An Efficient True-Motion Estimator Using Candidate Vectors from a Parametric Motion Model

Gerard de Haan, Senior Member, IEEE, and Paul W. A. C. Biezen

Abstract—Some efficient motion estimation algorithms select their output motion vector from a limited number of likely correct candidate, or prediction, vectors. In this paper, next to the known spatial and temporal prediction vectors, an additional and independent prediction is proposed. This candidate is generated with a parametric model describing the global motion in a previously estimated motion vector field. The proposal is elaborated as an addition to the three-dimensional (3-D) recursive search block-matching algorithm. The evaluation shows that a subpixel accurate, true-motion estimator results with a very low operations count.

Index Terms—Efficient motion estimation, motion compensation, parametric motion model, scan rate conversion, true-motion estimation.

I. INTRODUCTION

MOTION estimators minimize or maximize a sometimes complex, but often simple, criterion function, selecting one of various possible motion vectors for every location in the image. They either apply the exhaustive search method, i.e., try all possible vectors in a predefined range, to obtain the global optimum of the criterion function, or use one of the efficient approaches and test only a limited number of candidate vectors [9]–[11], [13]. In the more advanced efficient motion estimation methods this limited candidate set contains the most likely prediction vectors. The likelihood can be based on analysis of the picture at a higher hierarchical level [2], [14], by analysis in the frequency domain [15]–[17], or on spatial and/or temporal proximity of the vectors in the candidate set [4]–[6], [12], [20].

Motion in video images is either due to object motion or caused by camera movements. Object size causes correlation of motion vectors in the spatial domain, while object inertia explains temporal correlation of vector fields. The category of camera motion includes motion due to pans, tilts, and travels of the camera, and zooming with its lens. This type of motion usually causes very smooth vector fields in the spatial and in the temporal domain. A zoom with the camera lens results in motion vectors that are linearly changing with the spatial position. A pan, tilt, or travel with the camera, on the other hand, causes a uniform motion vector value for the entire television screen. These types of motion can be described with a three parameter model, as has been suggested before [3]. Also, methods have been suggested in the literature how to extract the parameters of such a model from a rough motion vector field [1]. Even more complex global motion, like rotation, can be described with a parametric model, which then needs up to eight parameters [18]. In this paper we propose to use the rough estimate of the motion vector field in the previous image for generating, with a simple parametric description of this field, an additional prediction vector for every location in the picture. This prediction is included in the candidate set of an advanced efficient block-matcher that further includes temporal and spatial prediction vectors and an update thereof [4], [6]. As the parametric model enables calculation of vectors even at the boundaries of the image and, if required, with subpixel accuracy, it can bring a considerable improvement in case of camera motion, particularly if there are areas with little or low-contrast detail near the boundaries of the image, where temporal and spatial predictors sometimes fail.

Section II of this paper introduces our notation and summarizes the motion estimation algorithm of [6], which we used as a basis to validate our improvement. Section III briefly introduces the parametric model and the extraction of the additional candidates from previously calculated motion vector fields. Section IV describes the synthesis of the new motion estimator. Section V gives an evaluation of the improvement in a critical application (deinterlacing), and Section VI summarizes our conclusions.

II. THE 3-D RECURSIVE SEARCH BLOCK MATCHER

As we considered perfection of an advanced motion estimator most interesting, we applied the high quality method of [6]. This algorithm yields a quarter pel accuracy and a close to true-motion vector field, relevant for scan rate conversion [4], [5]. Furthermore, the efficiency of this algorithm is such that it is currently the only single chip true-motion estimator [7], [8].

In block-matching motion estimation algorithms, a displacement vector is assigned to the center \(X = (X_x, X_y)^T\) of a block of pixels \(B(X)\) in the current field \(n\) by searching a similar block within a search area \(SA(X)\), also centered at \(X\), but in the previous field \(n-1\). The similar block has a center, which is shifted with respect to \(X\) over the displacement vector \(D(X, n)\). To find \(D(X, n)\), a number of candidate vectors \(\mathcal{C}\) are evaluated applying an error measure \(e(C, X, n)\) to quantify block similarity. Fig. 1 illustrates the procedure.

More formally, \(CS_{\text{max}}\) is defined as the set of candidates \(\mathcal{C}\), describing all possible (usually integer) displacements with respect to \(X\) within the search area \(SA(X)\) in the previous image

\[
CS_{\text{max}} = \{ C | N \leq C_x \leq N + M, -M \leq C_y \leq +M \} \tag{1}
\]

where \(N\) and \(M\) are constants limiting \(SA(X)\). Furthermore,
a block $B(X)$ centered at $X$ and of size $X$ by $Y$ consisting of pixel positions $\mathbf{z} = (x, y)^T$ in the present field $n$, is now considered

$$B(X) = \{\mathbf{z} | X_x - X/2 \leq x \leq X_x + X/2 \land X_y - Y/2 \leq y \leq X_y + Y/2\}.$$  

(2)

The displacement vector $D(X, n)$ resulting from the block-matching process, is a candidate vector $C$ which yields the minimum value of an error function $e(C, X, n)$

$$D(X, n) \in \{C \in C_{\text{candidate}} | e(C, X, n) \leq e(E, X, n) \forall E \in C_{\text{candidate}}\},$$

(3)

If, which is the common case, the vector $D(\mathbf{z}, n)$ with the smallest matching error is assigned to all pixel positions $\mathbf{z}$ in the block $B(X)$:

$$\forall \mathbf{z} \in B(X): \quad D(\mathbf{z}, n) \in \{C \in C_{\text{candidate}} | e(C, X, n) \leq e(E, X, n) \forall E \in C_{\text{candidate}}\},$$

(4)

rather than to the center pixel only, a large reduction of computations is achieved. As an implication, consecutive blocks $B(X)$ are not overlapping.

The error value for a given candidate vector $C$ is a function of the luminance values of the pixels in the current block and those of the shifted block from a previous field, summed over the block $B(X)$. A common choice, which we will use, is the sum of the absolute differences (SAD)

$$e(C, X, n) = \text{SAD} = \sum_{\mathbf{z} \in B(X)} |F(\mathbf{z}, n) - F(\mathbf{z} - C, n)|,$$

(5)

although a mean square error (MSE) or a normalized cross correlation function (NCCF) are sometimes proposed.

Rather than calculating all possible candidate vectors, the three-dimensional (3-D) recursive search block-matcher of [6] takes spatial and/or temporal “prediction vectors” from a 3-D neighborhood and a single updated prediction vector. This implicitly assumes spatial and/or temporal consistency. We applied a candidate set $CS(X, n)$ containing candidate vectors $C$ from which the block matcher selects its result vector $D$, defined by

$$CS(X, n) = \left\{ C \in C_{\text{candidate}} | e(C, X, n) \right\}.$$

(6)

where the two options for candidate vector $C_1$ alternate on block basis, and the update vector $U(X, n)$ is taken from a limited fixed integer update vector set $US$, in our case

$$US_1 = \left\{ \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \begin{bmatrix} 0 \\ 2 \end{bmatrix}, \begin{bmatrix} 0 \\ -2 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} -1 \\ 0 \end{bmatrix}, \begin{bmatrix} 3 \\ 0 \end{bmatrix}, \begin{bmatrix} -3 \\ 0 \end{bmatrix} \right\}.$$

(7)

To realize subpixel accuracy, the update set of (7) is extended with fractional update values. We realized a quarter pel resolution by adding the following fractional vectors to the update set:

$$US_f = \left\{ \begin{bmatrix} 0 \\ 0.25 \end{bmatrix}, \begin{bmatrix} 0 \\ -0.25 \end{bmatrix}, \begin{bmatrix} 0.25 \\ 0 \end{bmatrix}, \begin{bmatrix} -0.25 \\ 0 \end{bmatrix} \right\}.$$

(8)

Because of the small number of candidate vectors, the method is very efficient. Furthermore, due to the inherent smoothness constraint, it yields very coherent vector fields that closely correspond to the true motion of objects. This particularly makes it suitable for scanning format conversion. Fig. 2 illustrates the candidate set by showing the locations, relative to the current block, of the spatial and spatio-temporal prediction vectors.

III. CAMERA MOTION AND THE PARAMETRIC MODEL

Camera motion includes motion due to panning, tilting, traveling, and zooming of the camera. This type of motion
has a very regular character causing very smooth velocities, i.e., motion vectors, compared to object motion. Zooming with the camera will generate motion vectors that linearly change with the spatial position. Panning, tilting, or traveling with a camera, on the other hand, will generate a uniform motion vector field for the entire television screen.

A. A Three-Parameter Model

These types of motion can be described with a three-parameter model, as has been suggested before [3], using

\[ D(x, n) = \begin{bmatrix} p_1(n) + p_3(n)x \\ p_2(n) + p_4(n)y \end{bmatrix} \]

(9)
in which \( p_1 \) describes the panning, \( p_2 \) the tilting, and \( p_3 \) the zooming of the camera. This model is valid only if the blocks are exactly square on the image plane. At least in television this assumption is often invalid, even when the blocks are square on the pixel grid, due to the use of various aspect ratios with fixed sampling frequency.

In this case, a four-parameter model can be used instead

\[ D(x, n) = \begin{bmatrix} p_1(n) + p_3(n)x \\ p_2(n) + p_4(n)y \end{bmatrix} \]

(10)
where a fixed ratio results between \( p_3 \) and \( p_4 \), which is determined by the ratio of the horizontal (H) and vertical (V) sampling density. If the ratio \( p_3/p_4 \) does not correspond to the ratio of the \( H \) and \( V \) sampling density, the extracted parameters are unreliable. It is possible therefore, by measuring this ratio, to identify situations in which it is better not to use the additional candidate in the motion estimator.

Extending the model to a six-parameter model makes it possible to include rotations. This type of camera motion, however, is not very likely in television broadcast material, and therefore leads to a mostly useless complication, not further considered in this paper.

B. Upgrading the 3-D RS Block-Matcher with a Parametric Candidate

An implementation of the proposal results by adding a single candidate vector for every block to the set \( CS(X, n) \) defined in (6), where this additional candidate \( C_4(X, n) \) for the block \( B(X) \) is calculated according to the four-parameter model of (10), i.e.,

\[ C_4(X, n) = \begin{bmatrix} p_1(n) + p_3(n)x \\ p_2(n) + p_4(n)y \end{bmatrix} \]

(11)

Rather than adding a candidate vector to the motion estimator, it is possible to alternate (e.g., on block basis) this additional candidate with another candidate vector. In this case, the operations count of the estimator hardly increases, while the advantage was found to be comparable. Consequently, candidate vector \( C_4 \) in (6) is modified to

\[ C_4(X, n) = \begin{cases} \begin{bmatrix} p_1(n) + p_3(n)x \\ p_2(n) + p_4(n)y \end{bmatrix}, & \text{odd blocks} \\ D \left[ X + \left( \begin{array}{c} 0 \\ 2Y \end{array} \right), n - 1 \right], & \text{even blocks} \end{cases} \]

(12)

A refinement of this thought is to have the occurrence frequency of the parametric model candidate depend on the reliability, mentioned in Section III-A, of the model. In our implementation this led to a further modification of \( C_4 \) according to

\[ C_4(X, n) = \begin{cases} \begin{bmatrix} p_1(n) + p_3(n)x \\ p_2(n) + p_4(n)y \end{bmatrix}, & \text{model reliable} \\ D \left[ X + \left( \begin{array}{c} 0 \\ 2Y \end{array} \right), n - 1 \right], & \text{else} \end{cases} \]

(13)

C. Extraction of the Parameters from the Image Data

There are many options to extract the parameters of a global motion model from an estimated motion vector field. In our case, where the model is integrated in the 3-D RS block matcher, it makes sense to start from already available motion vectors, i.e., the vectors available in the temporal prediction memory. To keep the operations count low, it is furthermore attractive to use only a limited set of the vectors available in this memory.

As a consequence of the choice to use vectors from the prediction memory, the validity of the candidates generated with the parametric model increases in time due to the recursiveness of this approach. If the parametric model is valid for significant portions of the image, the additional candidate will be selected by the estimator engine more often, which in the next field improves the quality of the parameters extracted from the vectors in the temporal prediction memory. This again improves the accuracy of the generated additional candidate, it will be selected more often, and the model will improve further, etc.

To estimate the parameters of the model describing the global motion, we took a sample set \( S(n) \) containing nine motion vectors, \( D(X, n - 1) \) from different positions \( X \) on the block grid in a centered window of size \( (W - 2m)X \) by \( (H - 2q)Y \) in the picture with width \( W \) and height \( H \) from the temporal vector prediction memory according to

\[ S(n) = \{ D(X, n - 1)|X = -(\frac{1}{2} W - m)X, 0; \]
\[ + \left( \frac{1}{2} W - m \right)X, X_y = -(\frac{1}{2} H - q)Y; 0; \]
\[ + \left( \frac{1}{2} H - q \right)Y \}

(14)
where the value of \( m \) and \( q \) is not critical. We selected \( m = 0.1W \) and \( q = 0.1H \). Fig. 3 illustrates the position in the image from which the vectors in the sample set are taken. Choosing \( m \) and \( q \) to be small generally yields a more accurate estimate of the zoom parameters \( p_3 \) and \( p_4 \). Vectors too close to the image boundary, however, can be unreliable in the case of fast camera motion. Our choice seems a good and noncritical compromise. It is clear that other sample sets are also possible. They may even prove to be better in specific cases, but we see no evident cause for systematical improvement with another set of similar size. Larger sample sets can improve the result, but also increase the computational burden.

A four-parameter model can be solved with four equations, i.e., two “independent” sample vectors. Independent, here,
means that these vectors are not taken from the same horizontal or vertical position of the image. The number of combinations to choose \( r \) samples from a set of \( s \) is known from elementary statistics

\[
\binom{s}{r} = \frac{s!}{(s-r)!r!} \quad (r, s \in \mathbb{N}^+) \quad (15)
\]

In our case we arrive at \( n = 9 \) and \( s = 2 \), which implies \( 36 - 18 = 18 \) options to solve the model, as there are 36 combinations of 2 out of 9, and 18 pairs of “dependent” samples, i.e., samples on the same row or column (There are three rows and three columns, each enabling three pairs of two vectors being selected from the available three vectors, so \( 2 \times 3 \times 3 = 18 \) dependent pairs).

Extraction of the parameters from the \( i \)th pair of independent sample vectors \( \Delta(X_{i1}, Y_{i1}, n - 1) \) and \( \Delta(X_{i2}, Y_{i2}, n - 1) \), both in \( S(n) \), is straightforward

\[
p_j^2(n) = \frac{\Delta_b(X_{i1}, Y_{i1}, n - 1) - \Delta_b(X_{i2}, Y_{i2}, n - 1)}{X_{i1} - X_{i2}} \quad (16)
\]

and

\[
p_j^2(n) = \frac{\Delta_b(X_{i1}, Y_{i1}, n - 1) - \Delta_b(X_{i2}, Y_{i2}, n - 1)}{Y_{i1} - Y_{i2}} \quad (17)
\]

while

\[
p_j^2(n) = D_x(X_{i1}, Y_{i1}, n - 1) - p_j^2(n)X_{i1} \quad (18)
\]

and finally

\[
p_j^2(n) = D_y(X_{i1}, Y_{i1}, n - 1) - p_j^2(n)Y_{i1} \quad (19)
\]

Now for every pair of two independent sample vectors, a set of four parameters is available, and the best parameter set has to be selected. We propose here to assign the median value of all options for each parameter to the eventual model parameter, to eliminate the effect of outliers due to object motion. Consequently, for \( i = 1 \) up to \( i = 18 \) we rank the individual parameter values such that

\[
\forall[j | 1 \leq j \leq 4]: \quad P_j^{i-1}(n) \leq P_j^i(n) \leq P_j^{i+1}(n) \quad (20)
\]

and derive the value of the four-model parameters according to

\[
\forall[j | 1 \leq j \leq 4]: p_j(n) = \frac{1}{2} \left[p_j^{0}(n) + p_j^{4}(n)\right] \quad (21)
\]

It is good to notice that since \( p_1 \) and \( p_3 \) as well as \( p_2 \) and \( p_4 \) are dependent ([18], [19]), it would be correct to estimate them simultaneously rather than individually. We can, however, afford to neglect the dependencies, as temporary misestimates have a negligible negative effect on the estimator, as will become evident in the next section, whereas a temporary correct estimate improves the vector fields immediately and for many fields thereafter due to the temporal prediction available in the recursive estimator. In other words, the correct parameter set looses its importance for some time after it was once successful. For robust estimation of lines instead of scalars the median is not defined, and a more complicated method would have to be used, with arguable advantage.

Finally, the ratio of \( p_3 \) and \( p_4 \) is calculated to check the reliability of the model. In the experiments we allowed 25% deviation from the nominal value. This implies that if the ratio should be \( p_3/p_4 = R \) according to the \( H \) and \( V \) sampling densities (see Section III-A), the parameters are assumed reliable if \( 0.75R < p_3/p_4 < 1.25R \). If the parameters pass this test, it is possible to slightly improve the accuracy of the model by replacing \( p_3 \) and \( p_4 \) by their average value after correcting for the different sampling densities \([p_3 \text{ becomes } p_3 + R p_4]/2 \text{ and } p_4 \text{ becomes } p_4 + 1/R p_3]/2\). This sophistication was implemented in the evaluated algorithm.

IV. EVALUATION OF THE IMPROVEMENT

The expected advantage of our proposal was an increased accuracy of the estimated motion vectors in case of camera motion. As the original 3-D RS block-matching algorithm is already very accurate, a critical application is required to show the improvement. We found in an earlier stage already [6] that deinterlacing requires a high accuracy from the motion estimator, i.e., the subpixel fraction of the estimate plays an important role in the deinterlacing quality. Therefore, here again, we applied the 3-D RS blockmatcher in the time-recursive deinterlacing algorithm of Wang et al. [19] to verify the advantage of the parametric candidate vector. We essentially calculate a motion-compensated mean square prediction error, similar to what we used in [6], to get an indication of the quality of the motion vectors.

More precisely, the MSE is calculated in the measurement window (MW) of \((W-2m)X\) by \((H-2n)Y\) pixels, indicated in Fig. 3, between the luminance of the current input field \(I(x,n)\) and the previously calculated sequentially scanned picture \(I(x,n-1)\) shifted over the estimated displacement vector \(\Delta(x,n)\)

\[
\text{MSE}(n) = \frac{1}{(W-2m)X(Y-2n)Y} \sum_{x \in \text{MW}} \left[ I(x,n) - I(x-\Delta(x,n),n-1) \right]^2 \quad (22)
\]

where \(x\) runs through all positions in the measurement window on the pixel grid of the odd lines in an odd input field and
through all pixel positions of the even lines in case of an even input field. Fig. 4 illustrates this MSE calculation.

One picture of each of the test sequences used for our evaluation is shown in Figs. 5–7. In Fig. 8 we show the value of MSE$(n)$ for a number of fields of the sequence “Doll” with and without the parametric candidate vectors in the 3-D RS block-matcher of [6]. It can be clearly seen that the old and the new algorithms converge in the first few fields, and that the new estimator with parametric candidates gives a significant improvement of the performance in the later fields. Fig. 9 shows the same for another sequence. The sequences Doll and Car&Gate were selected to provide critical test material. Both sequences contain highly detailed areas and camera motion (zoom) as well as object motion (the doll and the vintage car, respectively).

To provide confidence that the additional candidate does not degrade the performance in case of no camera motion, we included a test sequence “Bicycle” (Fig. 7) with only rotating object motion and no detail in the background. As can be seen from the result shown in Fig. 10 there is no degradation due to the additional candidate. As should be expected, there is no advantage either, as the algorithm switches off the candidates from the parametric model as the sample vectors result in an unreliable model (see Section III-C). In case this reliability detector fails, no dramatic degradation results, as is shown in Fig. 11. The evaluation shown in this figure results if the candidate vectors generated with the parametric model are used in the estimator although the model is judged unreliable.

Concerning the complexity of the algorithm, we like to emphasize that practically all calculations are required once per field only. The exception is the generation of the candidate vector using the model, which requires a few operations per block of 8 by 8 pixels. Considering that a picture contains some hundred-thousands of pixels, it is clear that the operations count of the addition is negligible compared with the cost of the motion estimator (which requires a few operations per pixel).

Finally, Figs. 12 and 13 enable a subjective impression of the estimated subpixel accurate motion vectors as obtained in the seventh picture of the “Car&Gate” sequence. The pictures show the estimated horizontal vector component as
a grey value for every location in the picture. It is possible to recognize the slow zooming of the camera by the vertical "equivelocity bands" in the background portions of the image. The position of the vintage car is visible as a more or less rigidly moving object, and the closing of the gate left of the car is visible. The increased quality of the estimate of Fig. 13 compared to that in Fig. 12, particularly for camera motion (zooming of the background), is evident.

V. CONCLUSION

A new motion estimator has been presented. This estimator selects its motion vector for the current image from four candidate vectors only and is characterized in that one of these candidate vectors is produced with a parametric model describing the global (camera) motion in the previous image.

We introduced this "parametric candidate" in a very efficient (3-D recursive search) block-matching algorithm. To this end, we block-alternatingly replaced one of the candidates of this estimator by a candidate generated with a four-parameter model. The parameters for the model were extracted once per field using nine widely spaced sample vectors only, taken from the previously estimated vector field. This results in an almost negligible additional processing power requirement for the eventual algorithm. We showed that, with these nine extracted motion vectors, it is possible to generate 18 sets of four parameters describing the camera motion. A median
operation was introduced to select the applied parameter set, thus eliminating the outliers due to object motion. Furthermore, we showed that knowledge of the horizontal and vertical sampling densities could be used to judge the reliability of the model. This information provided a means to switch off the block-alternating injection of the “parametric candidate” in the estimator in case it could lead to a degradation of the estimator performance.

In the evaluation part of the paper a significant advantage, up to 50% reduction in MSE, was found on critical material applying the motion vectors for deinterlacing. A photograph of the vector field enabled a subjective evaluation, suggesting also that the fractional part of the displacement vectors obtained from the resulting block-matcher closely correspond to the true-motion in the sequence. Finally, it was shown that the performance of the reliability indicator, although working properly in the experiments, was not too critical.

REFERENCES


Gerard de Haan (M’95–SM’97) was born in Leeuwarden, The Netherlands, in April 1956. He received the B.Sc. (1977), the M.Sc. (1979, cum laude), and the D.Sc. (1992) degrees from Delft University, The Netherlands.

In 1979, he joined the Philips Research Laboratories in Eindhoven, The Netherlands. He conducted research in the area of image processing and participated in European projects. He has coached students from various universities and has been teaching since 1988 for the Philips Centre for Technical Training. In 1991/1992, he worked as a Visiting Scientist 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 about 35 patents and patent applications, and several scientific papers.

Dr. de Haan is the first place winner of the 1995 ICCE Outstanding Paper Awards. The Philips 100-Hz TV with “Natural Motion” based upon his work received the European Innovation Award of the Year 1995/1996 from the European Imaging & Sound Association. In 1994, he was a Guest Editor for Signal Processing: Image Communications.

Paul W. A. C. Biezen was born in Goirle, The Netherlands, on March 22, 1965. He completed his study in electrical engineering at the College of Advanced Technology Rijswijk, The Netherlands, in 1990.

In August 1990, he started working at the Philips Research Laboratories, Eindhoven, The Netherlands, as a Member of the Technical Staff in the Television Systems Group. He contributed to the Philips Video Signal Processor and designed hardware for real-time video signal processing. He has contributed to the algorithmic design of the 100-Hz TV with “Natural Motion,” for which Philips received the European Innovation Award of the Year 1995/1996 from the European Imaging & Sound Association. His current research interests are in the area of motion estimation, motion compensation, and television scan rate conversion. His work in this area resulted in a number of patents and patent applications.