Robust Global Motion Estimation Oriented to Video Object Segmentation

Bin Qi, Mohammed Ghazal, *Student Member, IEEE*, and Aishy Amer, *Senior Member, IEEE*

***Abstract*—Most global motion estimation (GME) methods are oriented to video coding while video object segmentation methods either assume no global motion (GM) or directly adopt a coding-oriented method to compensate for GM. This paper proposes a hierarchical differential GME method oriented to video object segmentation. A scheme which combines three-step search and motion parameters prediction is proposed for initial estimation to increase efficiency. A robust estimator that uses object information to reject outliers introduced by local motion is also proposed. For the first frame, when the object information is unavailable, a robust estimator is proposed which rejects outliers by examining their distribution in local neighborhoods of the error between the current and the motion-compensated previous frame. Subjective and objective results show that the proposed method is more robust, more oriented to video object segmentation, and faster than the referenced methods.**

***Index Terms*—Global motion estimation (GME), hierarchical differential estimation, residual information, robust estimator, video object segmentation.**

I. Introduction

SINCE motion is an important part of video signals, motion estimation is one of the most widely used methods in video processing. In general, motion is classified as *local motion* (LM) or *global motion* (GM). The term *local motion* refers to the apparent 2-D motion caused by object movement, whereas the term *global motion* is used in this paper to describe the apparent 2-D motion introduced by camera motion that is parameterized by a motion model. The process to estimate the parameters of the model is known as *global motion estimation* (GME). GME is usually followed by *global motion compensation* (GMC) (e.g., for predictive coding [1] or for object segmentation [2], [3]).

GME has many applications, such as sprite generation, video coding, scene construction, and video object segmentation. Depending on the application, the requirements on GME may differ. For example, in video coding, estimated motion does not need to resemble the *true* motion as long as the bit rate is achieved for a given quality (e.g., [4]). Even if GMC fails, *local motion compensation* (LMC) is used to maintain the coding quality. On the other hand, video object segmentation requires accurate GME to compensate for GM and retain the motion of objects (e.g., [2]). In this case, LMC is avoided to preserve the objects. Most GME methods are designed for video coding while most object segmentation methods either assume no camera motion or directly use a coding-oriented GME method. Some coding-oriented GME techniques sacrifice quality to gain speed, which may not be suitable for object segmentation. Since accuracy usually means extra computations, computational complexity is another major challenge in segmentation-oriented GME.

GME approaches are classified into three categories: phase correlation based [5], [6], background matching based [7]–[9], and hierarchical differential based [1], [10], [11]. Phase correlation methods first transform the frames to the frequency domain using the *Fourier* transform. Then, using the *Fourier* shift property, the translation between consecutive frames is identified. The advantages of phase correlation methods are fractional-pel accuracy and insensitivity to illumination changes [12]. However, the translational model they assume is not suitable for many video sequences. Phase correlation is used as a coarse estimation followed by a refinement in the spatial domain [6]. Background matching methods are based on the block matching algorithm (BMA), but generalized to the whole background. They are easy to implement but lack both estimation accuracy and efficiency.

The hierarchical differential approach is efficient and effective for GME [4]. Hierarchical differential methods start by building a frame pyramid using spatial prefilters and down-sampling. The estimation starts at the coarsest level of the pyramid and the result is considered an initial motion estimate that aids convergence at consecutive levels. N-step search and GM parameters prediction are some of the adopted search techniques for initial motion estimation. The estimation result is later refined using optimization techniques (e.g., Newton–Raphson method) and then projected onto the finer level of the pyramid and the optimization is repeated. This loop is continued until the finest level of the pyramid is reached. Since GME is a computationally intensive task, many efforts focus on reducing its computational complexity, e.g., [10] and [11] are modified faster versions of [1]. In [10], history GM information is used for parameters prediction instead of traditional N-step search in the initial motion estimation step. In [11], several improvements are proposed such as motion edge selection, residual block based outliers rejection and adaptive weighting function.

This paper proposes a robust GME method that is oriented to video object segmentation and is based on the hierarchical differential approach. We introduce three key contributions aimed at increased accuracy and reduced computational complexity. First, we propose a robust motion estimator for the first frame

Manuscript received July 13, 2006; revised January 22, 2008. This work was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Onur G. Guleryuz.

The authors are with the Electrical and Computer Engineering Department, Concordia University, Montréal, QC H3G 1MB Canada (e-mail: b_qi@ece.concordia.ca; moha_mo@ece.concordia.ca; amer@ece.concordia.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2008.921985

in the absence of object information from the previous frame. This robust estimator considers the distribution of candidate outliers and provides a good starting point for the proposed method. Second, we integrate the three-step search from [13] with a modified GM parameters prediction based on [10] to propose an initial motion estimation scheme faster than the one in [1] and both faster and more accurate than the one in [10]. Finally, we propose a robust GM estimator that utilizes binary residual (i.e., object) information from the previous frame to reject outliers caused by LM or noise, thus preventing them from interfering with the estimation in the current frame.

The rest of the paper is organized as follows. In Section II, we rederive the general hierarchical differential GME approach. In Section III, we introduce the proposed GME method. We compare the proposed and referenced methods in Section IV and conclude with Section V.

II. General Hierarchical Differential GME

A. Motion Model and Estimation Criteria

The purpose of a motion model is to describe the real motion between consecutive frames F_n and F_{n-1} of a video sequence at time instances n and $n-1$. Based on this model, motion parameters are estimated using a motion estimation algorithm that requires a motion model, an estimation criterion or objective function, and an optimization method. In GME, a single model applies to the whole frame (compared to block or region based LM estimation). There are different parametric motion models to describe GM, e.g., affine, projective, and bilinear. Depending on the selected model, we control the level of detail and precision of the estimated motion.

We use the following six-parameter affine model

$$\begin{aligned} dx_i &= a_0 + a_1 x_i + a_2 y_i \\ dy_i &= a_3 + a_4 x_i + a_5 y_i \end{aligned} \tag{1}$$

where (x_i, y_i) is the location of the i^{th} pixel in the current frame F_n, (dx_i, dy_i) is the motion vector of the corresponding pixel from the previous frame F_{n-1} to F_n, and $\mathbf{a} = (a_0, a_1, a_2, a_3, a_4, a_5)$ is a vector whose elements are the affine GM parameters. Note that dx_i and dy_i in (1) are functions of both x_i and y_i. We select the affine model because it describes the projected 2-D motion of most camera motions [12].

With the motion model defined, we incorporate it into the displaced frame difference (DFD) estimation criterion which is based on the *constant-intensity* assumption. This assumption states that the intensity remains constant along motion trajectories. The estimation error $E(\mathbf{a})$ based on the DFD is defined as

$$E(\mathbf{a}) = \sum_{i=1}^{N} |F_{n-1}(x_i + dx_i, y_i + dy_i) - F_n(x_i, y_i)|^s \tag{2}$$

where N is the total number of pixels in F_n and $s = 1$ in case of the sum of absolute difference (SAD) or $s = 2$ in case of the sum of square difference (SSD).

B. Newton–Raphson Method as Optimization Criterion

To estimate the GM-parameters vector $\mathbf{a}$, we use the Newton–Raphson method [12] to search for the value of $\mathbf{a}$ that minimizes the objective function in (2) with $s = 2$. Because the six-parameter affine motion model in (1) depends nonlinearly on $\mathbf{a}$, the minimization proceeds iteratively until meeting a stopping criterion [4]. Let $\mathbf{a}^t$ be the value of the parameters at iteration t. $E(\mathbf{a})$ can be approximated by its second-order Taylor series [14]

$$E(\mathbf{a}) \approx E(\mathbf{a}^t) + \mathbf{d}(\mathbf{a} - \mathbf{a}^t)^T + \frac{1}{2}(\mathbf{a} - \mathbf{a}^t)\mathbf{H}(\mathbf{a} - \mathbf{a}^t)^T \tag{3}$$

where $E(\mathbf{a}^t)$ is the DFD error in (2) at $\mathbf{a}^t$, $\mathbf{d} = \nabla E(\mathbf{a}^t)$ is the gradient of $E(\mathbf{a})$ at $\mathbf{a}^t$, $\mathbf{H} = \partial^2 E(\mathbf{a}^t)/\partial a_k \partial a_m$ is the Hessian of $E(\mathbf{a})$ at $\mathbf{a}^t$, and T is the transpose. The variables k and m are indexes for the motion model parameters (e.g., in the proposed method $k, m \in \{1, 2, \ldots, 6\}$).

By differentiating both sides of (3) with respect to $\mathbf{a}$, the gradient $\nabla E(\mathbf{a})$ of $E(\mathbf{a})$ is

$$\nabla E(\mathbf{a}) = \mathbf{d} + (\mathbf{a} - \mathbf{a}^t) \cdot \mathbf{H}. \tag{4}$$

A minimum of $E(\mathbf{a})$ occurs when $\nabla E(\mathbf{a}) = 0$ in (4) (assuming $\mathbf{H}$ is positive definite). We set (4) to zero and solve for $\mathbf{a}$ to build the *update* equation $\delta\mathbf{a}$ as

$$\begin{aligned} &\mathbf{d} + (\mathbf{a} - \mathbf{a}^t) \cdot \mathbf{H} = 0 \quad \Rightarrow \quad \mathbf{a} = \mathbf{a}^t - \mathbf{d} \cdot \mathbf{H}^{-1} n \\ &\delta\mathbf{a} = \mathbf{a} - \mathbf{a}^t = -\mathbf{d} \cdot \mathbf{H}^{-1}. \end{aligned} \tag{5}$$

The value of $\mathbf{a}$ at the next iteration is $\mathbf{a}^{t+1} = \mathbf{a}^t + \delta\mathbf{a}$. We rewrite (5) in a nonmatrix form as the system of nonlinear equations

$$\sum_m H_{km} \delta a_m = d_k \tag{6}$$

where d_k denotes elements in $\mathbf{d}$ and H_{km} elements in $\mathbf{H}$. Using singular value decomposition (SVD) [12], [14], we solve (6) for the increments δa_m and add them to the current set of parameters in $\mathbf{a}^t$ to get the next estimation $\mathbf{a}^{t+1}$. This operation is repeated until $E(\mathbf{a})$ falls under a preset threshold E_{TH} or a maximum number of iterations $t_{\max}$ is reached, depending on which event occurs first. $t_{\max}$ and E_{TH} provide a trade-off between accuracy and complexity.

The closer the initial values of the parameters in $\mathbf{a}$ are to the true values, the earlier Newton–Raphson method may converge. To aid in the convergence, we perform the estimation in a multiresolutional manner.

C. Multiresolutional Representation of GM Estimation

Multiresolutional or hierarchical representation of a video frame is a widely used strategy in video processing, where the original frame is rebuilt as a pyramid (see Fig. 1). The finest level is the original frame and the resolution between successive levels is reduced by half, both horizontally and vertically. After the frame pyramid with V levels is built using downsampling, the estimation of the parameters in $\mathbf{a}$ starts at the coarsest level and progresses to the next finer level until it reaches the finest

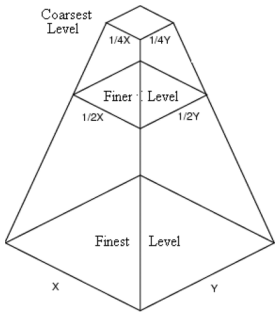


Fig. 1. Illustration of the structure of frame pyramid.

level. The result from the previous coarser level is projected to the next finer level as an initial solution.

There are three advantages of using the multiresolutional approach. First, details information at a finer resolution may interfere with the estimation, therefore, the result obtained at the coarsest level is more likely to be close to the true solution and is considered a good initial estimate. Second, if we define a search range R to search for corresponding pixels at the finest level, the search range scales down to $R/2^{V-1}$ at the coarsest level with a V-level pyramid. Third, since the projection of the result from the previous coarser level provides a good starting point for the search, the number of search iterations is reduced at the current level. Therefore, the total number of computations is smaller than that required by directly searching at the finest level, and the computational complexity is reduced.

III. Proposed Hierarchical Differential GME

While performing the estimation in a multiresolutional manner reduces complexity, the estimate of GM at the coarsest level has an impact on both the quality of the estimates and computational complexity. Moreover, outliers, caused by LM or noise, interfere with the estimation causing both loss in accuracy and increase in computations. This paper proposes, as shown in Fig. 2, 1) a fast initial estimation scheme combining three-step search and GM parameters prediction (Section III-A); 2) a robust estimator that uses residual object information from previous frames to reject outliers (Section III-B); and 3) a robust estimator that considers local neighborhoods in rejecting outliers when the residual binary information from previous frames is unavailable (Section III-C).

A. Initial Motion Estimation

At the coarsest level of each frame of the video sequence, we require initial GM parameters for the optimization method. The proposed initial estimation is divided into two phases. The first phase lasts for the first six frames and the second phase from the seventh frame on. In the first phase, we apply the three-step search from [13] to provide the initial estimate. In the second phase, we use a proposed GM parameter prediction scheme for the initial estimate that is based on [10] with modifications. Note that the three-step search gives reliable estimates but is time consuming and that the proposed GM parameters prediction is much faster. Thus, we integrate them to build an initial estimation scheme that is faster than the one in [1] and both faster and more accurate than the one in [10].

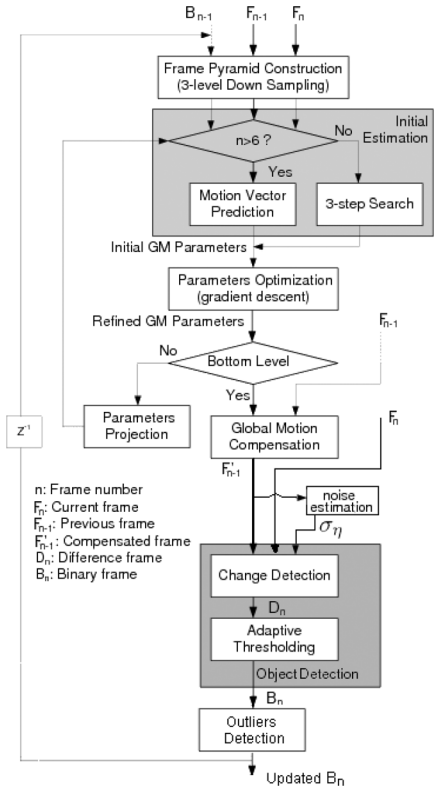


Fig. 2. Block diagram of the proposed GME method.

After the frame pyramid is built, the initial estimation is applied at the coarsest level of the pyramid where we assume that there is only translational camera motion with the two-parameters translation motion model

$$dx_i = a_0, \quad dy_i = a_3 \tag{7}$$

which is sufficient as a starting point to help in the convergence of the subsequent optimization steps. We use the three-step search to find the values of a_0 and a_3 in (7) that minimize the SAD [see (2)]. We use the SAD because it requires less computations than the SSD. Fig. 3 shows an example of how the three-step search is used to find these values. At the first step, the search range R is ± 4 pixels and the values with the minimum SAD in this range are found to be $[a_0, a_3] = [4, 4]$. Then, R is reduced down to ± 2 pixels in the second step and the values that minimize the SAD are $[a_0, a_3] = [4, 6]$. Finally, R is down to ± 1 pixel at the last step and the final values are $[a_0, a_3] = [3, 7]$. Only 25 trial vectors are used here but can cover a maximum displacement of ± 7 pixels at the coarsest pyramid level corresponding to a search range R of ± 28 pixels at the finest level. In most real video sequences, this range is large enough to cover the GM between two F_n and F_{n-1}.

After obtaining the translation parameters a_0 and a_3, we apply Newton–Raphson method (Section II-B) at each level of the frame pyramid starting from the coarsest and continuing to the finer level. The projection of the motion parameters from the current level onto the next one is performed by multiplying the translation parameters a_0 and a_3 by two and keeping the remaining parameters (a_1, a_2, a_4, and a_5) unchanged. While

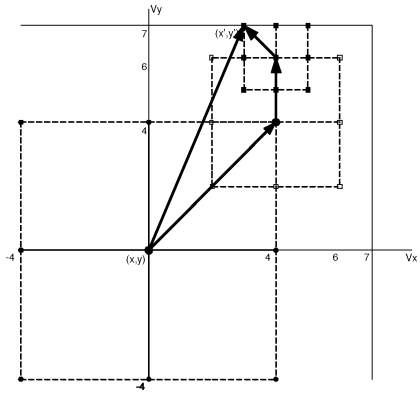


Fig. 3. Example of three-step search method. The final (minimum SAD) values are $a_0 = 3$ and $a_3 = 7$.

the three-step search provides a good initial estimation technique, it is computationally expensive. Therefore, we propose to use it as the initial estimator for the first six frames after which we start using GM parameters prediction. We propose to use only four predictors for the GM parameters to reduce the computational complexity of [10]. These predictors are

$$\begin{array}{ll} \text{zero predictor :} & \mathbf{a_n^{zero}} = 0 \\ \text{past predictor :} & \mathbf{a_n^{past}} = \mathbf{a_{n-1}} \\ \text{acceleration predictor :} & \mathbf{a_n^{acceleration}} = 2\mathbf{a_{n-1}} - \mathbf{a_{n-2}} \\ \text{long-term predictor :} & \mathbf{a_n^{average}} = \frac{1}{5}\sum_{i=1}^{5} \mathbf{a_{n-i}}. \end{array} \quad (8)$$

From (8), we select the prediction with the minimum SAD as in [10]. Note that to obtain all the predictors in (8), we need to have processed six frames. This is why we need the accurate three-step search as an initial estimator for the first six frames.

The motivations behind the selection of the predictors in (8) are: a) we achieve similar performance using the four predictors in (8) instead of the six in [10]; b) with this reduction, the proposed initial estimation is 1.5 times faster than in [10]; and c) they represent the most common scenarios in typical camera motion. For example, if the camera suddenly stops, the zero predictor $\mathbf{a_n^{zero}} = 0$ is the most accurate. If the camera is moving at a constant speed, the most accurate predictor is the past predictor $\mathbf{a_n^{past}} = \mathbf{a_{n-1}}$. If the camera is accelerating or decelerating, the acceleration effect is considered in the acceleration predictor $\mathbf{a_n^{acceleration}} = 2\mathbf{a_{n-1}} - \mathbf{a_{n-2}}$, making it the most accurate in this case. The long-term average predictor overcomes (by smoothing) unexpected sudden changes in GM.

The advantages of using the proposed GM parameters prediction starting from the seventh frame on instead of continuing to use the three-step search (as in [1]) are: 1) six GM parameters are predicted instead of two which results in better initial estimates, e.g., in zoom motion; 2) the camera motion usually has a predictable pattern over time; and 3) the three-step search requires 25 SAD calculations as opposed to eight making the proposed prediction 3.13 times faster.

B. Using Residual Information for Robust Estimation

One challenge in GME is that there is only one GM model applied to the whole frame, but not all the pixels in that frame experience the same GM. Therefore, pixels which have LM cause deviations in the SSD and bias the estimates of GM parameters. In other words, pixels experiencing LM are considered outliers, i.e., statistical data elements that deviate from the assumed global model [15], [16]. Outliers come also from noise. Robust estimation in GME aims at detecting and rejecting outliers introduced by LM or noise. Using Gaussian low-pass filtering while building the frame pyramid helps reduce outliers introduced by noise.

Outliers introduced by LM are detected as the set of pixels that are not undergoing GM. Remaining pixels are considered inliers [12]. Outliers are rejected from the next iteration in (5), and only the inliers are used for the rest of the estimation.

We propose a robust estimator based on the M-estimators [17] through minimization of

$$\text{SSD} = \sum_{i=1}^{N} \rho(E_i); \quad E_i = \left|F_{n-1}\left(x_i', y_i'\right) - F_n(x_i, y_i)\right|^2 \quad (9)$$

where $\rho(.)$ is symmetric positive-definite function [18], $x_i' = x_i + dx_i$, $y_i' = y_i + dy_i$, and E_i is the error of the i^{th} pixel. To propose an effective $\rho(.)$ for outliers rejection in this paper, we use binary residual frames $\{B_n\}$. Assuming that the GM is successfully compensated, the residual information B_n between F_n and F'_{n-1} (GM-compensated F_{n-1}) contains the objects and the newly appeared background. We use B_n to reject outliers when estimating the GM parameters of the next frame F_{n+1} (see Fig. 2). B_n is obtained by applying an object detection method [19] which consists of change detection to obtain the difference frame D_n between F_n and F'_{n-1} and thresholding of D_n to obtain the binary residual frame B_n.

To prevent outliers misclassification in B_n, we propose the following outliers detection strategy. We represent B_n as a set of nonoverlapped blocks b_l of size $W \times W$ or

$$B_n = \bigcup_{l=1}^{L} b_l; \;\; b_l \cap b_g = \phi, \;\; l \neq g \quad \text{and} \quad l, g \in \{1, 2, \ldots, L\} \quad (10)$$

where ϕ is the empty set and L is the number of blocks in B_n. Let $S_{B_n} = \{b_1, b_2, \ldots, b_L\}$ be the set of all blocks in B_n and $S_Z = \{Z(b_1), Z(b_2), \ldots, Z(b_L)\}$ be the set of the numbers of object pixels (white pixels) in each block. $Z(.)$ is an operator that returns the number of object pixels for a given block b_l. Also, let $S_{BB_n} \subset S_{B_n}$ be the set of all blocks on the boundary of the current frame (i.e., boundary blocks). The set of candidate outliers blocks $S_{B_n}^C \subset S_{B_n}$ is the set which includes the blocks with the $\omega\%$ (e.g., $\omega = 30$) largest numbers of object pixels or $Z(b_l)$. We decompose $S_{B_n}^C$ into $S_{B_n}^O$ and $S_{B_n}^I$ (i.e., $S_{B_n}^C = S_{B_n}^I \cup S_{B_n}^O$), where $S_{B_n}^O$ is the set of outliers blocks and $S_{B_n}^I = S_{B_n}^C - S_{B_n}^O$ is the set of inliers blocks. $S_{B_n}^I$ and $S_{B_n}^O$ are populated using three rules. These rules are: 1) a candidate block is an outliers block if it is a boundary block in F_n; 2) a candidate block is an outliers block if it is not a boundary block but has more than two candidate outliers blocks in its eight-neighborhood; and 3) a candidate block is an outliers block if it has at least one outliers block in its eight-neighborhood after the first two rules are applied.

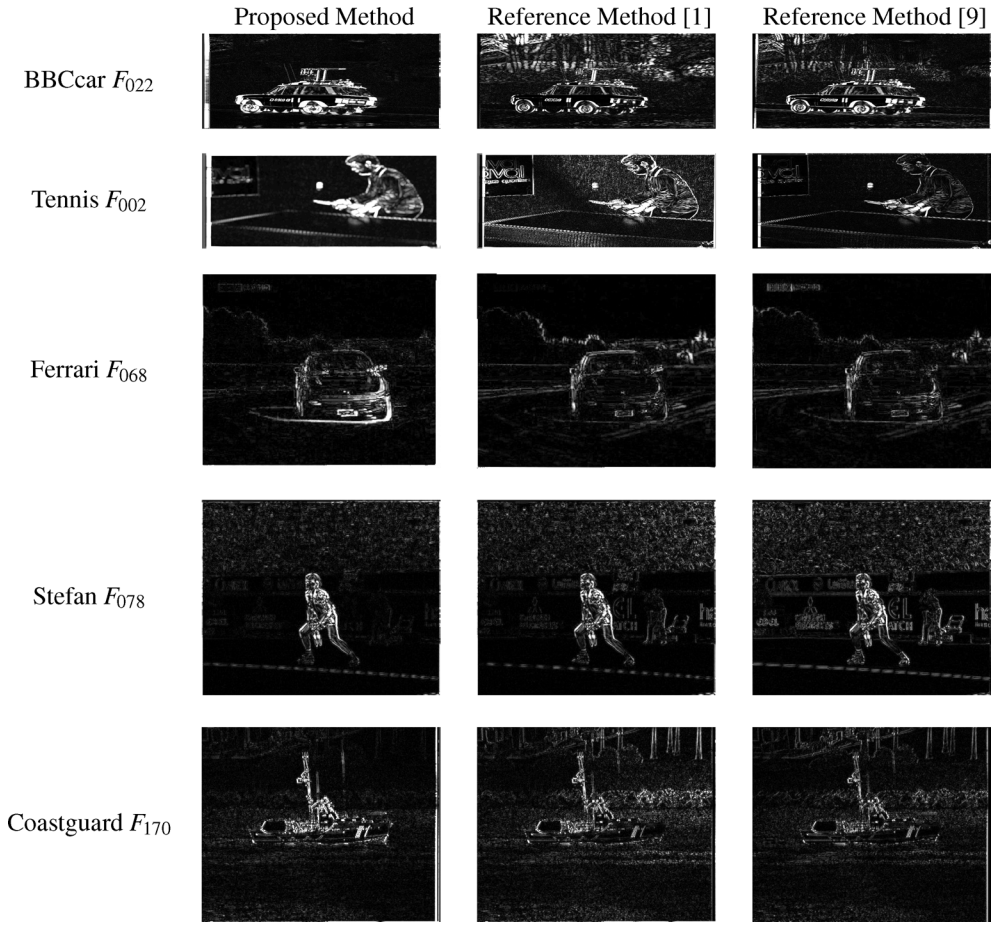


Fig. 4. Comparison: results of D_n using the proposed and the referenced methods [1] and [9], respectively.

We realize the three outliers rejections rules using two equations applied to all $b_l \in S^C_{B_n}$. First, we apply rule one and two to produce an intermediate set of outliers blocks, $S^{\hat{O}}_{B_n}$, using

$$S^g_l = \{b_g : b_g \text{ neighbor of } b_l\}$$
$$b_l \in S^{\hat{O}}_{B_n} \quad \text{if} \quad \begin{cases} b_l \in S_{BB_n} \vee \\ \left|S^g_l \cap S^C_{B_n}\right| \geq 3 \ \wedge \ b_l \notin S_{BB_n} \end{cases} \tag{11}$$

where $|.|$ represents the set cardinality in this equation. Then, we produce the final set of outliers blocks $S^O_{B_n}$ with

$$b_l \in S^O_{B_n} \text{ if } \begin{cases} b_l \in S^{\hat{O}}_{B_n} \vee \\ \left(b_l \in S^C_{B_n} - S^{\hat{O}}_{B_n}\right) \wedge \left(S^g_l \cap S^{\hat{O}}_{B_n} \neq \phi\right) \end{cases} \tag{12}$$

which realizes rule three. The proposed outliers detection strategy thus classifies all pixels in B_n either as pixels in the blocks of $S^O_{B_n}$, which are set to 1 indicating outliers pixels undergoing LM, or inliers pixels in the blocks of $S^I_{B_n}$, which are set to zero indicating pixels undergoing GM. Consequently, the proposed robust estimator in (9) becomes

$$\text{SSD} = \sum_{i=1}^{N} \rho(E_i), \ \rho(E_i) = \begin{cases} E_i & : (i \in b_l) \wedge \left(b_l \in S^I_{B_n}\right) \\ 0 & : (i \in b_l) \wedge \left(b_l \in S^O_{B_n}\right) \end{cases} \tag{13}$$

where i is the i^{th} pixel in B_n at (x_i, y_i) and E_i is as in (9).

To avoid propagating estimation errors if the GME in F_{n-1} fails (e.g., the total number of the outliers blocks changes drastically), we check the percentage change t_r in outliers count between time instances n and $n-1$

$$t_r = \frac{|P_n - P_{n-1}|}{P_{n-1}}, \quad P_n = \left|S^O_{B_n}\right|, \quad P_{n-1} = \left|S^O_{B_{n-1}}\right| \tag{14}$$

with P_n and P_{n-1} as the number of the outliers blocks in B_n and B_{n-1}, respectively. If $t_r > t_P$, the previous residual information B_{n-1} and $\mathbf{a}_{n-1}$ are used instead as follows:

$$\mathbf{a}_n = \mathbf{a}_{n-1}; \quad B_n = B_{n-1}$$
$$P_n = \lambda P_n + (1-\lambda) P_{n-1} \tag{15}$$

where $\mathbf{a_n}$ and $\mathbf{a_{n-1}}$ are the GM parameters of F_n and F_{n-1}, respectively, $0 \leq \lambda \leq 1$ is a confidence measure in current estimation, and $0.4 < t_P < 1$. Note that in (14) we calculate an initial value for P_n using the current number of outliers pixels. We update the value of P_n using (15) to temporally stabilize it subject to $t_r > t_p$.

The two advantages of using residual information in (13) and (14) are: 1) B_n contains pixels that do not undergo GM and, thus, is more accurate in rejecting outliers than a statistical estimate; and 2) no extra computation is involved since the residual information comes from the object segmentation used.

The above outliers rejection handles outliers resulting from LM. To reject outliers due to noise, we adapt the object detection step in Fig. 2 to noise. Since noise is modeled as additive

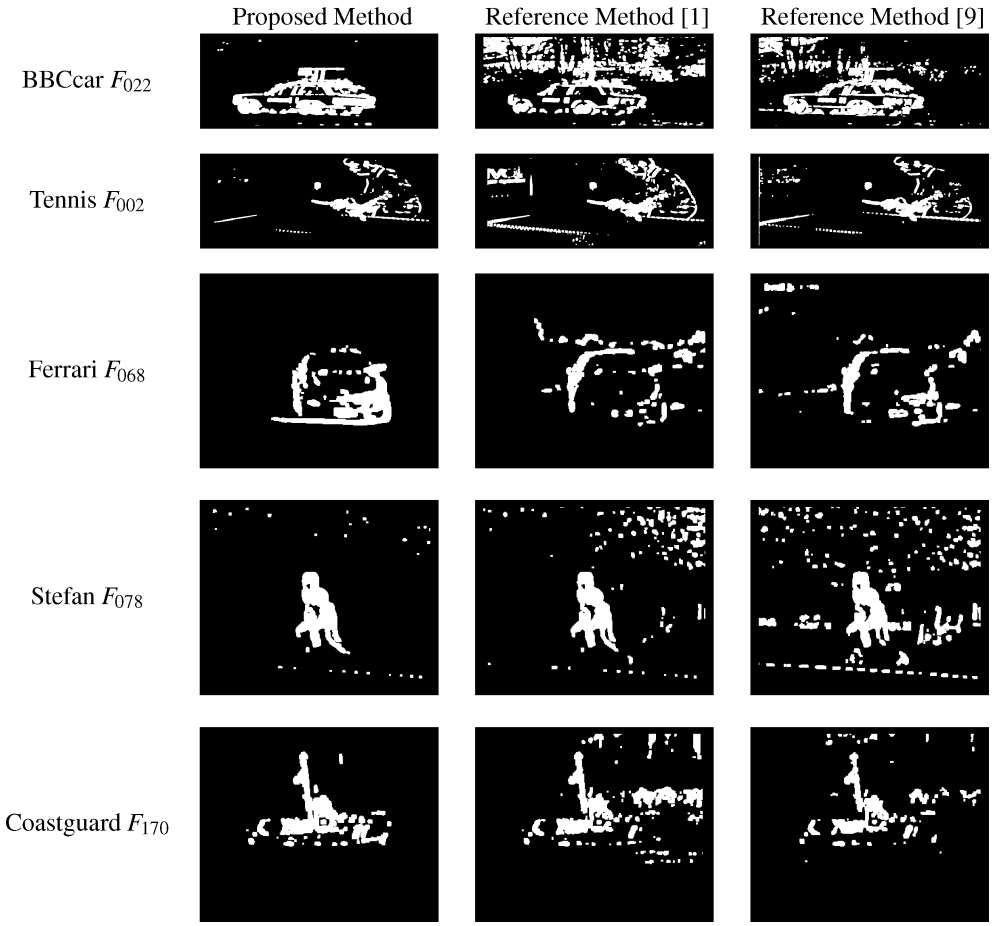


Fig. 5. Comparison: results of B_n using the proposed and the referenced methods [1] and [9], respectively.

white Gaussian noise (AWGN), only its standard deviation σ_η is needed. The choice of AWGN as a model for noise is motivated by AWGN being the most common noise model for terrestrial TV broadcasting [22] and because according to the central limit theorem, the aggregate effect of the high count of photon arrivals at CCD camera sensors can be well approximated by Gaussian statistics.

The effectiveness of the proposed outliers rejection strategy is mainly because video objects are spatio-temporally correlated and because LM is the main source of outliers for GME. The rules of (11) and (12) recognize this correlation. We exploit temporal correlation by selecting outliers pixels in F_n using object pixels in F_{n-1} and spatial correlation by re-examining our selection of outliers pixels in F_n. To elaborate, we distinguish between three types of pixels in F_n: 1) normal pixels; 2) candidate (probable) outliers pixels; and 3) outliers pixels (most probable). A first step in outliers detection is to find candidate outliers pixel: a pixel in the current frame F_n, which was surrounded by outliers pixels in the previous frame F_{n-1}, is a good candidate to be an outliers pixel in F_n. After establishing candidacy in F_n, the first rule of (11) states that a candidate outliers pixel that is located close to the boundary of the current frame is an outliers pixels. The reason is that this pixel most probably belongs to newly appeared background. The second rule of (11) states that a candidate outliers pixels whose immediate and extended neighbors are candidate outliers pixels is an outliers pixels. The reason is that this candidate outliers pixels is most likely a moving outliers pixels from the previous frame. After applying the first and second rules, we have a mixture of candidate outliers pixels and outliers pixels in F_n. This is when exploitation of spatial correlation comes with the third rule in (12). This rule states that a candidate outliers pixel that is close to outliers pixels in the current frame is an outliers pixel.

C. Robust Estimation for the First Frame

The robust estimator in Section III-B cannot be applied for the first compensated frame F_1' since no previous binary frame B_0 exists. Robust estimation in F_1 is, however, of significant importance for algorithm convergence. For F_1, we modified (13) as follows. Instead of considering the pixels in the error E_i individually, neighboring pixels are also considered when rejecting the outliers. Thus, we propose to reject outliers in F_1' as follows.

1) Sort the set $S^E = \{E_i, 1 \leq i \leq N\}$ for F_1' in descending order (N is the total number of pixels in a frame).
2) Ignore the top $\theta\%$ of S^E where $5 < \theta < 15$ and take the threshold t_E to be the next $E_i \in S^E$.
3) Classify a pixel $i \in F_1'$ as an inlier only if:
   a) $E_i \leq t_E$;
   b) i has m_i neighbors ($m_i > 6$) in its eight-neighborhood where each neighboring pixel j satisfies $E_j < t_E$.

As a result, (13) for F_1' changes to

$$\text{SSD} = \sum_{i=1}^{N} \rho(E_i), \quad \rho(E_i) = \begin{cases} E_i & : (E_i \le t_E) \wedge (m_i > 6) \\ 0 & : \text{otherwise.} \end{cases} \tag{16}$$

The effectiveness of the proposed robust estimator for the first frame is due to its consideration of the distribution of candidate outliers which provides an accurate starting point for the subsequent steps.

IV. Results and Comparison

To evaluate the performance of the proposed method, we compared it to the referenced methods [1], [9], [10]. Since [10] is a faster version of [1], the results are similar based on [10] and our simulations. Therefore, we do not show them in this paper. Simulations were carried out using standard test sequences with GM and LM: *BBCcar*, *Tennis* in PAL (720 × 576) format and *Ferrari*, *Stefan*, and *Coastguard* in CIF (352 × 288). *BBCcar* shows a fast moving jeep with a pan and small rotational camera motions and moving tree leaves. *Tennis* has camera zoom-in motion to a table-tennis player. *Ferrari* shows a fast moving car with complex camera motion and the object is large and its motion is dominant. *Stefan* shows a tennis player with inconsistent but fast camera motion, which is difficult to predict and there is LM in the audience region which interferes with the GME. *Coastguard* shows two ships cruising in opposite directions with mainly translational camera motion and the moving ships also cause the water to move which interferes with GME.

A. Algorithm Parameters

We set the parameters $t_{\max}$ and E_{TH} required as stopping criteria for (6) to $t_{\max} = 32$ and $E_{TH} = 0.001$, which we found suitable for video object segmentation. Increasing the value of $t_{\max}$ beyond 32 does not lead to a significant increase in accuracy compared to the increase in computations.

We use a three level frame pyramid (Fig. 1) because our simulation shows $V = 3$ is good for PAL frame size or lower (e.g., CIF). For frame size larger than PAL, V should be increased because we may utilize a better initial estimate by yet descending for an extra level. The search range $R = \pm 28$ in initial motion estimation (Section III-A) depends mainly on the speed of the GM and the frame rate. For general video sequences with GM, $R = \pm 28$ is enough to cover the expected amount of motion between two consecutive frames. For faster camera motion or when capturing at a low frame rate, R should be increased to compensate for the increased distance a pixel can move between two consecutive frames.

For GM parameters prediction in the initial motion estimation in Section III-A, we use a fifth order predictor, i.e., long-term predictor in (8), because a lower order does not allow for enough smoothing of fluctuating or unstable GM. A higher order predictor increases the computational time and is not necessary since motion changes over time.

In (10), we use $W = 8$ for CIF frames and $W = 12$ for PAL/NTSC frames. t_P for (14) is set in relation to the amount of GME error that can be tolerated by the target application. We set $t_P = 0.4$. For application with low tolerance to GME error, t_P should be increased. The lower bound of $t_P = 0.4$ is important to avoid unnecessary frequent adjustments. We experimentally set the confidence measure in the current estimate in (15) to $\lambda = 0.3$, and θ in Section III-C to 10%.

B. Subjective Results

Figs. 4 and 5 show selected output frames of each test sequence. Change-detection frame D_n between F_n and F'_{n-1}(see Fig. 2), followed by binary frame B_n are given to show the effect of using different GME methods on object segmentation. As can be seen, the proposed method is more effective than the referenced methods [1], [9] in separating objects from the background. For example, the complex and dominant camera motion in *Ferrari* makes the referenced methods fail to identify outliers, while the proposed method is successful. Also, the proposed method performs better than the referenced method despite the fast motion in *Stefan* and the moving water in *Coastguard*.

C. Objective Results

We use four criteria to evaluate the proposed GME objectively: mean absolute error (MAE), segmentation quality measures [20] (temporal color histogram difference and spatial color contrast along object boundary), stability of percentage of object (white) pixels in B_n, and computational complexity.

The MAE between the current frame F_n and GM compensated previous frame F'_{n-1} is

$$\begin{aligned} \text{MAE} &= \frac{1}{N}\text{SAD} \\ &= \frac{1}{N}\sum_{i=1}^{N} |F_{n-1}(x_i+dx_i, y_i+dy_i) - F_n(x_i, y_i)| \end{aligned} \tag{17}$$

where N is the number of pixels in the frame. Fig. 6 shows sample MAE comparison between our method and the referenced GME methods [1], [9]. As can be seen, the proposed method has significantly lower MAE than both referenced methods due to improved robustness in outliers rejection.

Fig. 7 shows the robustness to noise of the proposed method and referenced methods [1], [9] for the *Coastguard* which was corrupted with AWGN of 25- and 30-dB peak signal to noise ratio (PSNR). The proposed method is more robust (lower MAE) to noise than the referenced methods for both noise levels. Note that the higher the noise level (i.e., the lower the PSNR), the more outliers are present which may affect the estimation causing the MAE to increase. Note also that the proposed GME method is designed to estimates the perceived GM resulting from camera motion and does not explicitly take video noise into account. Thus, the GMC compensates for the GM and not for the noise. As a consequence, the GM compensated frame still includes noise. To implicitly compensate for the noise, we adapt the object detection step (see Fig. 2) to noise that we estimate using the method in [21]. This considerably improves the performance of the proposed method for noisy videos as can be seen in Fig. 6. Also, Fig. 6 shows the superiority of the proposed method compared to [1] and [9] due to noise-adaptive outliers detection where the difference in performance remains stable for all noise levels.

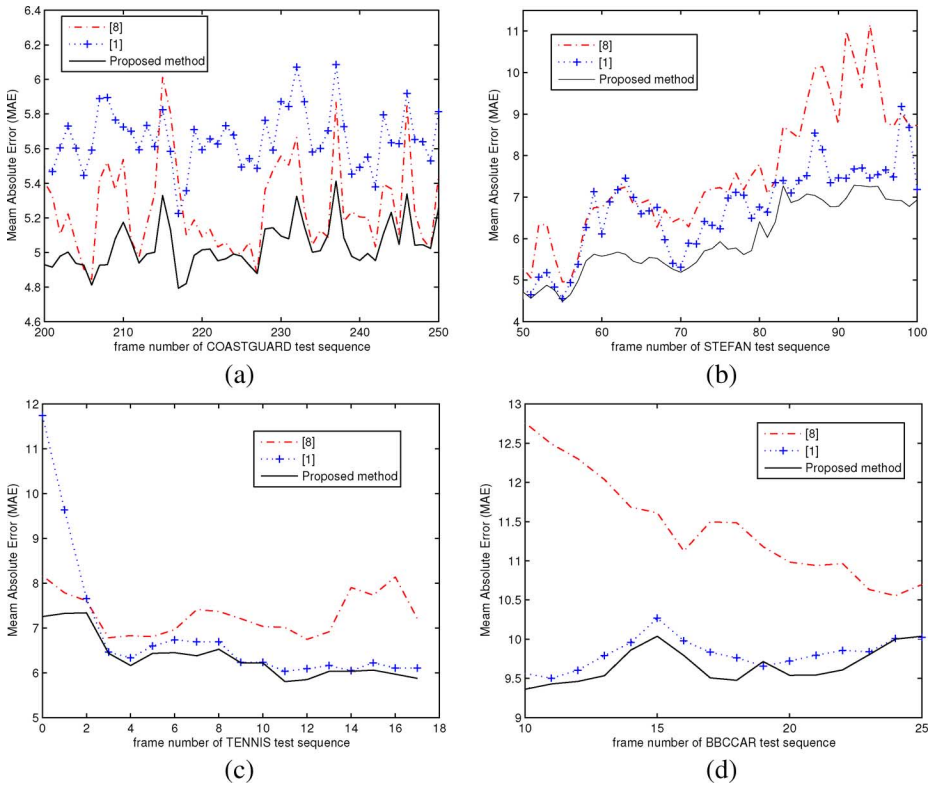


Fig. 6. MAE comparison of the proposed and the reference GME methods [1] and [9] for various test sequences. (a) Coastguard. (b) Stefan. (c) Tennis. (d) BBCCar.

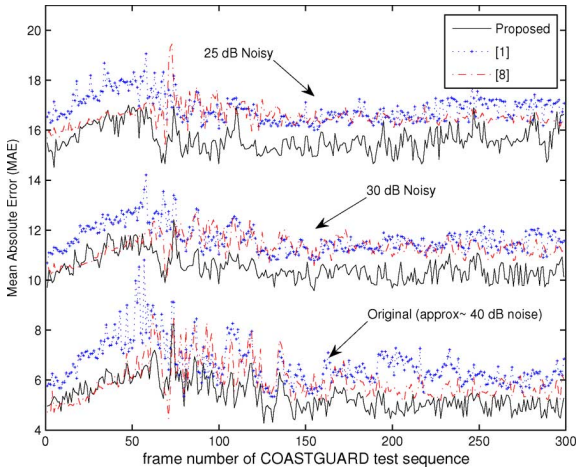


Fig. 7. MAE comparison of the proposed and the referenced GME methods [1] and [9] under 25 and 30 dB noise for *Coastguard* video.

Assuming that the color histogram of video objects is temporally stationary and is different from the color histogram of the background, then the difference between this histogram in F_{n-1} and F_n should represent the temporal stability of the object segmentation. Moreover, assuming that object boundaries are coincided with color boundaries, if the objects are successfully segmented, there should be a color contrast difference between the pixels along the segmented object boundaries. Therefore, spatial color contrast along object boundary measures the quality of object segmentation. We have, thus, integrated both the proposed and referenced methods [1], [9] into the object segmentation method in [19]. Then, we evaluated the segmentation output ("updated" B_n in Fig. 2) of the GME methods using the temporal color histogram difference and spatial color contrast along object boundary measures from [20]. Note that with these measures, a lower value indicates improved object segmentation. Fig. 8(a)-(c) shows that the color contrast along object boundary measure for the proposed method is lower than the referenced methods [1], [9] for the *Stefan*, *Ferrari* and *BBCCar* test sequences, indicating improved segmentation performance of our method. Fig. 8(d) shows sample results for the temporal color histogram difference measure which is significantly lower and more temporally stable when using the proposed method compared to the referenced methods [1], [9].

Assuming object features are temporally consistent in a video shot, the percentage of the segmented object pixels should change gradually throughout the sequence. Fig. 9 shows sample comparison of the percentage of white (object) pixels in B_n. Note that the evaluation criterion in this figure is the stability, not the lower percentage. Our method is more stable to compensate GM and preserve LM than [1], [9]. Fig. 9 also shows the standard deviation of each curve and the proposed method achieves the lowest standard deviation (approximately three times lower).

GME is a time-consuming task and efficiency is another important evaluation criterion. The proposed method is about 1.6 times faster than [1] under an Intel(R) Xeon(TM) CPU 2.40GHz operated by Linux. More specifically, see the following.

1) The proposed initial motion estimation requires on the long run eight SAD as opposed to 25 SSD calculations in [1], making it 7.82 times faster (for CIF) than in [1].

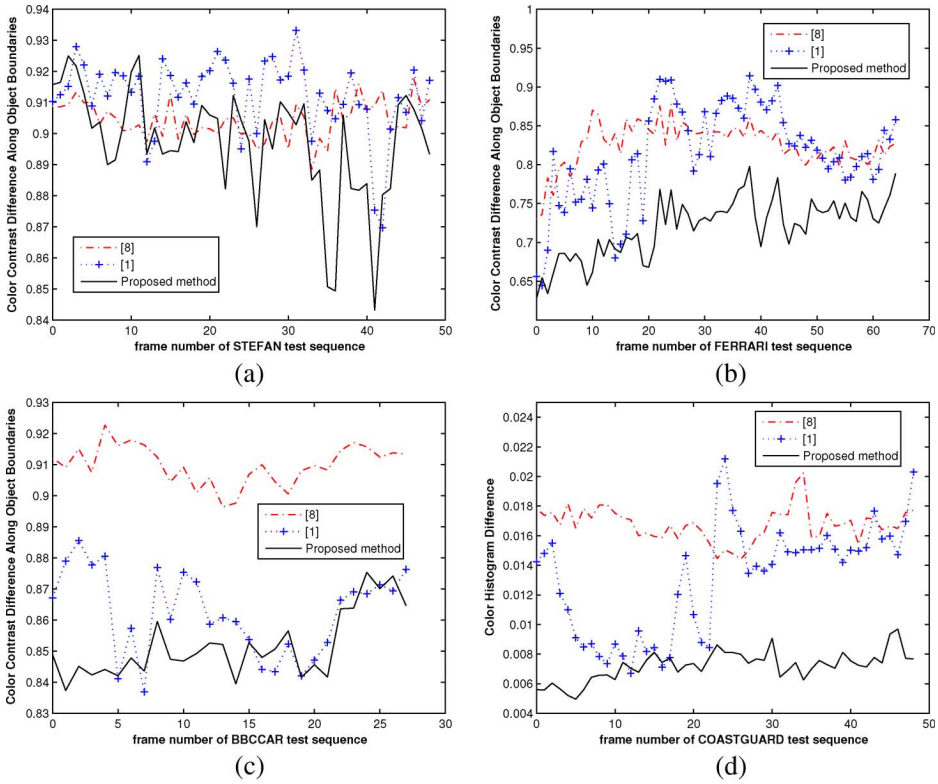


Fig. 8. (a)–(c) Color contrast difference along object boundaries and (d) color temporal histogram difference as segmentation measures among proposed and referenced methods [1] and [9]. (a) Stefan. (b) Ferrari. (c) BBCCar. (d) Coastguard.

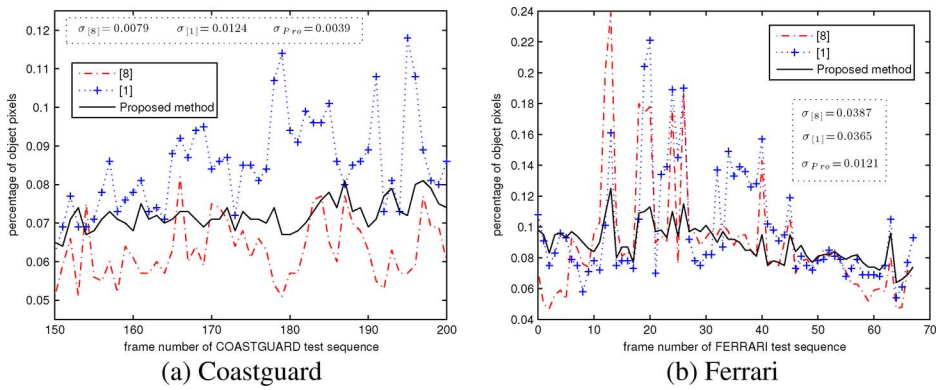


Fig. 9. Stability criteria: Percentage of white pixels for *Coastguard* and *Ferrari* for the proposed and referenced methods [1] and [9]. The proposed method shows the most temporally stable results with the least standard deviation.

2) The proposed parameter optimization is 1.26 times faster than [1] due to its smaller number of GM parameters.
3) The proposed outliers rejection is 1.38 times faster than [1] due to the elimination of histogram processing.

The proposed method is 1.2 times faster than [10] due to two reasons. First, we use four candidates to predict the GM parameters compared to six candidates in [10]. Second, the proposed robust estimator archives better GME estimates which improves the predictions in the next frames and makes the parameter optimization process faster. Note that our method is also 2.5 times faster than [9].

V. Conclusion

This paper proposed a fast and robust GME method oriented to object segmentation in video sequences. It is based on the hierarchical differential approach with several improvements. First, the proposed method integrates three-step search with GM parameters prediction in initial motion estimation for improved accuracy and reduced complexity. Second, it uses residual (object) information from the previous frame to reject outliers when estimating the GM parameters of the current frame. Third, it considers local neighborhoods in rejecting outliers in the first frame without object information from the previous frame. Both subjective and objective results show that the proposed method is more robust, faster, and more suitable for object segmentation than the referenced methods with mutual benefit between the segmentation of motion compensated frames and the estimation of motion in the form of a feedback loop.

REFERENCES

[1] F. Dufaux and J. Konrad, "Efficient, robust and fast global motion estimation for video coding," *IEEE Trans. Image Process.*, vol. 9, no. 3, pp. 497–501, Mar. 2000.

[2] E. Izquierdo, J. H. Xia, and R. Mech, "A Generic video analysis and segmentation system," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, May 2002, vol. 4, pp. 3592–3595.

[3] H. Xu, A. A. Younis, and M. R. Kabuka, "Automatic moving object extraction for content-based applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 6, pp. 796–812, Jun. 2004.

[4] MPEG-4 Video Verification Model Version v18.0, ISO/IEC JTC1/SC29/WG11 Coding of Moving Pictures and Audio N3908. Pisa, Italy, Jan. 2001.

[5] L. Hill and T. Vlachos, "On the estimation of global motion using phase correlation for broadcast applications," in *Proc. IEE Int. Conf. Image Processing and Its Applications*, 1999, vol. 2, pp. 721–725.

[6] E. Kumar, M. Biswas, and T. Q. Nguyen, "Global motion estimation in frequency and spatial domain," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, May 2004, vol. 3, pp. 333–336.

[7] F. Moscheni, F. Dufaux, and M. Kunt, "A new two-stage global/local motion estimation based on a background/foreground segmentation," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, May 1995, vol. 4, pp. 2261–2264.

[8] C. Hsu and Y. Tsan, "Mosaics of video sequences with moving objects," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2001, vol. 2, pp. 387–390.

[9] J. S. Lee, K. Y. Rhee, and S. D. Kim, "Moving target tracking algorithm based on the confidence measure of motion vector," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2001, vol. 1, pp. 369–372.

[10] W. C. Chan, O. C. Au, and M. F. Fu, "Improved global motion estimation using prediction and early termination," in *Proc. IEEE Int. Conf. Image Processing*, Sep. 2002, vol. 2, pp. 285–288.

[11] Y. He, B. Feng, S. Yang, and Y. Zhong, "Fast global motion estimation for global motion compensation coding," in *Proc. IEEE Int. Symp. Circuits and Systems*, May 2001, vol. 2, pp. 233–236.

[12] Y. Wang, J. Ostermann, Y.-Q. Zhang, Y.-Q. Zhang, and J. Ostermann, *Video Processing and Communications*. Englewood Cliffs, NJ: Prentice-Hall, 2002, vol. 1, pp. 141–193.

[13] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion-compensated interframe coding for video conferencing," in *Proc. IEEE National Telecommunication Conf.*, 1981, pp. 531–534.

[14] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C*. Cambridge, U.K.: Cambridge Univ. Press, 1992, pp. 59–71.

[15] P. J. Huber, *Robust Statistics*. New York: Wiley, 1981.

[16] G. L. Shevlyakov and N. O. Vilchevski, Robustness in Data Analysis: Criteria and Methods VSP BV, pp. 6–10, 2002.

[17] P. Meer, D. Mintz, A. Rosenfeld, and D. Y. Kim, "Robust regression methods for computer vision: A review," *Int. J. Comput. Vis.*, vol. 6, no. 1, pp. 59–70, 1991.

[18] C. V. Stewart, "Robust parameter estimation in computer vision," *SIAM Rev.*, vol. 41, no. 3, pp. 513–537, 1999.

[19] A. Amer, "Memory-based spatio-temporal real-time object segmentation," in *Proc. Int. Symp. Electronic Imaging, Conf. Real-Time Imaging*, Jan. 2003, vol. 5012, pp. 10–21.

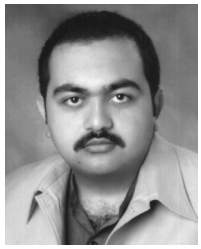
[20] C. E. Erdem, B. Sankur, and A. M. Tekalp, "Performance measures for video object segmentation and tracking," *IEEE Trans. Image Process.*, vol. 13, no. 7, pp. 937–951, Jul. 2004.

[21] A. Amer and E. Dubois, "Fast and reliable structure-oriented video noise estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 1, pp. 113–118, Jan. 2005.

[22] G. de Haan, T. G. Kwaaitaal-Spassova, M. Larragy, and 0. A. Ojo, "Memory integrated noise reduction IC for television," *IEEE Trans. Consum. Electron.*, vol. 42, no. 3, pp. 175–181, May 1996.

Bin Qi has received the B.Sc. degree in electrical engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 1992, and the M.A.Sc. degree in electrical and computer engineering from Concordia University, Montréal, QC, Canada, in 2005.

From 1992 to 2001, she was with China Telecom as a System Engineer and a Team Leader. She was involved in several projects, such as Shanghai Public Video Conference Systems. Her research interests include global motion estimation and video object segmentation. Currently, she is a Computer Engineer with SkyTrac Systems, BC, Canada.



Mohammed Ghazal (S'05) received the B.Sc. degree (highest honors) in computer engineering from the American University of Sharjah, Sharjah, United Arab Emirates (UAE), in 2004, and the M.A.Sc. degree in electrical and computer engineering from Concordia University, Montréal, QC, Canada, in 2006, where he is currently pursuing the Ph.D. degree in the Electrical and Computer Engineering Department.

He has authored ten publications and his research interests include video noise estimation and reduction and video motion estimation.

Aishy Amer (SM'07) received the Ph.D. degree in telecommunications from INRS, Université du Québec, Montréal, QC, Canada, in 2001.

Currently, she is an Associate Professor with the Department of Electrical and Computer Engineering, Concordia University, Montréal. From 1995 to 1997, she was with Siemens-AG/Munich and Dortmund University as a Research and Development Associate. Her research interests include video surveillance, advanced TV-systems, object segmentation, tracking, and noise reduction. She has four patents and over 50 publications. One of her algorithms (on noise reduction) is implemented in TV chip-sets of Siemens-AG, Munich, Germany.

Dr. Amer is serving as an Associate Editor for the *Springer Journal of Real-Time Image Processing (JRTIP)*.