Signal Processing: *Image Communication* 28 (2013) 1420–1434

Contents lists available at ScienceDirect

Signal Processing: *Image Communication*

journal homepage: www.elsevier.com/locate/image

Motion-based object segmentation using hysteresis and bidirectional inter-frame change detection in sequences with moving camera

Marina Georgia Arvanitidou*, Michael Tok, Alexander Glantz, Andreas Krutz, Thomas Sikora

Communication Systems Group of Technische Universität Berlin, Einsteinufer 17, Sekr. EN 1, 10587 Berlin, Germany

ARTICLE INFO

Article history:
Received 10 October 2012
Received in revised form
10 September 2013
Accepted 10 September 2013
Available online 1 October 2013

Keywords:
Inter-frame change detection
Object segmentation
Hysteresis
Global motion estimation

ABSTRACT

We present an unsupervised motion-based object segmentation algorithm for video sequences with moving camera, employing bidirectional inter-frame change detection. For every frame, two error frames are generated using motion compensation. They are combined and a segmentation algorithm based on thresholding is applied. We employ a simple and effective error fusion scheme and consider spatial error localization in the thresholding step. We find the optimal weights for the weighted mean thresholding algorithm that enables unsupervised robust moving object segmentation. Further, a post processing step for improving the temporal consistency of the segmentation masks is incorporated and thus we achieve improved performance compared to the previously proposed methods. The experimental evaluation and comparison with other methods demonstrate the validity of the proposed method.

1. Introduction

Object segmentation is an essential step for many applications such as content retrieval, interactive multimedia services and object-based video coding. Motion is among the salient characteristics that the human visual system perceives, and thus it comprises a very powerful feature that the image processing community has adopted to address object segmentation tasks.

1.1. Existing approaches

A common approach for dealing with the object segmentation task [1] is change detection. Given a set of video frames of the same scene, the change detection mask is the set of pixels that are "significantly different" between frames. The change detection mask may result from a combination of underlying factors, including appearance or disappearance of objects, motion of objects relative to the background, or shape changes of objects. A typical method is background subtraction, involving calculating a background model, subtracting each frame from it and processing the resulting information [2,3]. Many background models have been introduced to deal with several issues, such as small motion activity [4], complex scenes [5], lighting variations, and recently benchmark datasets that focus on such issues have been created and published [6] for further reference. These approaches rely on a training step to learn the reference background model and usually they do not take into account the temporal relations between frames.

Inter-frame change detection algorithms employ the difference between temporal neighboring video frames to perform object segmentation, and no background modeling is involved. In this category, many algorithms have been proposed that focus on inter-frame change detection employing one adjacent frame. Kim and Hwang [7] derive

[*] Corresponding author. Tel.: +49 3031428917.
E-mail address: arvanitidou@nue.tu-berlin.de (M.G. Arvanitidou).

0923-5965/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.image.2013.09.008

an edge map from the difference between two successive frames and after removing edge points which belong to the previous frame, the remaining edge map is used to extract the video object plane. The algorithm involves two thresholds, that are set heuristically and also requires manual definition of a background edge map. In the segmentation model proposed in [8] the change detection mask is obtained using the difference between two successive frames and a local thresholding relaxation technique is employed to enforce spatial continuity. In order to increase temporal stability, a buffer is incorporated such that the last *N* change detection masks participate in the final segmentation decision step. In the case of sequences with moving camera, Qi et al. [9] presented a Global Motion Estimation (GME) approach that is using one adjacent frame towards video object segmentation. This GME approach is employed to perform object segmentation, which is also used internally to predict and reject outliers for GME in the following frame.

Consideration of only one adjacent frame for inter-frame change detection yields partial foreground detection, since only edges of the corresponding motion direction are detected. The *double change* detection approach – based on three successive frames – has been adopted to overcome this issue. Kameda and Minoh proposed in [10] to use error frames from both directions. They end up with two binary masks and fuse them using the intersect operation. Shih et al. [11] employ three adjacent frames in a similar manner and additionally perform motion compensation followed by optical flow estimation to address cases with non-stationary background. Huang et al. [12] employ three successive frames for change detection in the wavelet domain and obtain the moving object edge map after applying the intersect operation between the edge maps of *significant difference pixel* of each pair in each direction. Liu et al. [13] employ a similar technique to [12] use three successive frames but they use fuzzy C-means clustering instead of frame difference to classify motion features. The change detection masks are obtained in the wavelet domain after applying the intersect operation to the binary masks of each directions.

1.2. Proposed approach

In this contribution, we focus on *inter-frame change detection* algorithms and specifically under the presence of camera motion and we propose a segmentation algorithm based on inter-frame change detection that employs a bidirectional fusion scheme of the global motion compensated error. We demonstrate that our error fusion scheme outperforms the intersection fusion scheme that is commonly employed. At first step, global motion is compensated between temporally adjacent video frames and between their corresponding motion vector fields. The compensated frames are employed for generating global motion compensated error maps and the compensated motion vector fields are employed in the post-processing step for improving temporal consistency. After low-pass filtering of error maps, hysteresis thresholding follows that exploits spatial connectivity of global motion compensated errors. In this step, we avoid setting the thresholding parameters heuristically, which is commonly found in the literature. Rather, we study the problem of optimal weight selection for hysteresis thresholding of error images using the weighted mean thresholding approach proposed in [14] and extended in [15]. Furthermore, we propose a novel adaptive scheme for mitigating the negative effect of temporal inconsistencies while avoiding the incorporation of a buffer. In this way, a large number of previous masks is not necessary to be processed for the final segmentation decision step. As shown in the experimental evaluation, background detection accuracy is increased while foreground detection is maintained to be complete enough through filtering of the preliminary binary masks, which is adapted according to the motion of the foreground.

The paper is organized as follows. Section 2 first overviews the system and then describes the employed robust global motion estimation and error generation approaches. Following, the segmentation algorithm that includes filtering, thresholding as well as a post-processing step is described in detail. The outcome of the proposed algorithm is evaluated on seven test sequences, and the results are presented in Section 3. Finally, Section 4 concludes the paper with discussion on perspectives on the subject.

2. Object segmentation based on bidirectional inter-frame change detection

The overview of the proposed algorithm is shown in Fig. 1. For the *n*th frame of a video sequence we employ two adjacent frames, one for each temporal direction. The luminance component contains the most important information for the scope of motion segmentation. The incorporation of the other chroma components, as illustrated in the example in Fig. 9, does not bring substantial improvement, and thus only the luminance component is taken into account. Since our approach deals with sequences with moving camera, the parametric model that describes global (i.e. mainly induced

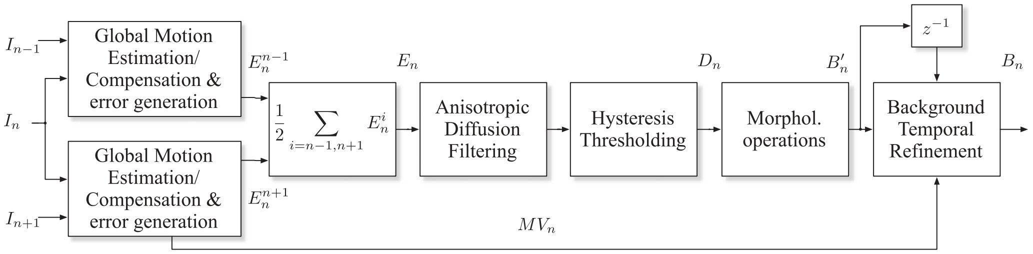
I_{n-1}
I_n
I_{n+1}
Global Motion Estimation/ Compensation & error generation
Global Motion Estimation/ Compensation & error generation
E_n^{n-1}
E_n^{n+1}
$\frac{1}{2}\sum_{i=n-1,n+1} E_n^i$
E_n
Anisotropic Diffusion Filtering
Hysteresis Thresholding
D_n
Morphol. operations
B'_n
z^{-1}
Background Temporal Refinement
B_n
MV_n

Fig. 1. Proposed system overview.

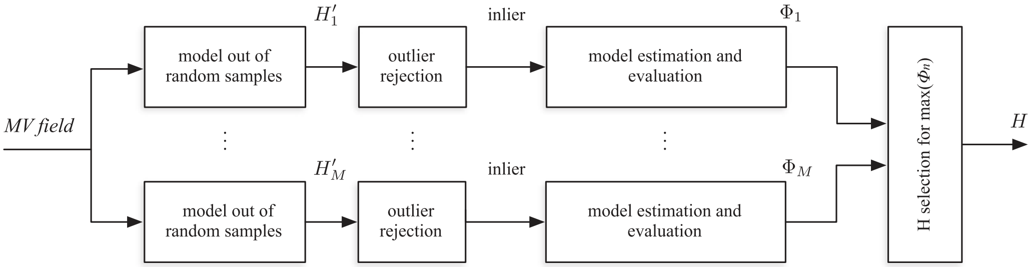
MV field
H_1'
model out of random samples
outlier rejection
inlier
model estimation and evaluation
Φ_1
H_M'
model out of random samples
outlier rejection
inlier
model estimation and evaluation
Φ_M
H selection for max(Φ_n)
H

Fig. 2. Global motion estimation algorithm using the Helmholtz Tradeoff Estimator and two motion models.

by camera) motion between two given video frames in each temporal direction (Global Motion Estimation, GME) has to be estimated first. Following, their global motion is compensated (Global Motion Compensation, GMC) and eventually the error maps E_n^{n-1} and E_n^{n+1} are obtained. Global motion is also compensated between the corresponding motion vector fields and the resulting information is employed in the post-processing step for improving temporal consistency. The obtained error energy maps are fused using averaging, resulting in E_n. While error locations indicate moving objects' boundaries in real scenes, exploiting directly the error frames for extracting boundary information of moving objects would suffer from great deal of noise even in the ideal case of perfectly compensated global motion. This is due to the fact that random noise created in one frame is different from the one created in successive frames [7], and thus results in slight changes of the error locations (i.e. potentially moving objects) in successive frames. Therefore, the error frame E_n is filtered, and subsequently a thresholding segmentation scheme, encompassing spatial localization of the error energy, is applied. In the obtained preliminary binary image $B_n^{'}$ every pixel is labeled as either foreground or background. Finally, the Background Temporal Refinement step reinforces spatiotemporal consistency, resulting in the final segmentation mask B_n.

The assumptions under which the proposed algorithm performs well as well as limitations and strong points are discussed below:

- The camera viewpoint is assumed to be fixed, for a valid representation of background motion by the parametric motion model involved in GME.
- There is no limitation in the number of objects in the scene that can be detected. Nevertheless, when objects are very close to each other they tend to be classified as one combined object.
- Regarding foreground object size the objects are not detected if they are smaller than approximately 10% of the image frame due to morphological processing. Additionally, if the object is larger than approximately 80% of the frame the Global Motion Estimation reflects inaccurately real camera motion and thus segmentation performance decreases dramatically.
- There is no background modeling involved and the algorithm does not need any training stage for parameter setting.
- No a priori information is assumed on the shape and texture of objects. In cases of lightly textured, low colored sequences, static camera or newly appearing objects; the algorithm performs robustly as long as there is apparent motion differentiation between foreground and background.

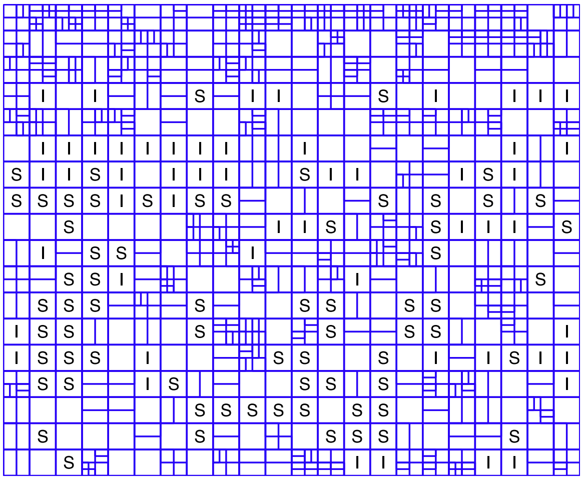

Fig. 3. The block motion vector field for frame 100 of the *Allstars* sequence encoded with the KTA reference software (QP 18), SKIP blocks (S) and INTRA blocks (I) are omitted from the global motion estimation process.

2.1. Global motion compensation

The employed global motion estimation algorithm, proposed by Tok et al. in [16] is a Monte-Carlo based method using the Helmholtz Tradeoff Estimator and is overviewed in Fig. 2. The algorithm derives background motion models from a set of local translational motion models such as motion vectors of encoded video streams. An example for such motion vector fields is shown in Fig. 3. Misestimated motion vectors and the ones belonging to foreground objects are removed by applying the Helmholtz principle. Thus, the global motion estimation algorithm can estimate parametric models from motion vector sets that have up to $\varepsilon = 80\%$ of outliers. In this section, frame indices are omitted for brevity.

For a pair of video frames, the algorithm generates preliminary motion models $\mathbf{H}'$ from randomly selected motion vectors and evaluates how well such a model fits the whole set of all K vectors. This step is repeated M times. In each iteration step $\nu \in \{1, \dots, M\}$, two (uniformly)

rotation	zoom	translation
$\begin{bmatrix}\cos(\alpha) & \sin(\alpha) & 0\\ -\sin(\alpha) & \cos(\alpha) & 0\\ 0 & 0 & 1\end{bmatrix}$	$\begin{bmatrix}z & 0 & 0\\ 0 & z & 0\\ 0 & 0 & 1\end{bmatrix}$	$\begin{bmatrix}1 & 0 & t_x\\ 0 & 1 & t_y\\ 0 & 0 & 1\end{bmatrix}$

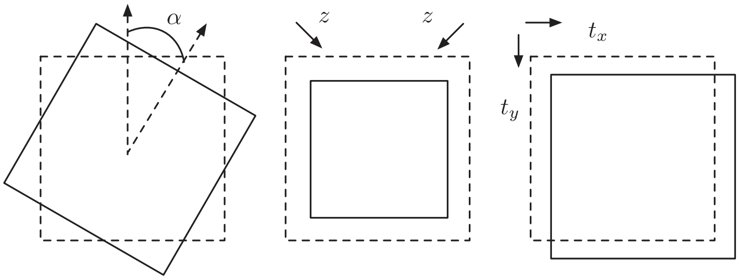

Fig. 4. Transformation matrices **A** for rotation, zoom and translation that transform a position $\mathbf{p}=(x,y,1)^T$ to a new position $\mathbf{p}'=(x',y',1)^T$ by $\mathbf{p}'=\mathbf{A}\cdot\mathbf{p}$.

randomly selected vectors are taken from the motion vector field to derive a preliminary four parameter model:

$$\mathbf{H}'_\nu=\begin{pmatrix} m'_{0,\nu} & m'_{1,\nu} & m'_{2,\nu}\\ -m'_{1,\nu} & m'_{0,\nu} & m'_{3,\nu}\\ 0 & 0 & 1\end{pmatrix} \quad (1)$$

to roughly describe the translational, rotational and zoom deformation (Fig. 4) between two frames induced by camera motion. For each vector of the whole set a fitting error related to the model $\mathbf{H}'_\nu$ is calculated. Following [17], the $(1-\varepsilon)$th percentile λ_ν is then taken to estimate an error standard deviation:

$$\sigma_\nu=1.4826\cdot\left(1+\frac{5}{K-p}\right)\cdot\lambda_\nu, \quad (2)$$

where p is the amount of observations (motion vector components, $\mathbf{MV}_X$ and $\mathbf{MV}_Y$) needed to describe a model $\mathbf{H}'_\nu$.

A new subset Θ_ν of all vectors that fit the motion defined by $\mathbf{H}'_\nu$ with an error smaller than $5/2\sigma_\nu$ is defined [17]. This subset is rated by its standard deviation $\sigma_{\Theta,\nu}$ and size $I_{\Theta,\nu}$:

$$\Phi_\nu=\frac{I_{\Theta,\nu}}{\sigma_{\Theta,\nu}}. \quad (3)$$

Finally the subset Θ_ν with the highest rating Φ_ν is taken to derive a perspective eight parameter model:

$$\mathbf{H}=\begin{pmatrix} m_0 & m_1 & m_2\\ m_3 & m_4 & m_5\\ m_6 & m_7 & 1\end{pmatrix} \quad (4)$$

using Least Squares regression. This model can describe more complex deformations between two video frames, such as translation, rotation, zoom and perspective deformation.

The probability P for selecting two vectors to derive a preliminary model H'_ν with $p=4$ parameters and an expected outlier percentage of ε is

$$P=1-(1-(1-\varepsilon)^p)^M. \quad (5)$$

Thus, the iteration count M can easily be estimated as

$$M=\frac{\log(1-P)}{\log(1-(1-\varepsilon)^p)}. \quad (6)$$

In this paper, P has been set to 99.5% and ε has been set to 70% to ensure accurate estimation of the background motion.

2.2. Bidirectional change detection

For the nth frame of the video sequence, let $\overline{I_n^{n-1}}$ and $\overline{I_n^{n+1}}$ be the estimations of I_n based on the corresponding eight-parameter global motion models as in (4) between I_{n-1} and I_{n+1} respectively. Based on these, as depicted in Fig. 1, the global motion compensated error frames for the two temporal directions are $E_n^{n-1}=|I_n-\overline{I_n^{n-1}}|$ and $E_n^{n+1}=|I_n-\overline{I_n^{n+1}}|$.

As discussed in Section 1.1, many inter-frame change detection algorithms in the literature focus on motion information of one temporal direction i.e. I_{n-1} or I_{n+1}. In this way, only edges of one motion direction are included in the foreground region. To overcome this issue, Kameda and Minoh proposed to use error frames from the preceding and succeeding frames. In [10], they perform thresholding on the global motion compensated errors of each direction E_n^{n-1} and E_n^{n+1} separately and then obtain a "double-difference image" by a logical intersect operation between the resulting binary masks B_n^{n-1} and B_n^{n+1}. This concept is also adopted by [11,12]. The intersect operation ensures that foreground misclassifications are drastically reduced (resulting in high Precision, as shown in Section 3) in the obtained B_n mask, but at the same time a significant amount of foreground regions are misclassified (resulting in low Recall rates).

This shortcoming affects the overall segmentation quality in a bad manner as we show experimentally in Section 3. In this paper, we overcome this issue by including information from both directions in an accumulative manner, instead of employing the intersect operation. E_n^{n-1} and E_n^{n+1} are combined as

$$E_n=\frac{E_n^{n-1}+E_n^{n+1}}{2} \quad (7)$$

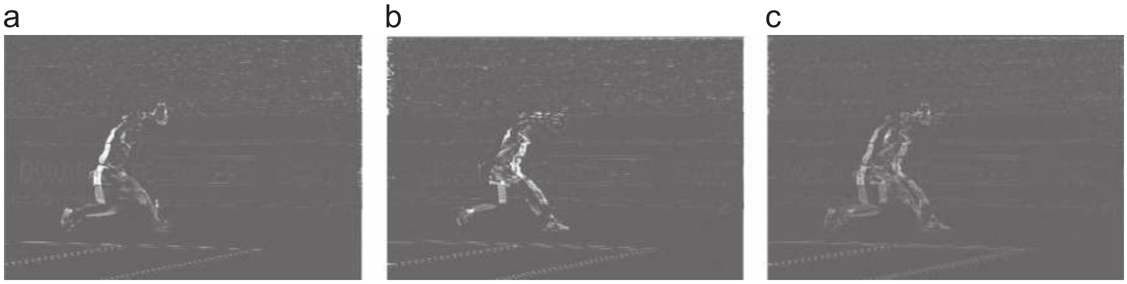

Fig. 5. *Stefan* sequence, example error frames. In (a) and (b) the error energy is located mostly on the left and right side of the foreground object, respectively, while in (c) error location indicates the foreground location better.

and the thresholding segmentation algorithm is then applied on E_n. By fusing the information of these two error frames, a more complete foreground detection is achieved, which should be reflected in higher *Recall* rates in the evaluation. This is due to approaching each frame "bidirectionally" as illustrated in Fig. 5. Additionally, accurate global motion estimation enables elimination of high error energy in the background region and consequently high *Precision* rates are achieved. Precision and Recall metrics are discussed in Section 3.

2.3. Thresholding using hysteresis

The advantages of segmentation algorithms based on inter-frame change detection are that they are straightforward to implement and enable automatic detection of new appearing objects. Their drawbacks include noise (small misclassified regions) and irregular object boundaries [7]. Thus, the error maps should be filtered prior to thresholding and morphological operations such as opening and closing might be incorporated after thresholding to alleviate noise. Here, we employ an enhancement of the algorithm proposed in [14] that encompasses anisotropic diffusion filtering, weighted mean thresholding and morphological processing. The enhancement concerns the thresholding step and will be elaborated in the following.

In the filtering stage, anisotropic diffusion filtering [18] is employed. Anisotropic diffusion offers a non-linear and space-variant filtering of the error frame, that while having a low pass character preserves the edges of the image. In this way it serves the reduction of high frequency noise due to misestimations in the background while enhancing edges. The filter has been set up to perform 20 iterations. The second conduction function defined in [18] has been used that privileges wide regions over smaller ones, and in line with the authors' suggestions, the local contrast *kappa* has been set to the 80% value of the integral histogram of the global motion compensated error image. At the final stage of morphological processing, small holes of the background are removed and holes inside foreground objects are closed in succession for refinement of the binary segmentation mask.

The weighted mean thresholding is given by

$$T(w) = w \cdot \max(E'_n) + (1-w)\mu \tag{8}$$

where w is a constant and μ is the mean of the normalized filtered error frame $E_n^{'}$ (E_n is normalized by its maximum). The weighted mean thresholding in (8) is adapted according to the intensity histogram of every frame, but does not take into account the error localization. In the global motion compensated error frame, e.g. as depicted in Fig. 5(c), there are significant error values in the foreground area and errors resulting from misestimations in the background area. To eliminate these missestimations, we enhance the weighted mean thresholding approach, as follows.

At first stage, pixels assigned with high error energy are labeled as foreground (F_0 region). An example is illustrated in Fig. 6(a). Following, pixels with lower error energy, that are spatially connected with F_0, are favored against the ones not connected with F_0, even when the latter have high error energy. Thus, we employ two *hysteresis* thresholds [15,19]. We begin by applying a low threshold $T(w_{\text{low}})$ using (8). This results in high amount of falsely detected foreground pixels, but we can be fairly sure that most regions of the foreground are correctly classified. We then apply a higher threshold $T(w_{\text{high}})$ only on regions that are connected with the binary result from $T(w_{\text{low}})$. Once this process is complete we have a binary mask where each pixel is marked as either foreground or background.

Eventually, the obtained segmentation mask $B_n^{'}$ is given by

$$B_n^{'} = k(D_{n,(w_{low},w_{high})}) \tag{9}$$

where

$$D_{n,(w_{low},w_{high})} = \theta(\phi * E_n^{'}) \tag{10}$$

$E_n^{'}$ is the normalized filtered error frame, ϕ denotes anisotropic diffusion filtering, θ weighted mean thresholding using hysteresis and k morphological processing. Following, frame indices are omitted for brevity.

2.4. Optimal weight selection

One main issue that affects the robustness of the weighted mean algorithm is the appropriate selection of the weight parameter w involved in (8). In [14], the authors proposed $w=0.1$ heuristically. By adopting hysteresis thresholding for sake of increasing accuracy, we have one more degree of freedom, due to the fact that we have to search for two optimal thresholding parameters i.e. their corresponding weights w_{low} and w_{high}.

Finding the optimal generic solution for hysteresis thresholding is considered to be a challenging issue [20,21], mainly due to the strong dependency of the optimal solution on the

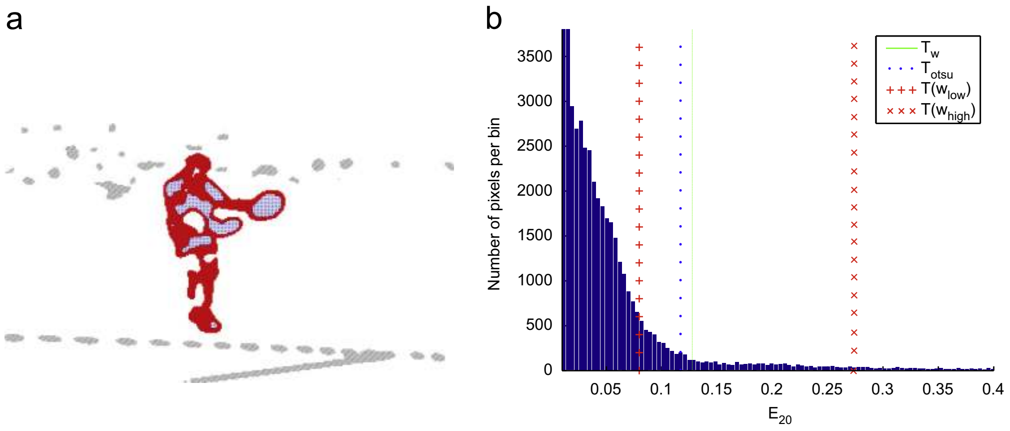

Fig. 6. *Stefan* sequence, thresholding examples for frame 20. (a) Segmentation initial classes using hysteresis thresholding. Pixels with $E'_{20}(x,y) > T(w_{\text{high}})$ are depicted in solid red (class F_0). Pixels with $E'_{20}(x,y) > T(w_{\text{low}})$ that are connected with class F_0 are depicted in dotted blue and with dashed gray are the discarded pixels for which $E'_{20}(x,y) > T(w_{\text{low}})$ and are not connected with the ones in class F_0. (b) The weighted mean (T_{w}), Otsu (T_{otsu}) and hysteresis weighted mean thresholds ($Tw_{\text{low}}, Tw_{\text{high}}$) are depicted on the intensity histogram of the normalized error.

Table 1
Basic steps for the optimal weight selection.

Step	Outcome
(1) Segment the image using the set of weights $\mathbf{W} = \{W_j\}, j = 1, \ldots, L$	D_j
(2) Estimate Ground Truth	*EGT*
(3) Threshold EGT using threshold $i = 1, \ldots, L$	PGT_i
(4) Compare PGT_i and D_j to find the optimal $PGT_{i=k}$ using χ^2 test	PGT_k
(5) Find ζ for the optimal segmentation mask D_j using χ^2 test	$D\zeta$

input image. A survey on this topic is presented in [22]. The method of Yitzhaky and Peli [21] is to the best of our knowledge the method that selects the optimal pair of hysteresis thresholds from a set of possible values. It is not a parametric approach and it eliminates manual determination to the parameter set selection. The algorithm performs statistical analysis on detection results produced by different parameters to create an Estimated Ground Truth (EGT) and finds the optimal pair of parameters for edge detection on images. We employ this algorithm to find the optimal weights for weighting mean thresholding using hysteresis on the global motion compensated error maps. As suggested in [21], the obtained optimal parameter set is appropriate for similar images, thus we find the optimal weight set of the first frame of a video sequence, and employ this for the rest of the frames. The range of parameters to be tested here is 28 (weight values range from 0.005 to 0.4 in steps of 0.05) which appears to be reasonable since it covers a wide range of detection results from noisy to sparse. The procedure is described in the following and is overviewed in Table 1. Given a set of L possible weight combinations:

$$\mathbf{W} = \{W_j = (w_{low}, w_{high})_j | w_{low}, w_{high} \in [0,1] \text{ and } w_{low} < w_{high}\} \tag{11}$$

where $j = 1, \ldots, L$ use the segmentation masks $\mathbf{D} = \{D_1, D_2, \ldots, D_L\}$ derived using (10) that correspond to these combinations, to construct the Estimated Ground Truth (EGT): a pixel location which is identified as foreground in all segmentation masks, will be assigned the highest level in the EGT, while a location identified as foreground only in one segmentation mask will be assigned the lowest level. Thus, the EGT is constructed having values within $[1, L]$. An EGT example is shown in Fig. 7.

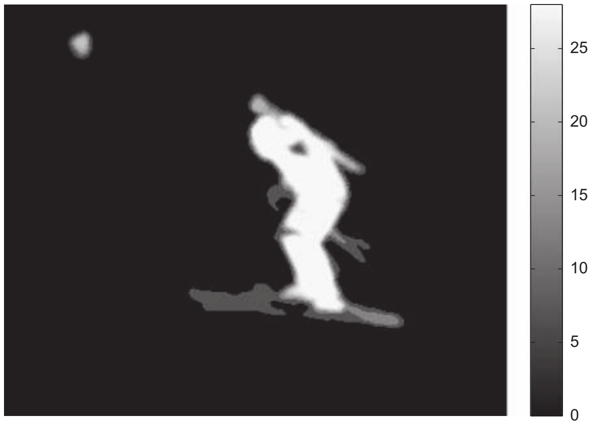

Fig. 7. Estimated Ground Truth – the first processed frame of the *Biathlon* sequence, L=28.

The EGT is then thresholded with each threshold level i in the set $I = \{1, \ldots, L\}$ forming the Potential Ground Truth (PGT_i) for the corresponding level i. Following, each PGT_i mask is compared to each D_j segmentation mask, where $j = 1, \ldots, L$ corresponds to each weight combination $(w_{low}, w_{high}) \in \mathbf{W}$ and generate four probabilities for each individual match:

$$\overline{TP}_{PGT_i} = \frac{1}{N}\sum_{j=1}^{N} TP_{PGT_i,D_j}$$
$$\overline{TN}_{PGT_i} = \frac{1}{N}\sum_{j=1}^{N} TN_{PGT_i,D_j}$$
$$\overline{FP}_{PGT_i} = \frac{1}{N}\sum_{j=1}^{N} FP_{PGT_i,D_j}$$
$$\overline{FN}_{PGT_i} = \frac{1}{N}\sum_{j=1}^{N} FN_{PGT_i,D_j}. \tag{12}$$

Now, if each PGT_i is regarded as ground truth, the above statistical terms are defined as *True Positives* (TP): correctly classified as foreground pixels, *True Negatives* (TN): correctly classified as background pixels, *False Positives* (FP, also known as *Type I error*): falsely classified as foreground pixels and *False Negatives* (FN, or *Type II error*): falsely classified as background pixels.

The best PGT_i mask is the one that yields the best match according to the *Chi-square test* metric. The Chi-square test of the optimal weight set [21] is

$$\overline{\chi}^2_{PGT_i} = \frac{\overline{sn}_{PGT_i} - Q_{PGT_i}}{1-Q_{PGT_i}} \cdot \frac{\overline{sp}_{PGT_i} - (1-Q_{PGT_i})}{Q_{PGT_i}} \tag{13}$$

where

$$Q_{PGT_i} = \overline{TP}_{PGT_i} + \overline{FP}_{PGT_i} \tag{14}$$

$$\overline{sn}_{PGT_i} = \frac{\overline{TP}_{PGT_i}}{P} \tag{15}$$

$$\overline{sp}_{PGT_i} = \frac{\overline{FP}_{PGT_i}}{1-P} \tag{16}$$

$\overline{sn}_{PGT_i}$ is the *sensitivity* or True Positive Rate (TPR), and $\overline{sp}_{PGT_i}$ is the *specificity* which is equivalent to 1-FPR (False Positive Rate). *Prevalence P* is the average relative number of positive detections. A higher $\overline{\chi}^2_{PGT_i}$ indicates a better parameter set selection. Fig. 8 demonstrates an example of the values of the Chi-square measure for different weight levels. The best match between PGT_i and the EGT is given for $k = \arg\max_i \overline{\chi}^2_{PGT_i}$, thus obtaining the optimal potential ground truth PGT_k. Based on this, the following Chi-square is calculated:

$$\chi^2(D_j) = \frac{sn_{PGT_k,D_j} - Q_{PGT_k,D_j}}{1-Q_{PGT_k,D_j}} \cdot \frac{sp_{PGT_k,D_j} - (1-Q_{PGT_k,D_j})}{Q_{PGT_k,D_j}} \tag{17}$$

where

$$Q_{PGT_k,D_j} = TP_{PGT_k,D_j} + FP_{PGT_k,D_j} \tag{18}$$

$$sn_{PGT_k,D_j} = \frac{TP_{PGT_k,D_j}}{TP_{PGT_k,D_j} + FN_{PGT_k,D_j}} \tag{19}$$

$$sp_{PGT_k,D_j} = \frac{FP_{PGT_k,D_j}}{FP_{PGT_k,D_j} + TN_{PGT_k,D_j}} \tag{20}$$

and finally the segmentation mask D_ζ for $\zeta = \arg\max_j \chi^2(D_j)$ yields the optimal segmentation mask.

2.5. Spatiotemporal consistency

The obtained segmentation mask ($B_n^{'}$) usually suffers from misclassifications, i.e. falsely classified foreground pixels (False Positives) or falsely classified background pixels (False Negatives), caused by various sources. In this section, we identify the circumstances under which such misclassifications occur and then propose a strategy to address them. Misclassifications can occur when

- the sequence contains background noise (e.g. spectators' movement in sports sequences) → mainly causes FP;
- motion vectors are not describing real motion (e.g. when generated to optimize the rate-distortion trade-off) → can cause both FP and FN;

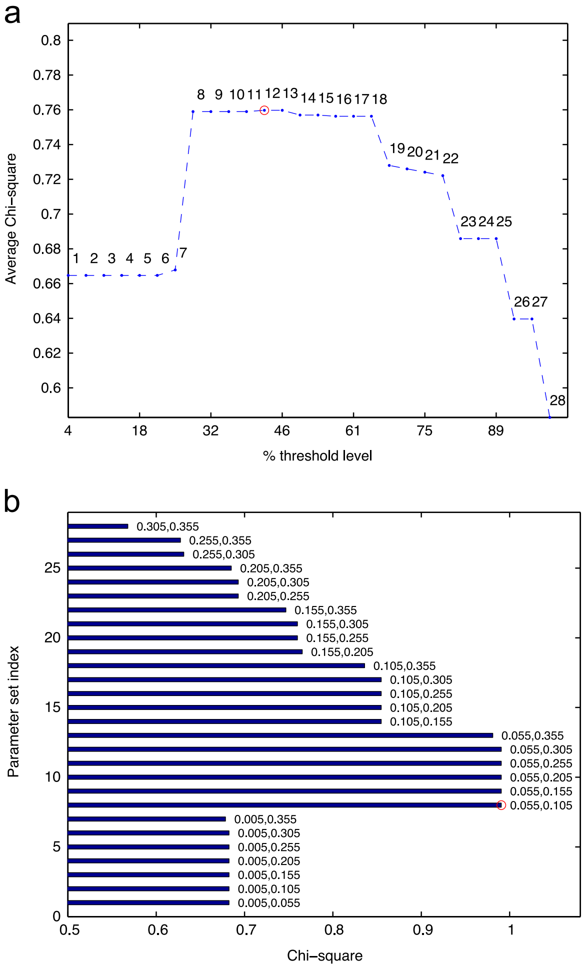

Fig. 8. Chi-square metric for finding the optimal weight pair for weighted mean thresholding for *Biathlon* sequence. (a) Average chi-square ($\overline{\chi}^2_{PGT_i}$) for every threshold level shows a maximum at level $k = 12$. (b) Chi-square ($\chi^2(D_j)$) between the different detections and the *EGT* shows a maximum for the weight set $(0.055, 0.105)$.

- the motion model in Eq. (4) is unable to describe accurately the undergoing camera motion → can cause both FP and FN;
- the motion of the foreground object (or part of it) matches the dominant motion of the video frame and the relative velocity (between foreground and background) is almost zero → can cause FN;
- very high foreground velocity occurs, i.e. large displacement between adjacent frames. This effect is known as “*ghosting effect*” and characterizes situations where the object seems to appear twice [23]. It is present in cases of inter frame change detection due to the lack of background modeling → can cause FN.

Additionally, one effect that can deteriorate the segmentation result is the temporal coherence of the estimated sequence of segmentation masks. Non-smooth changes between consecutive frames might cause bad effects, such as *flickering*. The hysteresis scheme can handle some of the above mentioned error cases to certain

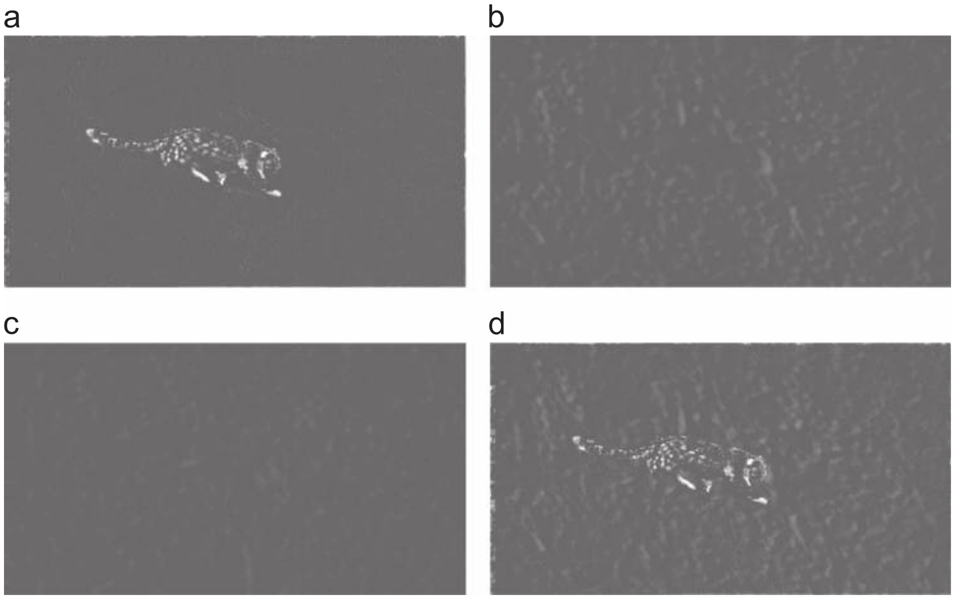

Fig. 9. Example of global motion compensated error frames for luminance and chrominance components as well as combination of them for the *mountain sequence*. (a) Y component, (b) U component, (c) V component and (d) Combination of YUV.

extent (e.g. Fig. 6(a)), due to the fact that it favours object boundaries' connectivity. In order to deal with the above described misclassifications and temporal inconsistencies, we propose the following strategy (Background Temporal Refinement, BTR). First, the obtained preliminary binary masks B'_{n-1} and B'_n are filtered with a two-dimensional isotropic Gaussian lowpass filter with standard deviation that is adapted to every frame according to the average magnitude of the motion vector subset of the current frame that corresponds to the foreground region of the previous frame. Next, the (grayscale) mask that is the Hadamard product (pairwise multiplication) of the filtered versions of the preliminary masks is binarized using Otsu thresholding [24] to produce the final segmentation mask B_n. The multiplication of the filtered preliminary masks serves the elimination of temporal inconsistencies that are observed, when every binary mask is produced independently of its adjacent ones. Error propagation is not an issue here, since B'_{n-1} and B'_n are created independently up to this point.

In more detail, filtering serves in creating a spatial attenuation of the object boundaries so that when the filtered masks are combined, and depending on the foreground object's velocity, the new parts of the foreground in B'_n that do not exist in B'_{n-1} are maintained. Especially in cases of fast moving objects, filtering helps towards a more complete object detection in the final mask. Filtering is adapted as described in the following:

$\mathbf{H_n}$ is the estimated eight-parameter model for the nth frame of the video sequence, as in Eq. (4). The corresponding global motion compensated vector field is calculated as

$$\mathbf{MV}^{\mathrm{GMC}}(x,y,n) = \mathbf{MV}(x,y,n) - \mathbf{MV}(x,y;\mathbf{H}_n) \tag{21}$$

where $\mathbf{MV}(x,y,n)$ is the motion vector field and $\mathbf{MV}(x,y;\mathbf{H}_n)$ is the motion vector field that represents the estimated global motion. $\mathbf{MV}^{\mathrm{GMC}}(x,y,n)$ and B'_{n-1} are used to calculate an adaptive isotropic Gaussian filter. Let $\mathbf{\Omega}$ be the region that B'_{n-1} defines and corresponds to N motion vectors. The preliminary binary mask B'_n is then convolved with Gaussian filter with kernel size $(\phi_n \times \phi_n)$, where $\phi_n = \lceil 4 \cdot \sigma_n + 1 \rceil$ and

$$\sigma_n = \frac{1}{N} \sum_{i=1}^{N} \sqrt{(\mathbf{MV}_{Xi}^{\mathrm{GMC}})^2 + (\mathbf{MV}_{Yi}^{\mathrm{GMC}})^2} \tag{22}$$

standard deviation. $i \in \mathbf{\Omega}$ and $\mathbf{MV}_{Xi}^{\mathrm{GMC}}$, $\mathbf{MV}_{Yi}^{\mathrm{GMC}}$ are the motion vector components for X and Y directions respectively at frame n.

3. Experimental evaluation

Seven test sequences (1570 frames, common intermediate format up to standard definition resolution), which are described in Table 2 are considered for experimental evaluation. In order to objectively evaluate the performance of the proposed algorithm we employ manually created moving objects ground-truth segmentation sequences. The segmentation accuracy is measured in terms of Precision (P), Recall (R) and F-measure (F), that are respectively defined as

$$P = \frac{TP}{TP+FP} \tag{23}$$

$$R = \frac{TP}{TP+FN} \tag{24}$$

$$F = 2\left(\frac{P \cdot R}{P+R}\right) \tag{25}$$

where TP, FP and FN are defined as described in Section 2.4 in the case of comparing the calculated segmentation mask to manually created ground truth. Precision indicates

Table 2
Dataset description.

Sequence	Resolution	Frames	Objects	Camera movement, foreground object & texture description
Allstars	352 × 288	250	Up to eight	Slow pan and tilt, small objects, lightly textured background
Biathlon	352 × 288	200	One	Fast pan and slow zoom, medium sized object, lightly textured background
Mountain	352 × 192	100	One	Pan, tilt and zoom, highly textured background, medium object size
Race	544 × 336	100	Three	Fast pan, moderately textured background, variations in object sizes
Stefan	352 × 240	300	Up to two	Fast pan and zoom, one large object and presence of a much smaller one in several frames (ball), moderately textured background
BBC fish	720 × 576	120	One	Pan, tilt and zoom, medium object size, lightly textured background
Horse	352 × 288	120	One	Fast pan, fast tilt and zoom, one large object on highly textured background

Table 3
Test sequences and results of experimental evaluation in terms of Average Precision (P), Recall (R) and F-measure (F) of reference and proposed algorithms. The best Precision, Recall and F-measure results are shown in bold.

Sequence	Algorithm 1 [9]			Algorithm 2 [10]			Algorithm 3		
	P	R	F	P	R	F	P	R	F
Allstars	0.44	**0.69**	0.52	**0.84**	0.49	0.61	0.77	0.59	**0.65**
Biathlon	0.24	**0.83**	0.36	**0.92**	0.63	0.74	0.78	0.87	**0.82**
Mountain	0.60	**0.95**	0.73	**0.93**	0.59	0.72	0.84	0.85	**0.84**
Race	0.69	**0.84**	0.75	**0.89**	0.41	0.53	0.74	0.83	**0.78**
Stefan	0.65	**0.80**	0.69	**0.86**	0.41	0.52	0.71	0.79	**0.73**
BBC fish	0.75	**0.87**	0.80	**0.89**	0.38	0.53	0.82	0.83	**0.81**
Horse	0.65	**0.78**	0.70	**0.96**	0.24	0.38	0.88	0.70	**0.78**
Average (%)	57.4	**82.2**	64.9	**89.7**	45.0	57.3	79.0	78.1	**77.4**

how exact the segmentation is, meaning how accurately the background is estimated, whereas Recall shows how complete the foreground segmentation is. Balancing between these two contradictory quantities, Precision and Recall, comprises the main challenge that algorithms dealing with the task of object segmentation must address. F-measure is the harmonic mean of Precision and Recall and is widely used as an objective overall indication of the segmentation quality.

3.1. Algorithm scenarios

In order to compare global motion compensated error fusion approaches for object segmentation in sequences with moving camera, we compare the proposed algorithm, *Algorithm* 3, which is detailed described in Section 2 to the following GME error fusion approaches: *Algorithm* 1 proposed in [9] which uses one adjacent frame for the detection of object segmentation mask that is also used to predict and reject outliers for GME and *Algorithm* 2 proposed in [10] that employs the intersection fusion scheme as described in Section 2.2. The global motion estimation algorithm described in Section 2.1 is used for the error fusion scheme of *Algorithm* 2, and the segmentation of global motion error frames as described in Sections 2.3–2.5 are used in each case in order to have a fair comparison of segmentation performance.

Algorithm 1 produces segmentation masks that suffer from background misclassifications as well as incomplete foreground detection, especially in one side of the foreground object, due to the fact that one motion direction is used for global motion compensation. This results in low Precision, but fairly good Recall rates. Algorithm 2 presents enhanced background detection accuracy, since the intersect operation ensures that most of the background misclassifications are avoided, but the segmentation masks suffer from incomplete foreground detection, as described in Section 2.2. This is reflected by high Precision but very low Recall rates. Algorithm 3 enables complete foreground detection due to the proposed error fusion scheme and at the same time produces 33.1% on average more accurate background detection (in terms of Recall) compared Algorithm 2 on the whole test dataset.

Fig. 12 illustrates examples of the above cases and Table 3 shows Precision, Recall and F-measure for the algorithm scenarios described above. Algorithm 2 performs on average 32.4% better than Algorithm 1 in terms of Precision, but suffers from 37.3% lower Recall rates. The best performance in terms of Precision is achieved by Algorithm 2 and the best one in terms of Recall is achieved by Algorithm 1. Nevertheless, Precision and Recall are two contradictory quantities; often increment of each one of them means decrement of the other one. Thus, by achieving good but not the best Precision and Recall rates, but still above at least 59%, our proposed algorithm outperforms the reference algorithms and clearly improves the results in terms of F-Measure. Fig. 10 presents a comparative overview of the percentage of frames in each test sequence that have quality above 75% in terms of F-measure. Figs. 13 and 14 illustrate examples of the test dataset, as well as F-measure curves, using the reference and the proposed algorithms.

By incorporating BTR, the segmentation masks are temporally more consistent, false positives are eliminated, while foreground object detection is more complete. The "ghosting effect", which appears when foreground objects are moving fast, is also eliminated due to the filter adaptation according to the foreground object's velocity, and the object boundaries are smoothed over time as can be seen e.g. in Fig. 12. Nevertheless, in the case of *Allstars*, one of the football players is repeatedly moving and stopping in front of a static object, and he is in some cases falsely regarded to belong to the background together with that static object. This results in a slight (1.6%) degradation in terms of *F*-measure, as can be seen in Table 4 which presents the performance improvement in terms of *F*-measure by incorporating BTR, compared to the case where no background refinement step is involved, which has been presented in [15].

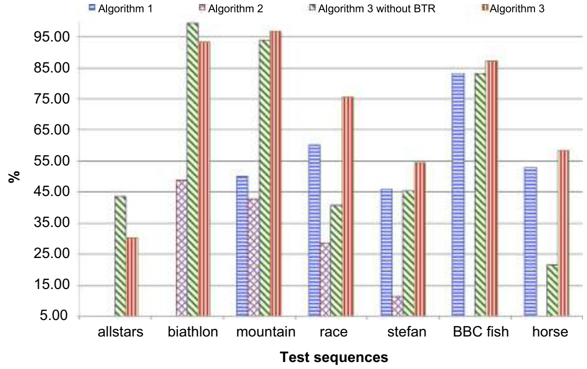

Fig. 10. Percentage of frames with quality above 75% in terms of *F*-measure. Comparison of reference and proposed algorithms.

Table 5 presents the evaluation of segmentation results that are produced using three thresholding schemes generated by Algorithm 3 for all the test sequences. The thresholding schemes compared are (i) the well known Otsu thresholding [24], which maximizes the ratio of inter/intra-class variance, (ii) the weighted mean thresholding (*WM*) [14] and (iii) the hysteresis weighted mean (*HWM*) as described in Section 2.3 (an example is illustrated in Fig. 6(b)). In every case, hysteresis mean thresholding outperforms the other two thresholding algorithms in terms of segmentation efficiency.

Additionally, the number of correctly detected objects is considered as quality measure. As described in Table 2, the test sequences contain foreground objects with various sizes that may move independently. As shown in Fig. 11, the proposed algorithm detects foreground objects with good accuracy. In the case of sequences with multiple objects presence, at least 79.13% of the foreground objects are detected with the proposed algorithm, 94.44% are detected with Algorithm 1 and 77.83% with Algorithm 2, whereas in sequences with single object presence (*Biathlon*, *Mountain*, *BBC fish* and *Horse*) the object is always correctly detected. As it is observed, Algorithm 1 shows higher detection rates than the proposed algorithm

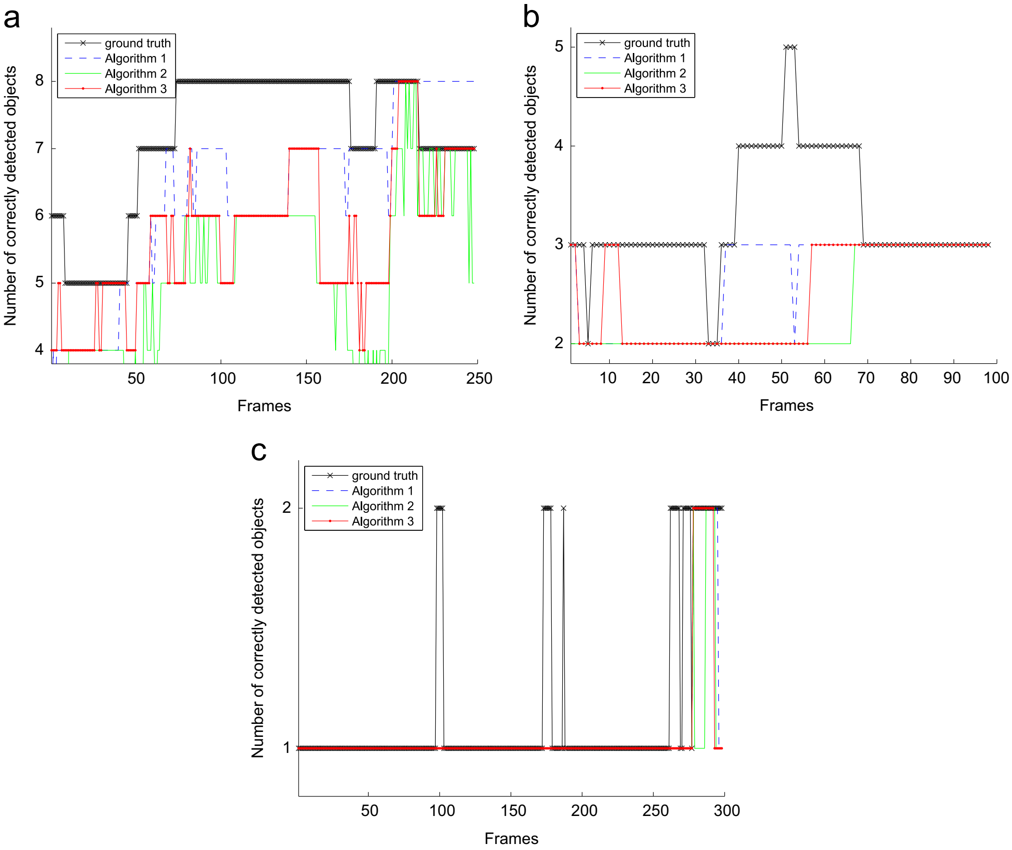

Fig. 11. Number of foreground objects detected with the proposed algorithm and reference algorithms in sequences with multiple objects. (a) *Allstars*, (b) *Race* and (c) *Stefan*.

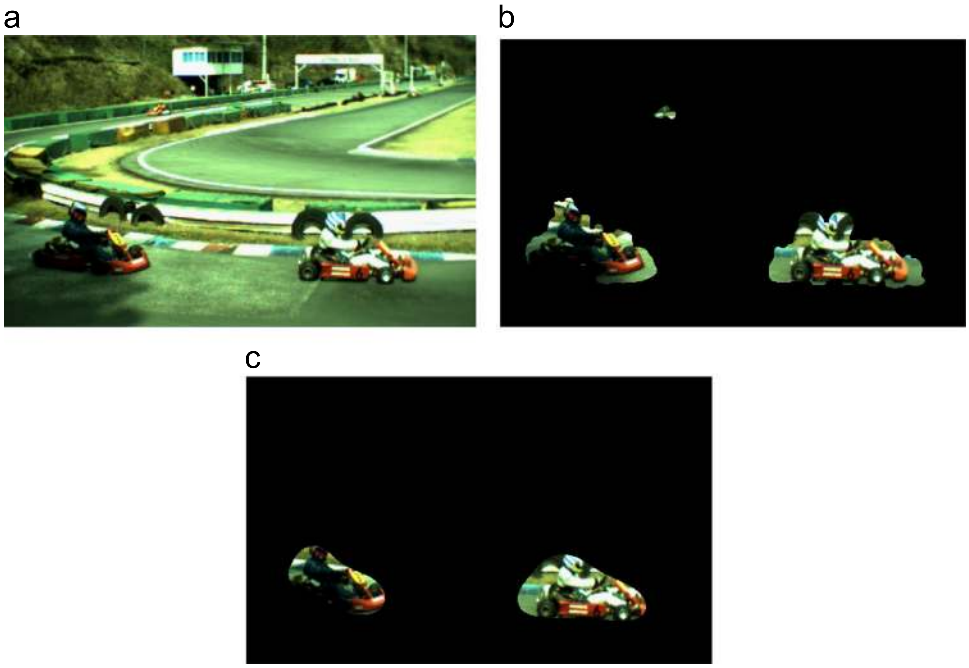

Fig. 12. Original frame and segmentation results of reference and proposed algorithms for *Race* sequence, frame 27. (a) Original frame, (b) Algorithm 3 without BTR and (c) Algorithm 3 with BTR.

(Algorithm 3), which is also in agreement with the higher recall rates in Table 3. However, these high detection rates are followed by high false foreground detection rates, which makes the performance of the proposed algorithm in general better compared to Algorithm 1. In more details, the average numbers of correctly detected objects in *Allstars* are 88.41%, 69.24% and 79.74%, in the case of algorithms 1, 2 and 3 (proposed) respectively, whereas in *Race* these rates are 81.27%, 73.70% and 78.53% and finally in *Stefan*, 95.17%, 93.50% and 94.67% respectively.

Regarding computational complexity, each part of the proposed algorithm is examined separately. For a frame with $m \times n$ pixels, bearing in mind that the number of iterations (motion vector outlier rejection M, anisotropic diffusion filtering, set of weights W, etc.) is fixed, and the involved parameters (motion model, gaussian kernel, etc.) have fixed size, the computational complexity of each part of the algorithm is $T_{GME}(m,n) = O(n \cdot m \cdot \log(n \cdot m))$, $T_{GMC}(m,n) = T_{error\ gen.}(m,n) = T_{filtering}(m,n) = T_{weight\ sel.}(m,n) = T_{thresholding}(m,n) = T_{morph.proc.}(m,n) = T_{BTR}(m,n) = O(n \cdot m)$. The $n \cdot m \log(m \cdot n)$ term in T_{GME} derives from the Helmholtz Tradeoff Estimator algorithm, where the fitting errors between all motion vectors and the preliminary motion model are calculated, for a maximum of $m/4 \times n/4$ blocks of size 4×4 (pixels). After this calculation, the set of errors has to be sorted in order to calculate the percentiles, and this sorting results in this term. Thus, in the worst case scenario, the complexity of the proposed algorithm is $O(n \cdot m \cdot \log(n \cdot m))$. Algorithm 2 involves the convergence rate, κ, of the gradient descent [9], which determines its complexity. Assuming that κ is not fixed, the computational complexity of Algorithm 2 is $O(n \cdot m \cdot \kappa)$, whereas in case Algorithm 3 the complexity is the same as the proposed one, since the fact that most of the included steps have to be performed twice, does not change O.

Regarding runtime, the proposed algorithm needs 1.6 s on average for a frame of a CIF sequence (*Biathlon*) under a 2.2 GHz AMD opteron 8354 with 48 GB RAM. From this time, 0.97 s are used for GME, 0.11 s for GMC, 0.38 s for filtering, 0.05 s for binarization, 0.01 s for morphological processing and 0.12 s for BTR. More concisely, 1.1 s is needed for GME/C and 0.5 s for segmentation which is implemented in MatLab. The algorithm of Kameda and Minoh [10] is not faster than the proposed algorithm, since all the steps have to be performed twice, for each direction, before combining the segmentation masks using the intersect operation. The algorithm [9] can save 75% of time in the GME step, based on the code provided as an executable by the authors. Comparing to the segmentation performance, the proposed algorithm outperforms [10] for 20% and [9] 12% in terms of *F*-measure. The GME algorithm that we employ here is based on the Helmholtz Tradeoff Estimator which ensures robustness against noise. This is reflected to the fact that the proposed algorithm outperforms Algorithm 1 in terms of segmentation efficiency and the possible employment of a faster GME approach would enable real-time application scenarios.

3.2. Segmentation of H.264/AVC compressed video data

In many application scenarios cameras are equipped with encoding capabilities and the reference video is not available at the decoder side for processing and extraction of content information. We test our approach with video streams from the state-of-the-art video coding standard H.264/AVC as depicted in Fig. 15, where the input is the

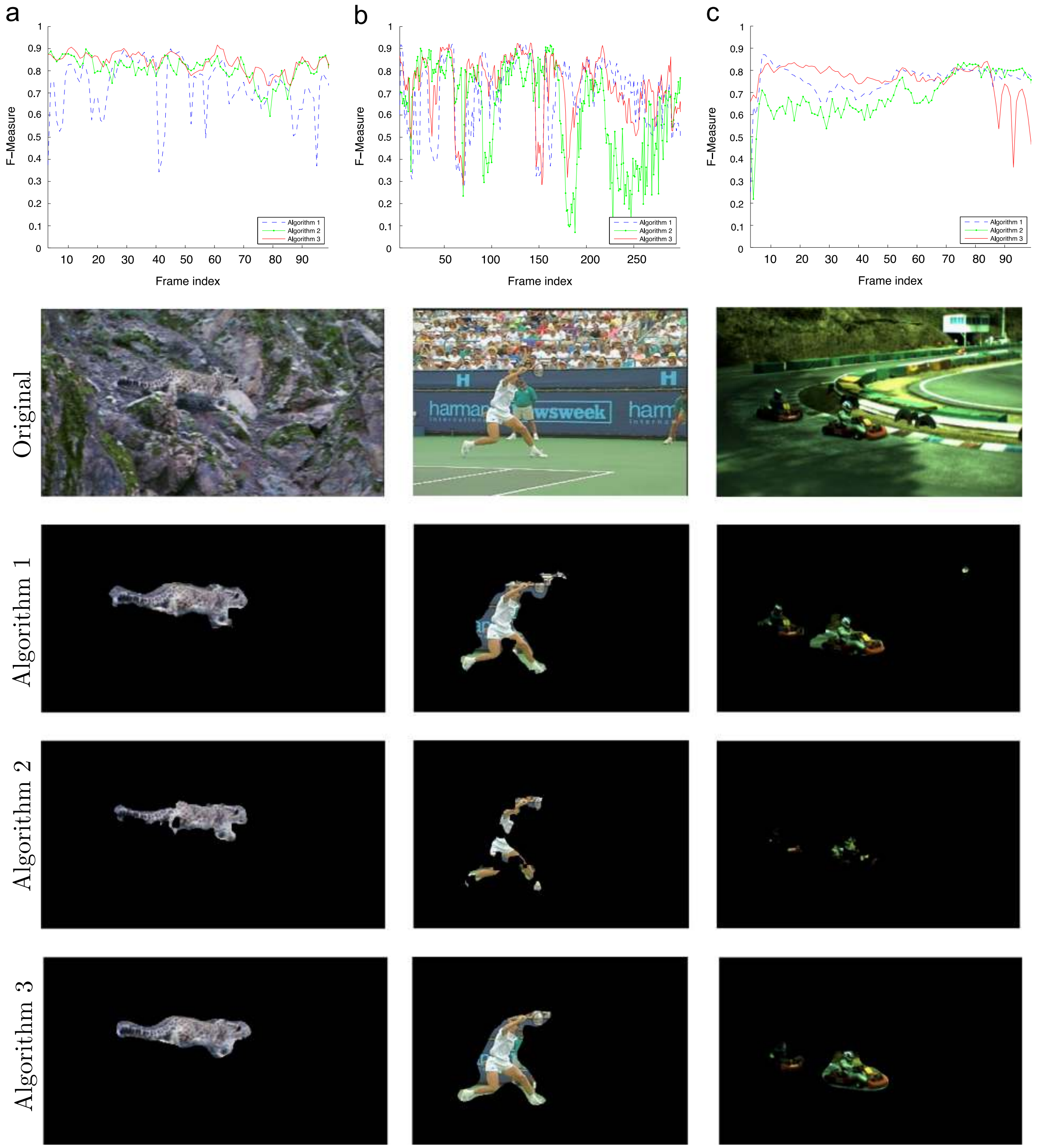

Fig. 13. In the first row, *F*-measure is shown using reference Algorithms 1, 2 and proposed Algorithm 3 for the whole sequences. The second row shows example frames of *Mountain* (frame 95), *Stefan* (frame 196) and *Race* (frame 15) and the third, fourth and fifth rows show the segmentation examples using Algorithms 1, 2 and 3 respectively. (a) *Mountain*, (b) *Stefan* and (c) *Race*.

decoded video sequence and the motion vectors are extracted from the coded stream. The reference software KTA [25] has been used. We perform evaluation using motion vectors derived from H.264/AVC motion estimation (IPPP …GOP structure, EPZS motion estimation with 32×32 search range, 4×4 smallest block size). A uniformly sampled 4×4 MV field is obtained by macroblock splitting (e.g. when there is only one motion vector per 16×16 macroblock, its value is assigned in every 4×4 sub-block of it). In case of INTRA mode macroblocks, there is no motion information and the macroblock is omitted from GME and Gaussian filter calculation.

Table 6 presents the results, where motion vector fields are obtained from encoding the test sequences with

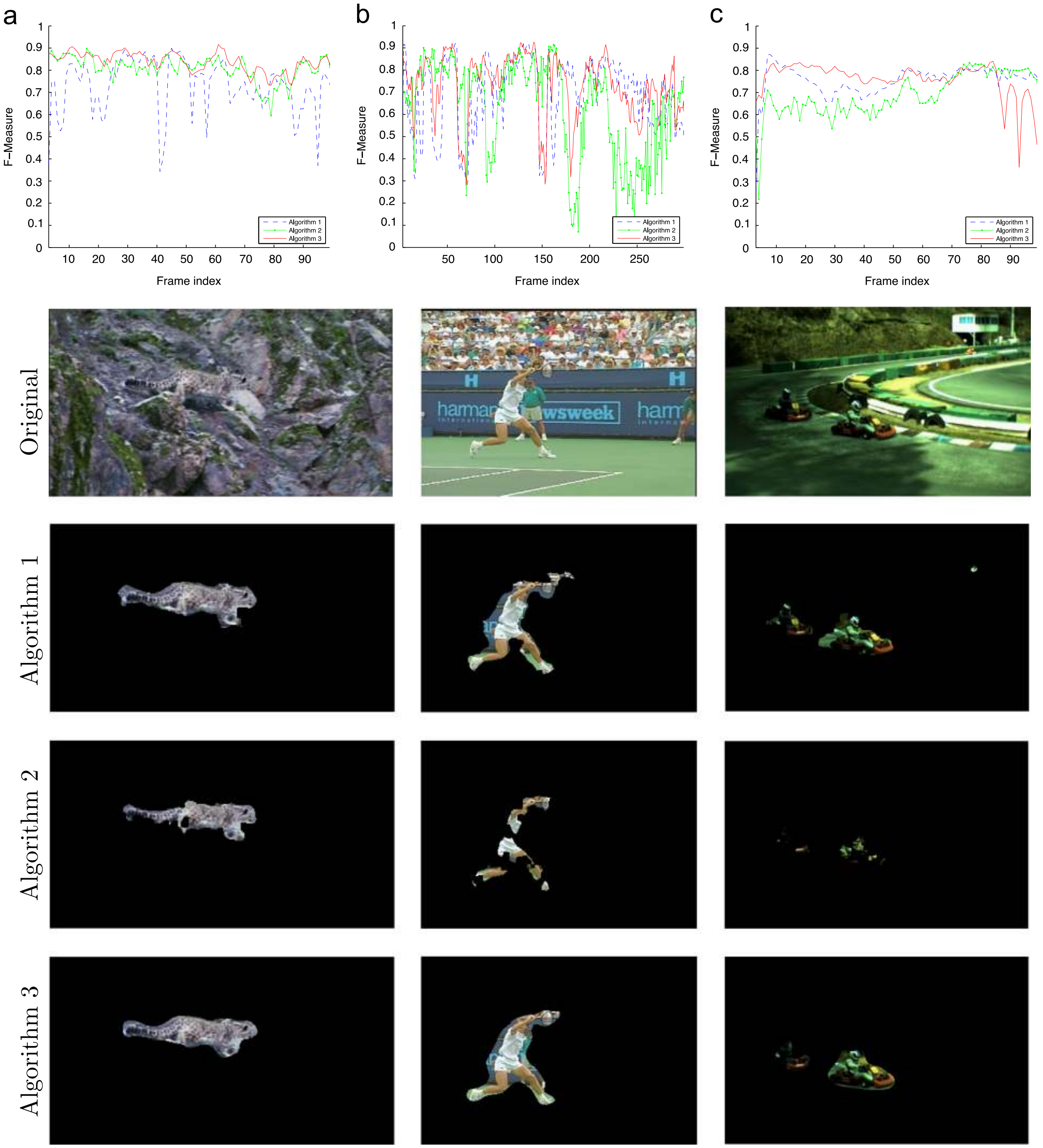

Fig. 14. In the first row, *F*-measure is shown using reference Algorithms 1, 2 and proposed Algorithm 3 for the whole sequences. The second row shows example frames of *Biathlon* (frame 173), *Allstars* (frame 162), *BBC fish* (frame 103) and *Horse* (frame 41) and the third, fourth and fifth rows show the segmentation examples using Algorithms 1, 2 and 3 respectively. (a) *Biathlon*, (b) *Allstars*, (c) *BBC fish* and (d) *Horse*.

various *Quantization Parameters* ($QP \in \{4, 16, 28, 38\}$) and Fig. 16 provides an overview. The results show that our approach is quite robust against bit rate changes, where motion information is not always representing real motion due to rate distortion optimization. By increasing *QP*, the number of SKIP macroblocks is also increased resulting in motion vectors with unreliable motion. Nevertheless, the results appear to be quite stable; up to $QP=28$ the maximum loss, in terms of *F*-measure compared with the $QP=4$ case, is 1% and for $QP=38$ the corresponding maximum loss is 13% for the *Horse* sequence.

In the cases of *Allstars*, *Stefan* and *BBC fish*, a slight increase (up to 2%) in terms of *F*-measure is observed by increasing *QP*. This can be explained, considering the fact that these sequences contain homogenous areas (soccer field, tennis field, blue sea) which, by increasing *QP*, are increasingly blurred as a consequence of the H.264/AVC deblocking filtering. This results in stronger blurring of minor details (spots in the sports field, spots in the sea, etc.) and also increases the number of large macroblocks that potentially follow global motion, thus benefiting global motion estimation and eventually segmentation.

Our approach can also be employed in cases of B-Frames presence. The advantage in this case would be the availability of motion vector fields from two directions in the encoder, and the disadvantage that the motion vector information may be prone to errors due to the larger distance between reference frames and subsequently larger displacements. Regarding I-frames, that contain no inter-frame motion displacement information, the adjacent P-frames' segmentation masks could be temporally interpolated in order to assign segmentation masks to them. When applying our segmentation approach on MPEG-2 streams, a slight quality decrease in terms of *F*-measure should be expected as MPEG-2 only uses half-pel motion compensation, instead of quarter-pel that H.264/AVC uses, and does not use deblocking filters.

Table 4
Contribution of the background temporal refinement to the performance improvement in terms of average Precision (*P*), Recall (*R*) and *F*-measure (*F*).

Sequence	Without BTR [15]			With BTR			ΔF (%)
	P	*R*	*F*	*P*	*R*	F	
Allstars	0.71	0.66	0.67	0.77	0.59	0.65	−1.6
Biathlon	0.71	0.94	0.80	0.78	0.87	0.82	+1.6
Mountain	0.77	0.90	0.83	0.84	0.85	0.84	+1.2
Race	0.63	0.87	0.73	0.74	0.83	0.78	+5.2
Stefan	0.61	0.83	0.69	0.71	0.79	0.73	+4.2
BBC fish	0.74	0.89	0.80	0.82	0.83	0.81	+2.1
Horse	0.81	0.65	0.72	0.88	0.70	0.78	+5.7

Table 5
Average *F*-measure for Otsu, weighted mean (wm) and hysteresis weighted mean (hwm) thresholding algorithms.

Sequence	OTSU [24]	WM [14]	HWM
Allstars	0.58	0.61	0.65
Biathlon	0.79	0.81	0.82
Mountain	0.82	0.82	0.84
Race	0.75	0.75	0.78
Stefan	0.69	0.71	0.73
BBC fish	0.75	0.76	0.81
Horse	0.67	0.68	0.77

4. Summary and conclusions

An unsupervised motion-based object segmentation algorithm for video sequences with moving camera has been presented. The proposed algorithm is based on bidirectional inter-frame change detection and the proposed motion compensated error fusion scheme outperforms the previously proposed ones. In addition to that, spatial error localization is considered in the thresholding step for improving segmentation efficiency in terms of

Table 6
Average Precision, Recall and *F*-measure for various quantization parameters. The PSNR column indicates average PSNR values (in dB) between raw video sequences and ones coded with QP. $\Delta_F = F_{QP=4} - F_{QP}$ and $\Delta_{PSNR} = PSNR_{QP=4} - PSNR_{QP}$.

Sequence	QP	PSNR	P	R	F	$\Delta_{\mathbf{PSNR}}$	$\Delta_{\mathbf{F}}$
Allstars	**4**	59.03	0.76	0.60	0.66	–	–
	16	47.18	0.76	0.59	0.65	−11.84	−0.01
	28	37.71	0.76	0.61	0.66	−21.31	0.01
	38	30.89	0.77	0.63	0.68	−28.14	0.02
Biathlon	**4**	59.89	0.77	0.88	0.82	–	–
	16	47.12	0.77	0.88	0.82	−12.77	0.00
	28	38.01	0.77	0.88	0.81	−21.87	−0.01
	38	31.69	0.74	0.88	0.80	−28.20	−0.02
Mountain	**4**	59.11	0.83	0.86	0.84	–	–
	16	46.11	0.83	0.86	0.84	−13.00	0.00
	28	34.53	0.82	0.87	0.84	−24.58	0.00
	38	27.01	0.81	0.87	0.83	−32.10	−0.01
Race	**4**	59.76	0.74	0.84	0.78	–	–
	16	46.57	0.74	0.84	0.78	−13.18	0.00
	28	37.43	0.74	0.84	0.78	−22.33	0.00
	38	30.89	0.73	0.79	0.74	−28.87	−0.04
Stefan	**4**	59.87	0.74	0.80	0.75	–	–
	16	46.43	0.74	0.80	0.75	−13.44	0.00
	28	35.93	0.72	0.80	0.74	−23.94	−0.01
	38	26.75	0.76	0.78	0.76	−33.13	0.01
BBC fish	**4**	59.46	0.81	0.84	0.82	–	–
	16	49.14	0.82	0.87	0.84	−10.32	0.02
	28	42.98	0.82	0.82	0.81	−16.48	−0.01
	38	36.67	0.78	0.72	0.74	−22.78	−0.08
Horse	**4**	59.99	0.90	0.66	0.76	–	–
	16	46.55	0.90	0.66	0.76	−13.45	0.00
	28	34.84	0.90	0.66	0.76	−25.16	0.00
	38	27.92	0.76	0.55	0.63	−32.07	−0.13

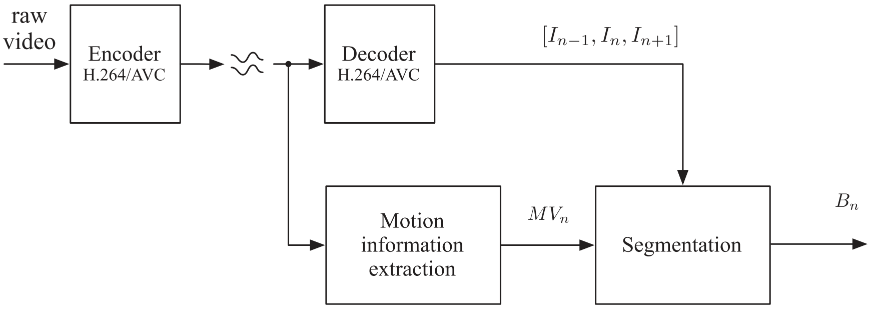

Fig. 15. System input when implemented at the decoder side.

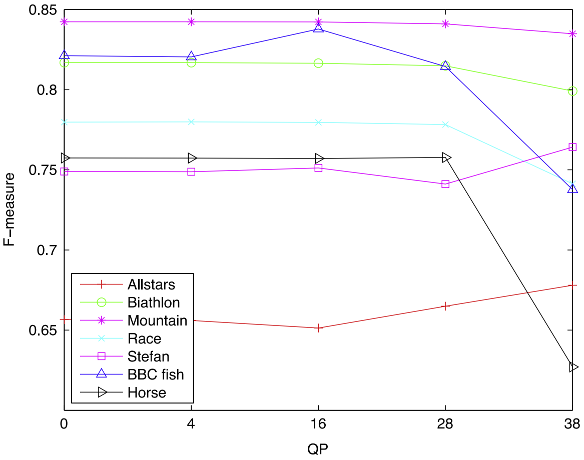

Fig. 16. Average *F*-measure for segmenting at decoder side under various quantization parameters.

F-measure. The issue of optimal weight selection for weighted mean hysteresis thresholding is addressed employing a statistical approach. This enables robust segmentation performance that avoids heuristics and training algorithms for parameter selection that are common approaches. Furthermore, a final post-processing step is incorporated to enable temporal consistency of the segmentation masks using filtering of the preliminary binary masks, which is adapted according to the motion of the foreground. The experimental evaluation demonstrates the validity of the proposed method and is also shown that it is quite robust under various quantization parameters that influence motion estimation quality.

Acknowledgments

The research leading to these results has received funding from the European Community FP7 under Grant agreement number 261743 (NoE VideoSense). We would also like to thank Prof. Aishy Amer of Concordia University for providing the executable implementation of the method from [9].

Appendix A. Additional material

For the original and segmented video sequences with reference and proposed algorithms, as well as further *F*-measure, Precision, Recall curves refer to http://www.nue.tu-berlin.de/research/stmos-br/.

References

[1] R. Radke, S. Andra, O. Al-Kofahi, B. Roysam, Image change detection algorithms: a systematic survey, IEEE Transactions on Image Processing 14 (3) (2005) 294–307.

[2] T. Aach, A. Kaup, R. Mester, Statistical model-based change detection in moving video, Signal Processing 31 (2) (1993) 165–180.

[3] A. Cavallaro, T. Ebrahimi, Accurate video object segmentation through change detection, in: Proceedings of the IEEE International Conference on Multimedia and Expo (ICME), vol. 1, 2002, pp. 445–448.

[4] S. Berrabah, G. De Cubber, V. Enescu, H. Sahli, MRF-based foreground detection in image sequences from a moving camera, in: IEEE International Conference on Image Processing (ICIP), 2006, pp. 1125–1128.

[5] A. Bugeau, P. Perez, Detection and segmentation of moving objects in complex scenes, Computer Vision and Image Understanding 113 (4) (2009) 459–476.

[6] N. Goyette, P. Jodoin, F. Porikli, J. Konrad, P. Ishwar, Changedetection. net: a new change detection benchmark dataset, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2012, pp. 1–8.

[7] C. Kim, J.-N. Hwang, Fast and automatic video object segmentation and tracking for content-based applications, IEEE Transactions on Circuits and Systems for Video Technology 12 (2) (2002) 122–129.

[8] A. Alatan, L. Onural, M. Wollborn, R. Mech, E. Tuncel, T. Sikora, Image sequence analysis for emerging interactive multimedia services—the European cost 211 framework, IEEE Transactions on Circuits and Systems for Video Technology 8 (7) (1998) 802–813.

[9] B. Qi, M. Ghazal, A. Amer, Robust global motion estimation oriented to video object segmentation, IEEE Transactions on Image Processing 17 (6) (2008) 958–967.

[10] Y. Kameda, M. Minoh, A human motion estimation method using 3-successive video frames, in: Proceedings of International Conference on Virtual Systems, 1996, pp. 135–140.

[11] M.-Y. Shih, Y.-J. Chang, B.-C. Fu, C.-C. Huang, Motion-based background modeling for moving object detection on moving platforms, in: Proceedings of the International Conference on Computer Communications and Networks, 2007, pp. 1178 –1182.

[12] J.-C. Huang, T.-S. Su, L.-J. Wang, W.-S. Hsieh, Double-change-detection method for wavelet-based moving-object segmentation, Electronics Letters 40 (13) (2004) 798–799.

[13] H. Liu, X. Chen, Y. Chen, C. Xie, Double change detection method for moving-object segmentation based on clustering, in: Proceedings of the IEEE International Symposium on Circuits and Systems, 2006, p. 4.

[14] A. Krutz, M. Kunter, M. Mandal, M. Frater, T. Sikora, Motion-based object segmentation using sprites and anisotropic diffusion, in: Proceedings of the International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS), 2007, p. 35.

[15] M. G. Arvanitidou, M. Tok, A. Krutz, T. Sikora, Short-term motion-based object segmentation, in: Proceedings of the IEEE International Conference on Multimedia & Expo, Barcelona, Spain, 2011, pp. 1–6.

[16] M. Tok, A. Glantz, M. G. Arvanitidou, A. Krutz, T. Sikora, Compressed domain global motion estimation using the Helmholtz tradeoff estimator, in: Proceedings of the IEEE International Conference on Image Processing, Hong Kong, 2010, pp. 777–780.

[17] P.J. Rousseeuw, A.M. Leroy, Robust Regression and Outlier Detection, John Wiley & Sons, Inc., 1987.

[18] P. Perona, J. Malik, Scale-space and edge detection using anisotropic diffusion, IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (7) (1990) 629–639.

[19] P. Rosin, T. Ellis, Image difference threshold strategies and shadow detection, in: Proceedings of the British Machine Vision Conference, BMVA Press, 1995, pp. 347–356.

[20] E. Hancock, J. Kittler, Adaptive estimation of hysteresis thresholds, in: IEEE Conference on Computer Vision and Pattern Recognition, 1991, pp. 196–201.

[21] Y. Yitzhaky, E. Peli, A method for objective edge detection evaluation and detector parameter selection, IEEE Transactions on Pattern Analysis and Machine Intelligence 25 (8) (2003) 1027–1033.

[22] R. Medina-Carnicer, F. Madrid-Cuevas, A. Carmona-Poyato, R. Mu noz-Salinas, On candidates selection for hysteresis thresholds in edge detection, Pattern Recognition 42 (7) (2009) 1284–1296.

[23] D. Farin, T. Haenselmann, S. Kopf, G. Khne, W. Effelsberg, Segmentation and classification of moving video objects, in: Handbook of Video Databases, vol. 8, CRC Press, Boca Raton, FL, USA, 2002, pp. 561–591.

[24] N. Otsu, A threshold selection method from gray-level histograms, IEEE Transactions on Systems, Man and Cybernetics 9 (1) (1979) 62–66.

[25] H.264/AVC KTA Software. Available at: 〈http://www.tnt.uni-hannover.de/∼vatis/kta/〉, November 2010.