Intuitive Strategy for Parameter Setting in Video Segmentation

Elisa Drelie Gelasca, Elena Salvador and Touradj Ebrahimi

Signal Processing Institute (ITS)
Swiss Federal Institute of Technology (EPFL)
CH-1015 Lausanne, Switzerland

ABSTRACT

In this paper, we propose an original framework for an intuitive tuning of parameters in image and video segmentation algorithms. The proposed framework is very flexible and generic and does not depend on a specific segmentation algorithm, a particular evaluation metric, or a specific optimization approach, which are the three main components of its block diagram. This framework requires a manual segmentation input provided by a human operator as he/she would have performed intuitively. This input allows the framework to search for the optimal set of parameters which will provide results similar to those obtained by manual segmentation. On one hand, this allows researchers and designers to quickly and automatically find the best parameters in the segmentation algorithms they have developed. It helps them to better understand the degree of importance of each parameter's value on the final segmentation result. It also identifies the potential of the segmentation algorithm under study in terms of best possible performance level. On the other hand, users and operators of systems with segmentation components, can efficiently identify the optimal sets of parameters for different classes of images or video sequences. In a large extent, this optimization can be performed without a deep understanding of the underlying algorithm, which would facilitate the exploitations and optimizations in real applications by non-experts in segmentation. A specific implementation of the proposed framework was obtained by adopting a video segmentation algorithm invariant to shadows as segmentation component, a full reference segmentation quality metric based on a perceptually motivated spatial context, as the evaluation component, and a down-hill simplex method, as optimization component. Simulation results on various test sequences, covering a representative set of indoor and ourdoor video, show that optimal set of parameters can be obtained efficiently and largely improve the results obtained when compared to a simple implementation of the same segmentation algorithm with ad-hoc parameter setting strategy.

Keywords: Video Object Segmentation, Parameter Setting, Quality Evaluation, Ground-truth

1. INTRODUCTION

Segmentation is one of the fundamental problems in computer vision and image analysis and much effort has been devoted to it in the past three decades. It is well-known that segmentation is an ill-posed problem[1] and because of this fact, a priori knowledge is needed in order to resolve it. Most a priori knowledge is determined from the context in which segmentation takes place. As an example, in a surveillance application, one can consider every moving object as an object of interest and therefore this knowledge can be used to achieve segmentation. Often, such simplistic assumptions are not sufficient and may lead to weaknesses. In the latter example for instance, moving shadows may also be segmented as objects of interest, even if they might not represent an object of interest. Such observations usually lead to rather sophisticated segmentation algorithms in which several parameters are to be tuned and calibrated. The effect of such tuning is not often translating into a straightforward and human understandable impact on the segmentation result. This makes the problem of segmentation a rather difficult task. A further complexity stems from the fact that in general different sets of parameters should be found for different classes of image and video sequences. In addition to the above, it has been pointed out that low-level vision algorithms often do not provide human understandable results from semantic point of view. A high-level vision algorithm therefore becomes necessary in applications in order to

Correspondance: Elisa Drelie Gelasca, E-mail: elisa.drelie@epfl.ch

allow for semantically meaningful segmentation results.[2] In several previous works[2,3,4] the task of segmentation has been divided into two parts. First part concentrates on low-level processing which can be rather efficiently implemented in computers. The semantic input is provided in a second part either from a more high-level processing through a more semantic processing (artificial intelligence) or simply from a human user who will correct, guide or hint the segmentation algorithm in producing the final segmentation result. The order in which the human intervention is processed has its importance. Supervised segmentation[5] starts with user intervention, often by indicating the seed of regions which have to be segmented. The algorithm (region growing for instance) then terminates the segmentation task. An interactive segmentation[2] refers to an approach in which the low-level segmentation is first performed by the computer, the result of this stage is then corrected, interpreted, or completed by human user intervention, which could be as simple as identifying which of the homogeneous regions (according to a criteria such as textureness, motion, color, etc,) belong to the same semantic object. In this approach, a distinction is made between the notion of regions and that of objects. Regions refer to set of pixels which share a given property such as color, motion or texture and closely related to low-level processing. Objects are set of regions which define semantically meaningful regions. They embed high-level processing information. This approach has been further extended to segmentation and tracking of video, where segmentation and tracking are performed on a region by region basis, augmented by object basis considerations.

Another important shortcoming of segmentation is the lack of ground-truth and a reliable quality metric.[6] As opposed to image compression, in which the original uncompressed image is considered as the ground-truth (the reference in terms of desired quality), there is no ground-truth segmentation result readily and clearly available in most segmentation problems. In other words, the optimal segmentation is not known. Past work in no-reference quality metrics for image and video has been used to counter this problem. In a no-reference quality metric,[7] instead of approaching the result to a truth reference, one aims at first defining characteristics of a good quality image (in the case of segmentation, a good segmentation result). The quality of the image (or segmentation in our case) is then assessed by examining the degree in which the algorithm approaches the good characteristics mentioned above. Past work both in image quality assessment and segmentation quality assessment show there is a good potential behind this approach. Results obtained however are quite preliminary and unacceptable in terms of fidelity and correlation with a subjective metric performed by a human being. This is especially true for the case of segmentation quality metrics. This is an area in a open field of research and in its early days. A question to ask is:"Is there any practical use case for reference based[8,9,10] quality metrics?". The answer to this question, when considering the distortion metrics due to processing and compression is yes, as witnessed by the large amount of work in the fields of rate-distortion optimization, and image enhancement. This paper aims at bringing an interesting use case for the dual problem of segmentation metrics using a reference.

The paper is structured as follows. In Sec.2, we described the overall proposed framework and a specific implementation example. Experimental results are shown in Sec. 3. Finally, we draw the conclusions in Sec. 4.

2. PROPOSED FRAMEWORK

Figure 1 provides a block diagram of the general use case for segmentation algorithms that could benefit from a reference based segmentation quality metric. The proposed system is composed of three main blocks: a segmentation module, a quality evaluation module, and an optimization module. The segmentation block extracts moving objects from each frame of the video sequence. It is described in Sec. 2.1. Once extracted, the video objects are then analyzed in the second block of the system. Here, an objective evaluation metric is used to provide an assessment of the quality of the segmentation results. Section 2.2 presents the segmentation evaluation metric. For each frame in the sequence, the evaluation is then used in the optimization module to select the best set of parameter values for the segmentation algorithm of the first block of the system. Once the optimal parameters have been determined, their values are used to extract the final moving object collection for the frame under analysis. The parameter optimization process is illustrated in Sec. 2.3.

This framework aims at using the human intervention in order to efficiently tune and calibrate the parameters of a segmentation algorithm in a more intuitive and human understandable way. As mentioned in the previous section, the impact and role of a given parameter in the segmentation result is often difficult to understand from an intuitive point of view. In addition, tuning of parameters needs a detailed understanding of the algorithm used. To this, an additional complexity may be added if the number of parameters to tune is large, as it is

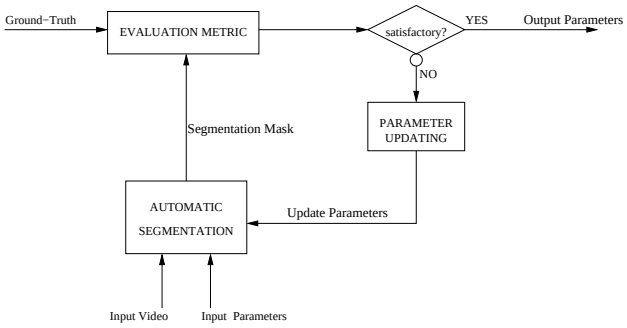

Figure 1. Block diagram of the proposed framework. Inputs are a ground-truth video object segmentation provided by the user, the video sequence, and an initial set of segmentation parameters. A segmentation module extracts moving objects from each frame of the video sequence; then, the object segmentation is assessed by an objective evaluation metric; finally, the segmentation parameters are optimized according to the evaluation result.

the case in many applications. Furthermore, such an approach will identify the degree of importance of each parameter of segmentation in the final results.

2.1. Video Object Segmentation

The video object extraction algorithm we consider in this paper is composed of two sequential steps: a moving object extraction step based on change detection[11] and a post-processing step that identifies[12] and removes shadows from the change detection results.

Change detection identifies changes in image sets or image sequences at two different time instants. In the context of semantic video object segmentation, the expected output of a change detection algorithm is a classification of pixels in each frame of the video sequence into one of two classes: foreground (moving objects) and background pixels. To this end, in the considered object segmentation algorithm, the difference between each frame of the sequence and a reference frame is analysed and classified. The reference frame represents the background of the scene. It can correspond to a frame in the sequence or to a reconstructed one. The image difference $D(x, y)$ is computed as $D(x, y) = |I(x, y) - I_r(x, y)|$ at each pixel position (x, y). $I(x, y)$ represents the luminance component of the image and $I_r(x, y)$ the luminance component of the reference image. Temporal changes identified by means of image differentiation may be generated not only by moving objects, but also by noise components. The main sources of noise are sensor noise and changes in the illumination conditions. The former is eliminated in the classification step, while the latter is handled in the post-processing step.

In order to eliminate camera noise, a statistical approach is adopted for the classification of image differences into changed (foreground) or unchanged (background) pixels. The approach models the noise statistics and applies a significance test in order to separate the contribution due to noise from those due to moving objects. In a noise free case, the condition $D(x, y) > 0$ would suffice to state that the pixel (x, y) belongs to the foreground. In real situations, noise alters the above test, and the test should be transformed to $D(x, y) > b$, where b takes care of the distortions introduced by noise. To obtain a more robust analysis for each pixel position, a square observation window, $\mathcal{W}_{(x,y)}$, centered in (x, y), is considered. In $\mathcal{W}_{(x,y)}$, the sum of differences $D_w(x, y)$, is computed as $D_w(x, y) = \frac{1}{W^2} \sum \sum_{(i,j)\in W} D(i, j)$. If the sum of differences is larger than the threshold b, then (x, y) belongs to a moving object. The threshold b is content dependent and should be therefore tuned for each sequence. Not all the parameters of a segmentation algorithm need to be manually tuned if there is a way to automatically adapt their values to the changes in the image sequence content. In the case of the considered algorithm, a locally adaptive threshold based on a probabilistic approach is employed. The thresholding approach is based on the assumption that the noise in the signal $D_w(x, y)$ respects a Gaussian distribution. A significance test then checks the validity of the hypothesis that a sample $D_w(x, y)$ comes from the Gaussian distribution. The pixel (x, y) is classified as a foreground pixel if the significance test is respected.

The parameters of the change detection algorithm are: the size W^Y of the observation window $\mathcal{W}_{(x,y)}$; the standard deviation of the noise distribution; and the significance level for the statistical test. The standard deviation is related to the variance of the camera noise and is automatically computed and updated over time based on the input data. The significance level is a rather stable parameter and does not need to be tuned along a sequence and for different sequences. The only parameter that therefore needs to be manually tuned is the size W^Y of the observation window $\mathcal{W}_{(x,y)}$. W^Y represents the number of pixels on which the statistics for the significance test are computed. By increasing W^Y, the statistics on $\mathcal{W}_{(x,y)}$ will be more reliable. This results in a reduced sensitivity to noise and in a more correct classification. However, the probability that the tested hypothesis remains valid on all pixels in $\mathcal{W}_{(x,y)}$ decreases. This causes a wrong classification of pixels along the edge of moving objects and the corresponding *halo effect*.

The second main source of noise that could lead to false alarms is given by local changes of illumination conditions, such as shadows cast by moving objects. Since shadows generate temporal changes, the detection of a moving object by the change detection step may include the detection of its shadow or part of its shadow. The post-processing step is used to recognize and to eliminate the contributions caused by shadows from those caused by the moving objects. The inputs to the post-processing stage are the binary masks resulting from change detection, original frames and reference frames. The presence of a shadow is first hypothesized by exploiting the assumption that a shadow region is darker than the same background region if there was no shadow. Then, this hypothesis is verified by analyzing photometric invariant color features and geometrical information.

Similarly to the change detection step, changes in the intensity values of the three colour channels R, G, B in the frame under analysis with respect to the reference frame are analysed. The image difference $\mathbf{D}(x,y)$, computed as $\mathbf{D}(x,y) = (D_R(x,y), D_G(x,y), D_B(x,y))$ at each pixel (x,y) position belonging to the change detection mask, is now considered. $D_R(x,y) = R_r(x,y) - R(x,y)$ and similar equations hold for G and B. $R_r(x,y)$ belongs as before to the reference image. In a noise free case, the condition $(D_R(x,y), D_G(x,y), D_B(x,y)) > (0,0,0)$ would suffice to state that the pixel (x,y) is darker than the corresponding background and therefore belongs to a candidate shadow. In real situations, the test becomes $D_R(x,y) < b_1, D_G(x,y) < b_2, D_B(x,y) < b_3$. To obtain a more robust analysis for each pixel position a square observation window, $\mathcal{W}_{(x,y)}$, centered in (x,y), is considered and the sum of differences is analyzed. To avoid the tuning of the threshold, the statistical approach described aboved is employed at this stage on the three color channnels. The pixel (x,y) is defined as candidate shadow pixel if the significance test is respected in all colour channels.

This first level of analysis leads to the detection of shadow pixels but also of object pixels that are darker than the corresponding background. The analysis of changes in the invariant colour features[13] allows to refine the hypothesis. The presence of a shadow does not alter the value of the invariant colour features. On the contrary, a material change modifies their value. Therefore, the difference in the $c_1c_2c_3$ invariant feature values, $\mathbf{d}(x,y)$, is analysed. As stated above, the condition $\mathbf{d}(x,y) = \mathbf{0}$ would suffice to state that the pixel (x,y) belongs to a candidate shadow in a noise free case. In real situations, the test becomes $|d_{c_1}(x,y)| < T_1, |d_{c_2}(x,y)| < T_2, |d_{c_3}(x,y)| < T_3$,. As for the first level, a window $\mathcal{W}_{(x,y)}$, centered in (x,y) is considered, and the sum of differences $\mathbf{d}_w(x,y)$ is analyzed. If $|\mathbf{d}_w(x,y)|$ is below the threshold for each component, then the shadow hypothesis is strenghtened for pixel (x,y). The setting of the threshold $\mathbf{T} = (T_1, T_2, T_3)$ is driven by experiments on different sequences. It is different from the one used for RGB, because the dynamic range of the invariant features is smaller than those of the RGB components.

The last evidence about the existence of a shadow is derived from geometrical properties. This verification is based on analysis of boundary of the candidate shadow regions and test of the position of shadows with respect to objects. A necessary condition for the existence of a shadow is given by the presence of a line that separates the shadow pixels from the background pixels. Therefore, in case a hypothesized shadow is fully included in an object, the shadow hypothesis is discarded.

The parameters that need careful manual tuning in the post-processing step are: the size W^{RGB} of the observation window $\mathcal{W}_{(x,y)}$ for the test on the RGB color components; the size W^{Inv} of the observation window $\mathcal{W}_{(x,y)}$ for the test on the invariant color features; finally, the threshold $\mathbf{T}$. The same considerations made for the window size in the change detection step can be used for the post-processing step. Moreover, the value of the threshold $\mathbf{T}$ is critical for an accurate detection and elimination of shadows.

2.2. Evaluation Metric

An evaluation metric is adopted to assess the quality of the video object segmentation results. The performance evaluation metric[10] is based on the availability of a *ground-truth* segmentation which represents the ideal segmentation and can be generated either manually or via a reliable procedure. The metric is defined on two kinds of errors, namely objective errors and perceptual errors. Objective errors can be simply obtained by computing the deviation of the resulting segmentation from the ground-truth. The perceputal errors are based on the fact that different errors are more or less salient according to their category. Different categories of errors contribute therefore differently to the resulting segmentation quality.

2.2.1. Objective errors

An algorithm for object segmentation can in principle be evaluated by estimating the amount of undetected pixels (*false negative*) and the amount of incorrectly detected pixels (*false positive*). Let us denote by $C(k)$ the set of pixels segmented at frame k, and with $C_r(k)$ the pixels belonging to the reference segmentation. The set of false positive errors, $\epsilon_p(k)$, can be expressed as

$$\epsilon_p(k) = card\big(C(k) \cap \bar{C}_r(k)\big) \tag{1}$$

where the function $card(\cdot)$ represents the cardinality of a set, and $\bar{C}_r(k)$ is the complement of $C_r(k)$. *False negatives*, $\epsilon_n(k)$, appearing in the reference segmentation $C_r(k)$, but not in the result under analysis, $C(k)$, can be expressed as

$$\epsilon_n(k) = card\big(\bar{C}(k) \cap C_r(k)\big) \tag{2}$$

By computing the total amount of false detections, a simple objective measure of the spatial accuracy of segmentation results can be obtained. Using Eq. (1) and Eq. (2), a measure of the *absolute spatial accuracy* can be derived at frame k

$$\epsilon(k) = \epsilon_p(k) + \epsilon_n(k) \tag{3}$$

corresponding to the amount of false detections for each time instant k. The larger ϵ, the lower the spatial accuracy is. The measure of spatial accuracy so defined here, is an objective discrepancy parameter that quantifies the deviation of the segmentation result from the ground-truth provided by the human operator.

The significance of the error value, $\epsilon(k)$, depends on both the size of the segmented object and of the ground-truth. The larger $card\big(C_r(k)\big)$, the less important is $\epsilon(k)$. Similarly, the larger the object detected, $card\big(C(k)\big)$, the less important is $\epsilon(k)$. For this reason, a relative measure of the total amount of false detections is introduced, referred to as *relative spatial accuracy*. The relative spatial accuracy can be computed by normalizing the total amount of false detections by the total number of possible false pixels. Therefore, the *relative spatial accuracy* is defined:

$$\epsilon'(k) = \begin{cases} 0 & \text{if } card\big(C_r(k)\big) = 0 \text{ and } card\big(C(k)\big) = 0, \\ \frac{1}{card\big(C(k)\big) + card\big(C_r(k)\big)}\epsilon(k) & \text{otherwise.} \end{cases} \tag{4}$$

We define the *objective spatial accuracy*, $\nu(k)$, to be *inversely* proportional to the amount of deviations between resulting segmentation and ground-truth as follows:

$$\nu(k) = 1 - \epsilon'(k), \tag{5}$$

with $\nu(k) \in [0, 1]$. The value $\nu(k) = 1$ indicates perfect spatial accuracy at frame k, that is, a perfect match between segmentation results and the ground-truth.

2.2.2. Error saliency

The measure of spatial accuracy proposed in Eq. (5) is an objective discrepancy parameter that quantifies the deviation of the segmentation result at hand from the ground-truth segmentation. It has to be taken into account, however, that a human observer, gives effectively different importance to different errors. Therefore, errors have to be weighted differently according to their visual importance, since an evaluation of segmentation results similar to that of a human observer is aimed. By identifying different perceptual errors and by weighting them properly, we achieve a weighted spatial accuracy.

A false positive contributes differently to the quality than a false negative. Missing parts of objects (holes), in fact, are more salient in terms of error than added parts (background). In addition to this, the more we move away from the border of the object, the more the error is annoying. Therefore, false negatives are more significant than false negative, and the larger the distance from the nearest correct object, the more significant is the error. Consequently, the weights for false positives, w_p are different from those for false negatives, w_n, and increase with distance from the object contour. The weights for false negative pixels are larger than those for false positive pixels at the same distance from the border of the object.

By considering the spatial context, the measure of the spatial accuracy, $\epsilon_w(k)$, becomes

$$\epsilon_w(k) = \sum_{i=1}^{\epsilon_p(k)} w_p^i + \sum_{j=1}^{\epsilon_n(k)} w_n^j. \tag{6}$$

The relative spatial accuracy can be computed by normalizing the total amount of $\epsilon_w(k)$ by the total number of possible weighted false pixels. The maximum amount of possible false positives is equal to the number of elements in the segmentation, $card\big(C_n(k)\big)$ multiplied by the positive weighting factor. The maximum amount of false negatives is given, in the worst case, by the number of elements in the reference mask (ground-truth), $card\big(C_r(k)\big)$, multiplied by the negative weighting factor. It follows:

$$\epsilon'_w(k) = \begin{cases} 0 & \text{if } card\big(C_r(k)\big) = 0 \text{ and } card\big(C(k)\big) = 0, \\ \frac{1}{card\big(C(k)\big) \times w_p + card\big(C_r(k)\big) \times w_n} \epsilon_w(k) & \text{otherwise.} \end{cases} \tag{7}$$

Finally, the weighted spatial accuracy is defined as:

$$\nu_w(k) = 1 - \epsilon'_w(k). \tag{8}$$

The weighted spatial accuracy $\nu_w(k)$ is the *quality function* used for evaluation of the best set of parameters for the segmentation algorithm in our proposed strategy.

2.3. Optimization of Segmentation Parameters

In our strategy, the values of the segmentation parameters described in Sec. 2.1 are adapted for each frame of the input video sequence according to the varying content by maximizing the quality function $\nu_w(k)$ described in Sec. 2.2.2. The quality function depends on the parameter set $\mathbf{P} = [p_1, p_2, ..., p_j, ..., p_N]$ of the segmentation scheme at hand, where p_j is the specific parameter and N is the number of parameters.

For the specific segmentation considered in this paper, the main parameters are:

- the size of the observation window for the change detection analysis W^Y;
- the size of the observation window for the analysis on RGB color components W^{RGB};
- the size of the observation window for the analysis on photometric invariant color components W^{Inv};
- the threshold for the photometric invariant color analysis $\mathbf{T}$.

A statistical approach described in Sec. 2.1 is employed for tuning the threshold both for change detection analysis and for RGB color components analysis.

Therefore, N is 4 and the parameters are:

$$\mathbf{P}(k) = [W^Y(k), W^{RGB}(k), W^{Inv}(k), \mathbf{T}(k)] \tag{9}$$

at frame k.

To maximize the quality function at each frame k, as the quality function $\nu_w(k)$ has more than one independent variable, a multidimensional optimization has to be considered. In the literature there are no systematic ways

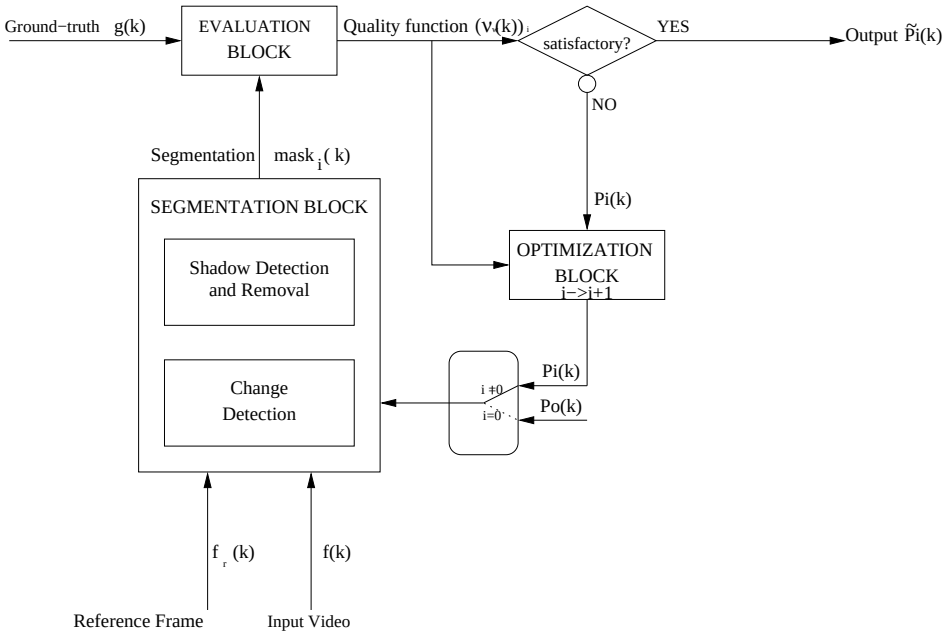

Figure 2. Block diagram of the overall proposed strategy. At each frame k and iteration i, the inputs are: the input frame $f(k)$, the reference frame $f_r(k)$, the parameter vector $\mathbf{P}_i(k)$ and the ground-truth $g(k)$. In the segmentation block $f(k)$ and $f_r(k)$ are processed and the segmentation mask $m_i(k)$ is obtained. In the evaluation block, the quality of the $m_i(k)$ is evaluated and compared to $g(k)$. In this block, the quality function $(\nu_w(k))_i$ is computed. Then, the quality obtained undergoes the test of satisfactory or not. In case the satisfaction is not reached, i becomes $i+1$ and a new set of parameters $\mathbf{P}_i(k)$ is defined by the optimization block for the same frame k. In case at iteration i the satisfaction is reached, the set of parameters $\mathbf{P}_i(k)$ is stored as the best for that frame, $\widetilde{\mathbf{P}}_i(k)$. Successively, the time is incremented and the initial parameter vector $\mathbf{P}_0(k)$ is the parameter vector of the previous frame assessed as to be that providing the best segmentation.

to select the optimization algorithm. An optimization method can be selected among methods that need only evaluations of the function and methods that also require evaluations of the derivative of that function. It has to be taken into account that the computational effort is dominated by the cost of evaluation function. Algorithms using derivatives are somewhat more powerful than those using only the function, but not always enough so as to compensate for the additional calculations of derivatives. Consequently, a method without the computation of derivatives has been chosen. The optimization is performed using the *downhill simplex method*[14] . This method makes no special assumption about the function, handles the multidimensional case with a storage requirement of order N^2 and derivative calculation is not required.

For multidimensional optimization we have to set a starting point $\mathbf{P}_0$, that is an $N-$dimensional vector of independent variables. The algorithm is then supposed to make its own way until it encounters an optimum, up to the desired tolerance. The downhill simplex method starts with $N+1$ points. If we take $\mathbf{P}_0$ as the starting point, we need to take other N points to define the starting *simplex*:

$$\mathbf{P}_i = \mathbf{P}_0 + \lambda \mathbf{e}_i \tag{10}$$

where $\mathbf{e}_i$ are unit vectors and $\lambda = [\lambda_1, ..., \lambda_N]$ is a vector of constants that is fixed according to the problem's characteristic length scale. Summarizing, for the selected optimization method, the following should be fixed:

- the characteristic length scale, λ;
- the initial guess, $\mathbf{P}_0$;

- the tolerance of the results;

In the proposed framework, these three figures are choosen on the basis of the following considerations. λ depends on the nature of the parameters which it is dealing with. For example, it is meaningful to consider only integer values for the size of the observation windows, $W^Y(k)$, $W^{RGB}(k)$, $W^{Inv}(k)$ and thus to fix λ to an integer value. On the other hand, the average intensity computed on the pixels belonging to such a window may not take integer values. Hence, it makes more sense to put the threshold, $\mathbf{T}(k)$, equal to a non-integer value and therefore to choose λ consequently. The starting point, $\mathbf{P}_0$, should represent a compromise between mis-segmentation due to undetected pixel (false negative resulting in holes in the objects) and incorrectly segmented pixel (false positive: shadows, halo effect, added regions due to noise). The tolerance of the results is the fractional convergence tolerance[14] to be achieved by the function value. Once the tolerance has been selected, the precision of the numerical results that should be obtained for the quality function has been established. In our case, it makes little sense to attain a very high precision. In fact, the human eye cannot distinguish between two segmentation results whose numerical values of quality differ by less than a so-called *just noticeable difference*. Once the evaluation metric has been selected, by means of subjective experiments, the *just noticeable difference* in terms of segmentation quality will be translated into a numerical difference of quality values. This represents the tolerance.

The overall proposed strategy is depicted in the block diagram in Fig. 2. At the first iteration of the proposed strategy for the first frame, the initial frame, the reference frame, the starting parameter vector $\mathbf{P}_0$ are provided to the *segmentation block*. The quality of the segmentation results after each iteration is evaluated using the metric described in the *evaluation block*. At this stage, we require from the human operator an intuitive input, the ideal segmentation, in order to compare the two images and compute the quality function ν_w. Then, the numerical value of the quality obtained undergoes the test of *satisfactory or not*. In case the satisfaction is not reached, a new set of parameters $\mathbf{P}_{i+1}(k)$ is defined by the *optimization block* and a new iteration $i+1$ begins for the same frame k. In case at iteration i the satisfaction is reached, the set of parameters $\mathbf{P}_i(k)$ is stored as the best $\widetilde{\mathbf{P}_i}(k)$ for that frame k. Then, the time is incremented and the starting parameter vector $\mathbf{P}_0(k)$ is the parameter vector of the previous frame assessed as to be that providing the best segmentation: $\mathbf{P}_0(k) = \widetilde{\mathbf{P}_\mathbf{i}}(k)$.

3. EXPERIMENTAL RESULTS

Simulations have been performed to evaluate the performance of the proposed strategy. The experiments have been carried out on the MPEG-4 test sequence *Hall monitor*, the MPEG-7 test sequence *Highway*, as well as on the European IST project *art.live*[*] sequence *Group*. Both indoor and outdoor sequences, small and large foreground objects have been considered. The spatial resolution of the test sequences is 288×352 pixels (CIF format) and the temporal resolution is 25 images per second. The ground-truth segmentation for the test sequence *Hall Monitor* is provided by the European project COST 211 [†]. 60 images from the two other sequences have been segmented by hand to provide the user defined ground-truth: 30 images were taken from *Group*, and 30 images from *Highway*.

The results are visualized as follows. Two columns are displayed: on the left side, examples of segmentation results obtained without the optimization strategy are shown. The parameters are kept to the initial parameter vector $\mathbf{P}_0$ along the entire sequence. On the right, examples of segmentation results after the employment of the optimization strategy are shown for the corresponding frames in the same row. The ground-truth segmentation contour is superimposed in white on the segmentation results.

$\mathbf{P}_0$ has been selected as follows: at the change detection step the observation window W^Y is set to 5×5 pixels; in the RGB color component analysis, W^{RGB} is also set to 5×5; when the photometric invariant color features are tested, the observation window W^{Inv} size is 7×7; the value of the threshold $\mathbf{T}$ is set to 7 for each component. In our optimization strategy, the characteristic length of scale, λ_i, is fixed to 1 for $i = 1, 2, 3$ and to 0.5 for $i = 4$; the tolerance is equal to 1.0 e-4; the maximum number of iterations allowed for each frame is 30.

[*]European project IST 10942 *art.live*, http://www.tele.ucl.ac.be/PROJECTS/art.live/.

[†]http://www.tele.ucl.ac.be/EXCHANGE/

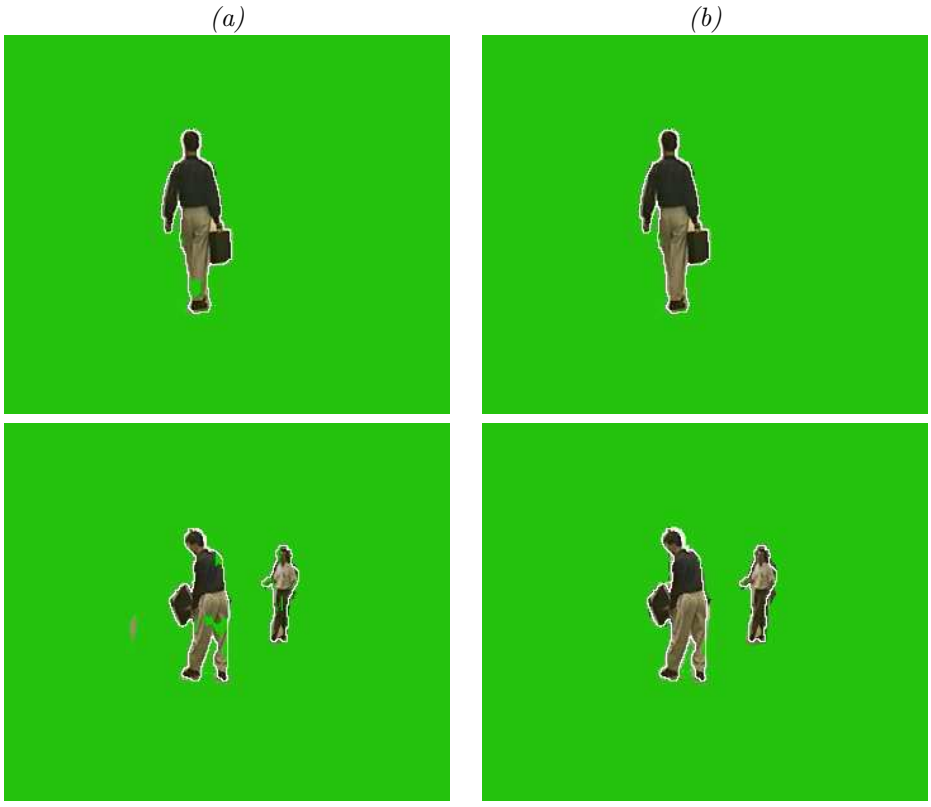

Figure 3. Segmentation results for test sequence *Hall Monitor*, at frames #55 and #91. (a) Segmentation results without the parameter optimization. (b) Final segmentation results with the parameter optimization.

Two types of errors are visible in the results of column (a) when compared to the ground truth: holes are present in the objects, and parts of shadows still remain visible. Holes in objects can be noted in all results. Shadows are clearly visible in *Group* (Fig. 5 (a)). These errors are, in *Hall* (Fig. 3 (a)) and *Group*, mainly due to the parameters of the shadow detection and removal step, that is W^{RGB}, W^{Inv}, and $\mathbf{T}$. Parts of object, in fact, are misclassified as shadows and holes appear in the objects. In the case of *Highway* (Fig. 4 (a)) , the window size W^{Y} in the change detection step is also critical because objects of different size are present. Vehicules far away from the camera are very small compared to the ones that are approaching the camera.

Column (b) shows that the optimization method provides reliable results and in accordance with those given by the subjective opinion. Most of the misdetected pixels inside the objects are correctly classified in all results. Shadows still present in the object mask are almost completly removed (see Fig. 5 (b)). Moreover, the accuracy of the extracted object contours is not seriously affected. In sequence *Highway* (Fig. 4 (b)), the proposed strategy allows to segment the small objects on the top left corner of the image that were missing in Fig. 4 (a).

4. CONCLUSIONS

In this paper, we presented a general framework for optimization of segmentation algorithms based on intuitive inputs from human operators. This framework is very generic and can be implemented for virtually all the segmentation techniques which require some tuning of paramters. We showed that the results of a given segmentation algorithm could be largely improved by making use of this framework. In the absence of a reliable non-referenced segmentation quality metric, we showed how a reference (ground-truth) based segmentation quality metric can still be effieciently used for both research and exploitation purposes. The proposed framework beside providing optimal sets of segmentation parameters for a given segmentation algorithm, can also serve for

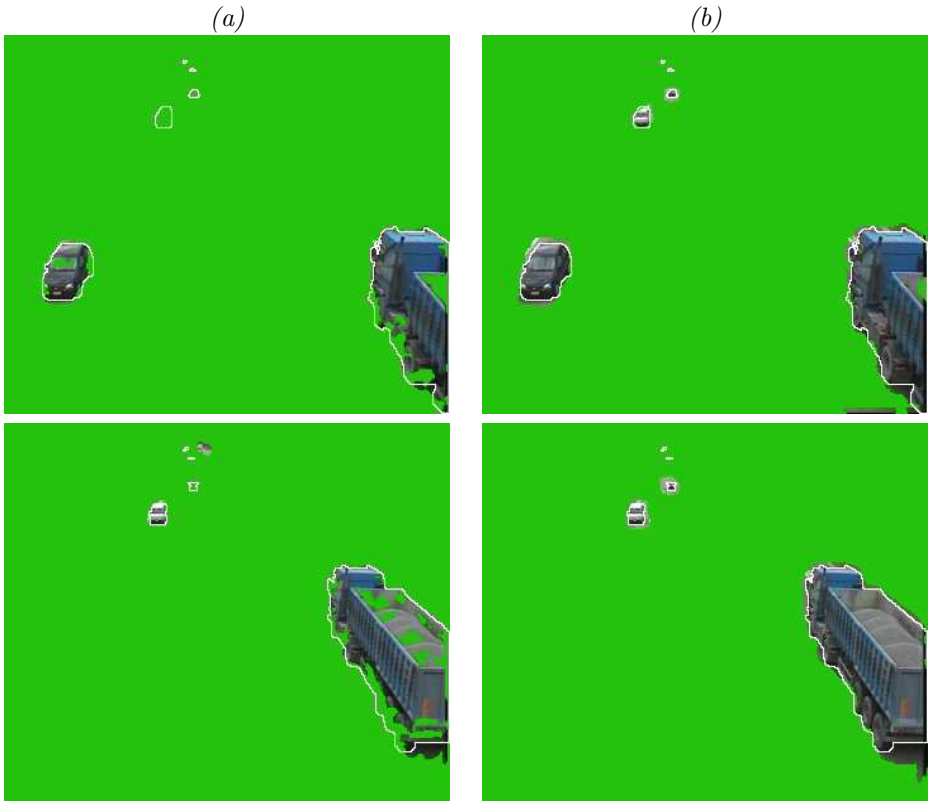

Figure 4. Segmentation results for test sequence *Highway*, at frames #75 and #83. (a) Segmentation results without the parameter optimization (b)Final segmentation results with the parameter optimization.

assessment evaluation purposes. Example of such assessments include, search for the best quality of segmentation that an algorithm can possibly achieve. In addition, the efficiency of different modules in the segmentation component can also be evaluated. An example is that of adaptive threshloding for change detection algorithm used here. Eventhough, in this paper, we chose to use adaptive tuning for some of the parametrs in the segmentation, an alternative would be to optimize the segmentation algorithm by allowing the optimization to change the values of all parameters. This would allow to assess the performance and efficiency of adaptive thresholding approach, but also, would potentially provide enough information in order to design better adaptive thresholding algorithm for some of the parameters.

ACKNOWLEDGMENTS

The authors would like to thank Andrea Cavallaro for providing the change detection software[11] and for the ideas that have contributed to this work.

REFERENCES

1. H. R.M. and S. L.G., *Computer and Robot Vision*, 1992.
2. R. Castagno, T. Ebrahimi, and M. Kunt, "Video segmentation based on multiple features for interactive multimedia applications," **8**, pp. 562–571, 1998.
3. A. Chalom and V. M. Bove, "Segmentation of an image sequence using multi-dimentional image attributes," in *in Proceedings ICIP, Lausanne, Switzerland, Sept 1996*, **2**, pp. 525–528, 1996.
4. P. S. et al, "Segmentaton-based video coding system allowing the manupulation of objects," **7**, pp. 60–74, 1997.

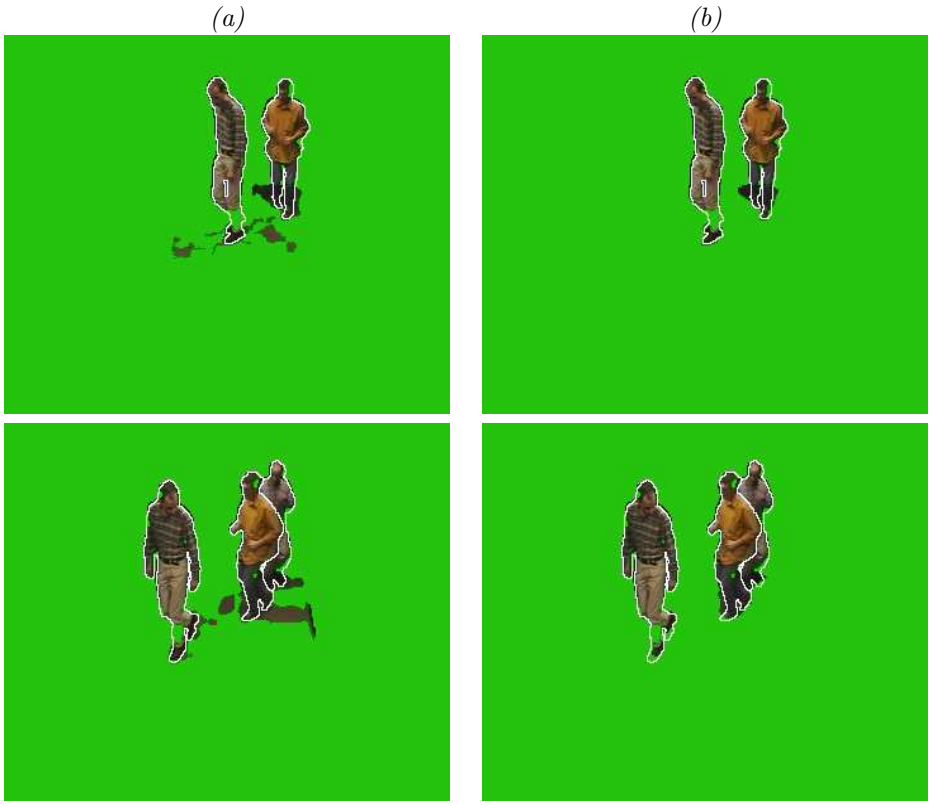

Figure 5. Segmentation results for test sequence *Group*, at frames #83 and #94 . (a) Segmentation results without the parameter optimization (b)Final segmentation results with the parameter optimization.

5. M. Sonka, V. Hlavac, and R. Boyle, *Image Processing: Analysis and Machine Vision*, Addison-Wesley, 1998.
6. Y. J. Zhang, "A survey on evaluation methods for image segmentation," **29**, pp. 1335–1346, 1996.
7. C. Erdem, A. M. Tekalp, and B. Sankur, "Metrics for performance evaluation of video object segmentation and tracking without ground-truth," in *Proc. Int. Conference on Image Processing*, 2001.
8. P.Correia and F. Pereira, "Objective evaluation of relative segmentation quality," in *Proc. Int. Conference on Image Processing*, **2**, pp. 308–311, 2000.
9. X. Marichal and P. Villegas, "Objective evaluation of segmentation masks in video sequences," in *Proc. Of X European Signal Processing Conference*, pp. 2139–2196, 2000.
10. A. Cavallaro, E.Drelie, and T. Ebrahimi, "Objective evaluation of segmentation quality using spatio-temporal context," in *Proc. of IEEE International Conference on Image Processing, Rochester (New York), 22-25 September 2002*, pp. III 301–304, 2002.
11. A. Cavallaro and T. Ebrahimi, "Video object extraction based on adaptive background and statistical change detection," in *Proc. of Visual Communications and Image Processing*, pp. 465–475, 2001.
12. E. Salvador, A. Cavallaro, and T. Ebrahimi, "Spatio-temporal shadow segmentation and tracking," in *SPIE Electronic Imaging 2003, Image and Video Communications and Processing*, 2003.
13. T. Gevers and A. W. M. Smeulders, "Color-based object recognition," **32**, pp. 453–464, 1999.
14. W. H. Press, *Numerical Recipes in C: the Art of Scientific Computing*, Oxford University, New York, 1992.