - \frac{15}{16 + \frac{1}{4} D(S_i(y), n)}
\]

(15)

with

\[
D(S_i(y), n) = \left| F_A(S_i(y), n) - F_{AF}(S_i(y), n - 1) \right|
\]

(16)

A further improvement is realized by including transversal vertical filtering. To this end, an averaging filter is applied, using the filtered average luminance value \( F'_{AF}(S_i(y), n) \) for a given segment \( S_i \) from a line \( y \) and the corresponding filtered average luminance value \( F'_{AF}(S_i(y - 1), n) \) for the segment \( S_i \) from the previous line.

\[
F'_{AF}(S_i(y), n) = m_i F_A(S_i(y), n) + (1 - m_i) F_{AF}(S_i(y - 1), n)
\]

(17)

where:

\[
m_i = \begin{cases} 
0.5, & F_{AF}(S_i(y), n) \leq F_{AF}(S_i(y - 1), n) \\
1, & \text{else} 
\end{cases}
\]

(18)

and \( I(n) \) is an adaptive threshold, which will be defined in Section 5.

The next problem to be solved is how to find the segment(s) in which no DC-change has occurred due to motion. We propose here to use two differential order statistic filters (DOSF), based on the same ordering. The basis for the ordering is the difference between the current segment average and the corresponding filtered value in the previous field. The assumption is that segments with a low temporal correlation are affected by motion or vertical detail. An OS-filter will thus reject extreme segment values in contrast with a simple average, which equals the basic principle described in Section 3.

The difference \( DIF(S_i(y), n) \) for each segment \( S_i(y) \) is calculated from:

\[
DIF(S_i(y), n) = F_A(S_i(y), n) - F_{AF}(S_i(y), n - 1)
\]

(19)

Let \( F_A(y,n) = (F_{A1}(y,n), F_{A2}(y,n), \ldots, F_{AS}(y,n))^T \) be a column vector, comprising the ordered average luminance values of the different segments, so that the corresponding differences are in ascending order. Similarly ordered, the filtered average luminance values can be represented with the column vector \( F'_{AF}(y,n) = (F'_{AF1}(y,n), F'_{AF2}(y,n), \ldots, F'_{AFS}(y,n))^T \). Then the outputs of the DOS filters can be the average value and the filtered average value, respectively, of the segment(s) with the least extreme difference. These values will be called respectively the reference average and the filtered reference average and denoted by \( F_{AR}(y,n) \) and \( F_{ARF}(y,n) \). Thus:

\[
F_{AR}(y,n) = \bar{F}_A(y,n) \times C
\]

(20)

and

\[
F_{ARF}(y,n) = \bar{F}_{AF}(y,n) \times C
\]

(21)

where \( C \) is a row matrix of coefficients \( C_j \) for which holds:

\[
C_j = \begin{cases} 
1/2, & j = (S + 1)/2 \\
1/4, & j = ((S + 1)/2) \pm 1 \\
0, & \text{else} 
\end{cases}
\]

(22)

The efficacy of this approach, i.e. the robustness for motion in comparison with a simple line-average, is illustrated in Figure 5.
The reference average \( F_{AR}(y, n) \) and the filtered reference average are used for the correction of the input signal. The output \( F_F(x_0, n) \) of the clamp noise reducing circuit for a pixel at position \( x_0 \) with an input luminance value \( F(x_0, n) \) is given by (this replaces eq. (10)):

\[
F_F(x_0, n) = F(x_0, n) - F_{AR}(y, n) + F_{ARF}(y, n)
\]  

Figure 6 illustrates the algorithm described so far with a block diagram. For clarity in this block diagram, no effort is put into realizing an efficient implementation.

Clipping at black and top-white is required to prevent under and overflows, and this is implemented on the IC. In the IC each line of a field is divided into seven segments. The memory necessary for storing the filtered average luminance value \( F_{AR}(S_i(y), n - 1) \) of each segment and every line in this case equals \( 288 \times 7 \) (for 625-line CCIR 601).

### 5. Automatic Adaptation to the Clamp Noise Level

It was found experimentally that artifacts can still be introduced in (almost) clean pictures. This problem was solved by adapting the filters to the clamp noise level in the image sequence. The adaptation takes the form of a feedback loop.

Ideally, it should be expected that in sequences with heavy clamp noise the effect of the filter on the DC-level is stronger than in sequences with little clamp noise. With such an ideal filter and a uniform distribution of the clamp noise it can even be expected that the average DC-correction, applied to the pixels on a line, is close to half the peak level of the clamp noise. Therefore the measured average correction can be used to adapt the filter such that in the case of differences larger than the expected peak level of the clamp noise the coefficient “\( k \)” of the recursive filters reaches unity. Consequently, the filtering becomes weaker with little noise and stronger in the case of heavy noise.
To this end, an average correction \( AC(n) \) for each field is calculated according to:

\[
AC(n) = \frac{1}{N_{LPF}} \sum_{y} LC(y,n)
\]  

(24)

where \( N_{LPF} \) is the number of lines per field and \( LC(y,n) \) is the absolute value of the DC-correction of the line at position \( y \):

\[
LC(y,n) = \left| F_{ARF}(y,n) - F_{AR}(y,n) \right|
\]  

(25)

To implement the adaptation in the IC, the calculated average correction \( AC(n) \) and the peak correction value in the current field \( I(n) \) are used to define the peak correction value \( I(n+1) \), which is used in the next field to limit the effect of filtering:

\[
I(n+1) = \begin{cases} 
I(n) + 0.25 \cdot (I(n) < p_1 \cdot AC(n)) \land (CntD(n) \leq p_2) \\
I(n) - 0.25 \cdot (I(n) \geq p_1 \cdot AC(n)) \land (I(n) > 0.25) \land (CntD(n) \leq p_2) \\
0.25 \cdot (I(n) \geq p_1 \cdot AC(n)) \land (I(n) = 0.25) \land (CntD(n) > p_2)
\end{cases}
\]  

(26)

where the parameters in the IC are programmable.

Good results are obtained choosing: \( p_1 = 2 \), \( p_2 = 5 \) and:

\[
CntD(n) = \sum_{y=1}^{N_{LPF}} D(y,n)
\]  

(27)

Let \( DIF(y,n) = (DIF_1(y,n), DIF_2(y,n), ..., DIF_{S}(y,n))^T \) be a column vector, comprising the ordered segment differences \( DIF(S_i(y,n)) \) (eq. (20)) in an ascending order. \( D(y,n) \) is now defined as:

\[
D(y,n) = \begin{cases} 
1, & \left| DIF((S+1)/2-1)(y,n) \right| > p_3 \\
\lor \left| DIF((S+1)/2+1)(y,n) \right| > p_3 \\
0, & \text{else}
\end{cases}
\]  

(28)

where \( p_3 = 85 \) is chosen.

The filtered average luminance value \( F_{AF}(S_i(y), n) \) for a given segment \( S_i \) with an average luminance value \( F_A(S_i(y), n) \) is modified according to:

\[
F_{AF}(S_i(y), n) = F_{A}(S_i(y), n) + \lim \left[ F_{AF}(S_i(y), n) - F_{A}(I(y), n) \cdot I(n) \right]
\]  

(29)

with:

\[
\lim [a;b] = \begin{cases} 
  a, & (|a| \leq b) \\
b, & (a > b) \\
-b, & (a > b)
\end{cases}
\]  

(30)

The flow chart of the algorithm, including the above described noise adaptation, is shown in Figure 7.

![Flow chart of the algorithm](Fig.7 Flow chart of the eventual algorithm.)
6. ARCHITECTURE AND APPLICATION ENVIRONMENT

As mentioned in the introduction, the new IC design combines the designed spatial noise filter and noise estimation circuit, described in earlier publications [9,10], with the new LF-noise temporal filter described extensively in this paper. A straightforward combination of the two filters is shown in Figure 8. Since the segment averages can be completely computed only at the end of the video line, the calculation of the clamp error correction signal introduces a pipeline delay of one video line. Therefore a compensation delay is necessary, as shown in the figure. Making use of the delay required in the sigma filter, a more efficient architecture can be realised, as shown in Figure 9. In this architecture the input to the clamp error calculator is the output of the sigma filter, while the correction value is added to the (line) delayed output of the sigma filter, thus compensating for the computation delay. A more detailed diagram of this architecture is shown in Figure 10.

The IC has analog data interfaces and pre-filtering for Y, U and V, while the noise processing functions are implemented digitally. It has a PLL to extract a clock from a sandcastle signal containing vertical and horizontal synchronisation information. The data path of the white noise reduction operates on 8-bit video input, but calculates to a precision of 10-bits which is later the input to the clamp noise estimator. The digital to analog conversion at the backend is based on 10-bit converters.

Fig. 8 Straightforward cascade of the two filters.

Fig. 9 Combined architecture of the two filters, eliminating the compensating delay.

Fig. 10 Internal details of Figure 9 above.
An I²C-bus supports programmable aspects of both analog and digital functions. Because of the relatively low speed (400 kHz) of this interface, it is allowed to update parameter registers during active video, but the effect of the change is buffered until the field blanking interval.

Both the white noise and clamp noise estimations are based only on the luminance signal. Clamp noise correction is only applied to the luminance path whereas all channels can be corrected for white noise. A PAL averaging function and a comb filtering for NTSC can be enabled in the color-difference channel. The chip has a split-screen (left-right) mode allowing comparison of processed with unprocessed data. The strength of the white noise filtering has 16 degrees of user control in addition to the auto-adaptive control arising from the global statistics of the image itself.

Further details of the device are shown in Table 1, while Figure 11 shows the chip photo.

### Table 1 Key IC data

<table>
<thead>
<tr>
<th>Process</th>
<th>CMOS 0.8 μm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Die size</td>
<td>53 mm²</td>
</tr>
<tr>
<td>Number of transistors</td>
<td>300,000</td>
</tr>
<tr>
<td>Memory</td>
<td>36.5 kbit</td>
</tr>
<tr>
<td>Dissipation</td>
<td>0.8 W (5 V, 13.5 MHz)</td>
</tr>
<tr>
<td>Package</td>
<td>SDIP42</td>
</tr>
<tr>
<td>I/O</td>
<td>Analog</td>
</tr>
<tr>
<td>Data clock</td>
<td>13.5 MHz</td>
</tr>
</tbody>
</table>

7. EVALUATION

Figure 12 shows the setup of the evaluation of this filter. The evaluation procedure was simple: Clamp error of known amplitude was added to the input signal $F_o(x, n)$. This was done by adding a random number to the pixels in each line of the original image, the number being limited in amplitude to realise the desired input noise level. This offset was kept constant from the beginning to the end of each line, such that the variation was only in the vertical direction. The (clamp) noisy signal, $F_n(x, n)$, was then passed through the temporal filter, with output $F_F(x, n)$, and the improvement measured against the input clamp error level. The improvement in the SNR was calculated as:

$$Gain_{SNR} = 10 \log \frac{\sum_{x=1}^{N,M} (F_o(x,n) - F_n(x,n))^2}{\sum_{x=1}^{N,M} (F_o(x,n) - F_F(x,n))^2} \, \text{dB} \quad (31)$$

for a video field of $N\times M$ pixels.

Figures 13-16 show individual images from each of the different test sequences that we used. These sequences contain horizontal motion (Fig.13), strong vertical motion (Fig.14), stationary images (Fig.15) and camera zooming (Fig.16). The evaluation result, in Figure 17, shows that on average we could expect an improvement of at least 3 dB in the SNR for input SNR levels between 30 and 45 dB. For relatively noise-free inputs, the improvement is small, as expected. Also, the automatic adaptation of the filtering to the input noise level ensures that artifacts are hardly introduced, as can be seen in the plots. For stationary images, the correlation in the temporal direction is high. A significant improvement in SNR should thus be expected.

As shown in Figure 17, a peak improvement of up to 8 dB in signal-to-noise ratio was recorded. Even for relatively low input noise levels, a gain of about 4 dB was realised. However, the improvement figure falls significantly when there is movement, as the temporal correlation drops with motion. Also, because the algorithm employs vertical filtering, the performance suffers somewhat when there is high vertical detail. This is the case for the remaining 3 plots.
Fig. 12  Setup of the filter performance evaluation.

Fig. 13  “Renata”

Fig. 14  “Interview”
8. CONCLUSION

A unique automatically adapting spatio-temporal noise filter for television application was designed and realized in silicon. The algorithm of the spatial noise filter and its noise measurement circuit were summarized, referring to an earlier publication. The innovative algorithm for temporal filtering with limited memory capacity demands on the current IC was described in detail. This clamp-noise filter calculates averages of groups of pixels for every line of pixels, stores these average values for every line of pixels in a memory, and contains means for filtering these averages. The original DC level of all pixels in a line is then replaced by the filtered ones in such a way that the clamp noise is reduced without introducing artefacts in clean signals. This method of degradation prevention for clean or difficult sequences was also described. Its main characteristic is that the temporal filter is adapted to the average modification of the DC-level in a previous picture or field of the image signal. This adaptation is based on the assumed fixed relation between the average and maximum value of noise in a single image.

The spatial noise filter on board of the presented IC yields a subjective gain on Gaussian noise of approximately 3 dB. The temporal noise filter in the IC realises some 3 to 8 dB gain in LF-noise, in particular on clamp noise. The higher gains are realized on noisy stationary images. The evaluation showed that the means to prevent degradation in the case of clean images or strong motion, successfully guaranteed positive improvement figures in all cases.
9. REFERENCES


BIORGAPHY

Gerard de Haan was born in Leeuwarden, The Netherlands, on April 4, 1956. He received the B.S., the M.S. and the D.S. degree from Delft University of Technology in 1977, 1979 and 1992 respectively. In 1979 he joined the Philips Research Laboratories in Eindhoven, where he led several research projects in the area of image processing, and participated in a number of European projects. He has coached students from various universities, and since 1988 has taught at the Philips Centre for Technological Training. In 1991/1992, he was a visiting researcher in the Information Theory Group of Delft University. At present, he is a senior scientist in the Television Systems Group of Philips Research and has a particular interest in algorithms for motion estimation, scan rate conversion, and image enhancement. His work in these areas has resulted in some 35 patents and patent applications, and in the first place of the 1995 ICCE Outstanding Paper Awards. He was responsible for the algorithmic design of the 100 Hz TV with “Natural Motion”, for which Philips received the European Innovation Award of the Year 95/96 from the European Imaging & Sound Association.

Tanya Kwaaitaal-Spassova was born in Kjustendil, Bulgaria on July 15, 1954. She received her degree in electronics at the Technical University Gabrovo, Gabrovo, Bulgaria in 1978. After her graduation she worked as a research assistant and later as a lecturer at the Automation and Computer Science Department of the Technical University Gabrovo Bulgaria. In 1993 she joined the Philips Research Laboratories, Television Systems Group. Presently she is working as a system designer in the Video-Memory Application Group, Philips Sound & Vision, BG-TV, TV-lab. Her current research interests are in the area of digital signal processing for television picture enhancement.
Margaret Larray was born in Dublin, Ireland on 15 April 1963. She studied at Trinity College, Dublin and received the B.A.I and M.Sc. degree in 1986 and 1988 respectively. Since then she has been working at Philips Research Laboratories, Eindhoven in an Integrated Circuit Design group on CMOS digital and mixed-signal chips for a wide range of application areas. Her interests include digital signal processing and design and verification methodologies.

Olukayode A. Ojo was born in Ijaka-Oke, Nigeria, on 15 August, 1965. He completed his Bachelor’s degree in Electrical Engineering at the University of Ibadan, Nigeria, in July, 1987. In May, 1990, he obtained his Master’s degree in Electronic Engineering from the Netherlands University Foundation For International Cooperation (NUFFIC). He joined the Philips Research Laboratories in Eindhoven in November 1990. Since then he has been active, as a research scientist, in the Television Systems Group. He is involved in the area of digital signal processing for video applications. His work, in collaboration with others, has led to a number of publications, VLSI realisations and patent applications. Notable among these is the motion-compensated film portrayal improvement feature in 100 Hz television sets (“Natural Motion”), to which he has made a significant contribution. His field of interest covers subjects like noise reduction, motion estimation and its applications, and so on. He is currently active in the specification and design of real-time architectures for video algorithms.

Robert Jan Schutten was born in Groningen, The Netherlands, on January 2, 1972. He received his Masters degree in Electrical Engineering from Delft University of technology in 1994. In 1995 he joined the Philips Research Laboratories in Eindhoven, where he works as a research scientist in the Television Systems Group. His areas of interest are in algorithms for image enhancement and scan rate conversion.