Detection of Moving Objects in Video Scene – MPEG like Motion Vector vs. Optical Flow

Kunio Takaya
Electrical and Computer Engineering, University of Saskatchewan
57 Campus Drive, Saskatoon, SK. S7N 5A9 Canada
kunio.takaya@usask.ca

Abstract

This paper demonstrates the use of motion vector as defined in MPEG video encoder to detect and crudely segment moving objects in video scene. The proposed motion vector search algorithm incorporated a mechanism to suppress search in the still background and to invalidate the motion vectors found at search boundaries. This paper presents four cases of video clips processed to calculate and identify the MPEG like motion vectors greater than a certain magnitude and to segment out corresponding macro-blocks, which constitute a crude segmentation of moving objects. The MPEG like motion vectors drawn for a consecutive image pair were compared against the vector field calculated by the optical flow method for subjective comparison.

1. Introduction

Tracking moving objects in video image sequences has drawn considerable attentions for various applications such as surveillance, 2.5D structural estimation from motions, video based velocity and position measurement, monitoring production lines, etc. Motion is an attribute specific to video images. Taking advantage of this additional attribute available to video images, objects in motion can be conversely defined in terms of motion. A group of pixels sharing a motion vector of a certain magnitude or direction may be considered as a moving object regardless of other physical attributes. Detection of moving objects in video scene thus depends on how the motion is defined and how algorithm is formulated to calculate the motion, or movement from one video frame to the subsequent frame. There are two major approaches to find the motion vector at a pixel point, a macroscopic approach and a microscopic approach. The macroscopic approach determines the motion in terms of the motion vector as defined the MPEG video codec. Motion vectors of this kind are calculated when video source is encoded by the MPEG encoder into a bit stream for transmission or recording in storage media. Since the motion vector in MPEG is calculated to satisfy minimum of the sum of squared error over a so called macro-block, the motion vector at a pixel point represents a motion of the macro-block rather than the pixel itself. On the other hand, the method of optical flow gives a measure of motion based on the conservation of pixel intensity. When a pixel moves from a location in the current frame to another location in the subsequent frame, the gray scale intensity (or color intensities combined) remains the same. Since the optical flow constraint equation is expressed by the partial derivatives at the pixel location, the optical flow represents a pixel motion in the microscopic sense.

The motion vector field generated by the MPEG like motion vectors and that generated by the method of optical flow are considerably different as expected from the ways that motion vectors are calculated either macroscopically or microscopically. The motion vector field generated by these two approaches are significantly different so that the segmented image of moving objects is expected to be different, because the motion vector field is used to detect and determine the shape of moving objects. In case of the motion field by the optical flow, an approach to minimize an energy function would lead to a tight segmentation with respect to the contour of moving objects [3], [4]. Where the MPEG like motion vectors are used to form a motion vector field, the macro-block that satisfies the segmentation criterion can produce crude segmentations by itself, immediately without additional computational burden. In this paper, the motion vector field generated by the MPEG like motion vectors was compared with the one generated by the optical flow method for four short video clips.

2. Motion Measurement

Motions is defined as a displacement of the object over a period of time. Therefore, motions are usually measured in two consecutive frames of the video image. Both of the motion vector and the optical flow use two consecutive video frames to measure the motion.

Motion Vector Search

Motion vector search is the algorithm developed for, and adopted in MPEG video image coding to search the best match of a local image segment called macro-block. Motion vector search algorithm uses a small patch of image area; 16×16 for gray scale images and 8×8 for color components. The search algorithm searches a macro-block in the target image that best matches the macro-block specified in the reference image. Minimizing the following performance criterion E determines the motion vector (dx, dy). The sum of squared errors

$$E = \sum_{x,y \in M} \sum_{k \in A} \{I_k(x + dx, y + dy.t + dt) - I_k(x, y, t)\}^2$$

is calculated over the macro-block space M, and image attributes A. The displacement vector (dx, dy) that yields the least square error defines the motion vector of the pixel at (x, y). The smoothing effect expected from the macro-block size increases the stability of motion measurements.

There are several motion vector search methods. Those are logarithmic search, exhaustive search, random walk search, and steepest gradient search. In the real time video codec where the time constraint is critical, the logarithmic search or its variation is used. When video data is compressed by the MPEG codec, it needs to keep up with the frame rate of 30 frames/second. In video data compression, whether the minimum reached is one of the local minima or the global minimum makes little difference in terms of the compression gain. Since MPEG encodes the pixel difference between the two macro-blocks, local minima that usually exist in the search area can still give significant compression even though they are not the best. The logarithmic search changes the search range to a half of the previous search stage, so that the global minimum is not guaranteed. Contrary to the case of video data compression, the motion vector to be used for identifying moving objects must take integrity of an object always into consideration. Point-to-point correspondence with respect to the contents of the macro-block is attainable if the global minimum is sought instead of the local minimum. The exhaustive search, which seeks the global minimum within a set search area, works better for moving object detection and identification, but is considerably slow.

When the correspondence of a macro-block between the reference image and the target image is of primary importance, attributes other than the RGB color components normally used in MPEG may improve the global minimum search. Slight change in the ambient light particularly in the outdoor video scenes often misleads the search. In order to make motion vector search robust, the gradients that defines the shapes of objects may be added to the RGB color components.

Optical Flow Method

In contrast to the motion vector search method, the optical flow method is an estimation method of pixel movement. Given the gradient and time derivative at a pixel location, the direction and the magnitude of pixel motion is estimated for individual pixels. Assuming that the intensity of a moving point stays constant throughout time. This induces the optical flow constraint equation,

$$\frac{d}{dt} I(\mathbf{x}, t) = \frac{\partial I}{\partial x} u + \frac{\partial I}{\partial y} v + \frac{\partial I}{\partial t} = \nabla I \cdot \mathbf{v} + I_t = 0$$

Among many algorithms, the Horn and Schunck method and the Lucas and Kanade method are well accepted for the accuracy and robustness. The Horn and Schunck's method [1] finds the optical flow $\mathbf{v} = (u, v)$ which minimizes the error criterion,

$$E(u, v) = \int_D \{(\nabla I \cdot \mathbf{v} + I_t)^2 + \lambda^2(\|\nabla^2 u\|_2 + \|\nabla^2 v\|_2)\} d\mathbf{x}$$

The second term of the Laplacian forces the u and v component of optical flow to have little variation within the small neighbourhood. Therefore, optical flows are smooth across motion boundaries. The Lucus and Kanade method [2] [5] uses the error criterion,

$$E(u, v) = \int_D w(\mathbf{x})(\nabla I \cdot \mathbf{v} + I_t)^2 d\mathbf{x}$$

Where, $w(\mathbf{x})$ is a Gaussian function centered at the pixel point where a optical flow is calculated. Convolution with the Gaussian function $w(\mathbf{x})$ calculates the weighted average of optical flows over the vicinity of the point. The Lucus and Kanade method is known for the high noise robustness. The following summarizes the discrete formulation of this method.

$$
\begin{aligned}
\frac{\partial E}{\partial u} &= \sum_{x,y\in R} w(x,y)\{I_x u + I_y v + I_t\} I_x = 0 \\
\frac{\partial E}{\partial v} &= \sum_{x,y\in R} w(x,y)\{I_x u + I_y v + I_t\} I_y = 0.
\end{aligned}
$$

Let

$$
\begin{aligned}
\mathbf{M} &= \sum_{x,y\in R} w \begin{bmatrix} I_x \\ I_y \end{bmatrix} [I_x, I_y] \\
&= \begin{bmatrix} \sum_{x,y\in R} wI_x^2, & \sum_{x,y\in R} wI_xI_y \\ \sum_{x,y\in R} wI_xI_y, & \sum_{x,y\in R} wI_y^2 \end{bmatrix}
\end{aligned}
$$

$$
\mathbf{b} = \sum_{x,y\in R} wI_t \begin{bmatrix} I_x \\ I_y \end{bmatrix} = \begin{bmatrix} \sum_{x,y\in R} wI_xI_t \\ \sum_{x,y\in R} wI_yI_t \end{bmatrix}
$$

The optical flow is then calculated by

$$
\mathbf{v} = -\mathbf{M}^{-1}\mathbf{b}.
$$

The optical flow is a measure of motion associated with a single pixel. Since the underlying concept is conservation of pixel brightness, the constraint equation of optical flow is an equation of first derivatives. Naturally, underlying assumptions are that pixel motion is slow, motion is not crossing an image boundary, and motion is free from temporal aliasing. Only the algorithm immune to noise can successfully draw a map of optical flows that reflects the motion involved in video images. The Lucus and Kanade method was used in this paper.

3. Implementation of Motion Vector Search

A MPEG like motion vector search algorithm [7] was implemented in a developed program written in Visual Basic. The developed program works on a series of individualized frames. The logarithmic search algorithm was implemented based on the macro-block size of 16×16 pixels of RGB color components. Search range was set to be ± 16 for both $x-$ and $y-$ axes. The search starts from a discrete offset of 8 pixels to test 8 possible locations in the target image to find the minimum of the sum of squared pixel difference. In the next stage, the offset is reduced down to 4 pixels to test 8 possible locations in the vicinity of the location where the minimum was found in the first stage. Then, the discrete offset of 2 pixels, then 1 pixel is used to reach a minimum. Since 8 locations were tested for the minimum at each of the logarithmic search stages of (8, 4, 2, 1) offset, only 32 macro-blocks in the target image are compared to the reference macro-block to reach the most matched macro-block. This is a big time saving compared to 256 comparisons required for the exhaustive search algorithm.

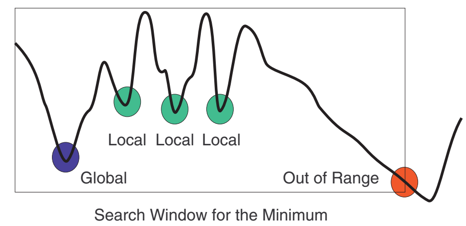

Figure 1. Local, Global and Out of Range Minima

Motion vector search in the program is similar to that used in MPEG video encoding. However, two new functionalities are added for the specific purpose to detect moving objects in the scene.

1. Suppress background

2. Invalidate out of range minima

When no significant motion is observed by comparing the macro-block in the target and that in the reference, this particular pixel is judged to be in the background. When the sum of pixel difference between the two macro-blocks is less than 4.2% of the possible maximum, the pixel from which a macro-block is referred is regarded as background. This figure of 4.2% was experimentally optimized for QCIF video images (320×240). Background check was done prior to motion vector search to save time.

The process to invalidate out of range minima is to ensure that the motion vector found resides inside the search boundary. The search range of ± 16 pixels is adequate for motions appearing in most of the QCIF video clips. Faster motions that go beyond this search range is actually rare. The case indicated by a red circle in Fig. 1 meaning the "Out of range" minimum would happen erroneously due to shading irregularity or image blurring rather than the best match between the macro-blocks. Therefore, validation of the found motion vector is an effective means to obtain reliable results. In the logarithmic search algorithm, either local minima indicated in green or global minimum indicated in blue in Fig. 1 are acceptable. If a found minimum is located on the search boundary, the search result is invalidated to ignore the found motion vector.

4. Results

The developed moving object detection program was applied to various short video clips captured from Canadian TV channels in QCIF format. Individual frames need to be extracted from the video clip recorded in AVI, MPEG or MOV format before processing with the program. Four different video clips, Prime Minister Paul Martin in Fig. 2, a scene of ice hockey in Fig. 3, a scene of two girls diving off the cliff Fig. 4, and a scene of racing cars in Fig. 5, were shown as the results of moving object detection and crude segmentation of moving objects. Fig. 2 is a simple case that only the hands of the Prime Minister Martin are moving. The rest of the scene is recognized as background. Fig. 3 is a scene of dynamic motion of the hockey players. However, the detected motion is less than actual players' motion, because the camera is tracking the hockey players. Because of the camera motion, the stationary posts holding the protection plexiglass are recognized as moving. Fig. 4 is detecting local motions of the non-rigid objects. The high lighted images in the third column are the logical OR (union) of all macro-blocks identified to have a motion vector greater than a threshold. Invalidated motion vectors (marked by red dots) are seen in the fast moving left arm of the girl in the left. Fig. 5 is another example of motions compensated by the camera motion tracking the racing cars. The motion of the lane dividers shows more clearly than the motions associated with the racing cars.

The right most columns of Fig. 2 through Fig. 5 show optical flows calculated for each of the video clips. General appearance of motion vectors are quite different from that of optical flows. Relatively good match is seen in Fig. 2 which has a almost stationary still background. High sensitivity to microscopic local motion at the arm area of Fig. 4 made the global motions of the two girls undetectable. Fig. 3 and Fig. 5 involve rapid camera motion. The objects recognized as moving are completely different. Motion vectors detected the line dividing lanes while optical flow detected the shadows on the road surface in Fig. 5.

5. Conclusion

This paper demonstrated that the motion vectors readily calculated and available in the MPEG video bit stream can be used to detect moving objects in the video scene. It is also shown that the macro-block which received a large motion vector in magnitude, is already an element of the segmented image by motion vectors. Although union, or logical OR of such macro-blocks does not define a tight boundary of the moving object, it provides a crude zig-zag boundary of the moving objects. Optical flows do not present any obvious boundaries of the moving objects. This difference seems due to the difference between the macroscopic motion vector search by pattern matching and the microscopic estimation of optical flow by the conservation of pixel intensity.

References

[1] Berthold K. P. Horn and Brian G. Schunck, "Determining Optical Flow", Artificial Intelligence Laboratory Memo No. 572, Massachusetts Institute of Technology, April 1980. 2

[2] Bruce Lucas and Takeo Kanade, Proceedings DARPA Image Understanding Workshop, pp. 121-130, April 1981. 2

[3] Chenyang Xu, Jerry L. Prince "Snakes, Shape and Gradient Vector Flow", IEEE Trans. on Image Processing, Vol. 7, No. 3, pp. 359-369, March 1998. 1

[4] Daniel Cremers, "A Variational Framework for Image Segmentation Combining Motion Estimation and Shape Regularization", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Madison, Wisconsin, June 2003. 1

[5] A.Bruhn, J.Weickert, C.Feddern, T.Kohlberger, and C.Schnorr "Real-Time Optic Flow Computation with Variational Methods" In Computer Analysis of Images and Patterns, Proc. 10th International Conference CAIP 2003, Groningen, The Netherlands, August 2003 - N.Petkov, M.Westenberg (Eds.) Lecture Notes in Computer Science, Vol.2756, Springer, Berlin, 222-229 2

[6] Joerg Heuer and Andre Kaup, "Global Motion Estimation in Image Sequences Using Robust Motion Vector Field Segmentation", Proc. ACM Multimedia 99, pp. 261-264, Orlando, Florida, Oct. 30 - Nov. 5, 1999.

[7] Rob Koenen, "Overview of the MPEG-4 Standard", International Organization for Standardization, ISO/IEC JTC1/SC29/WG11 N3342, March 2000 3

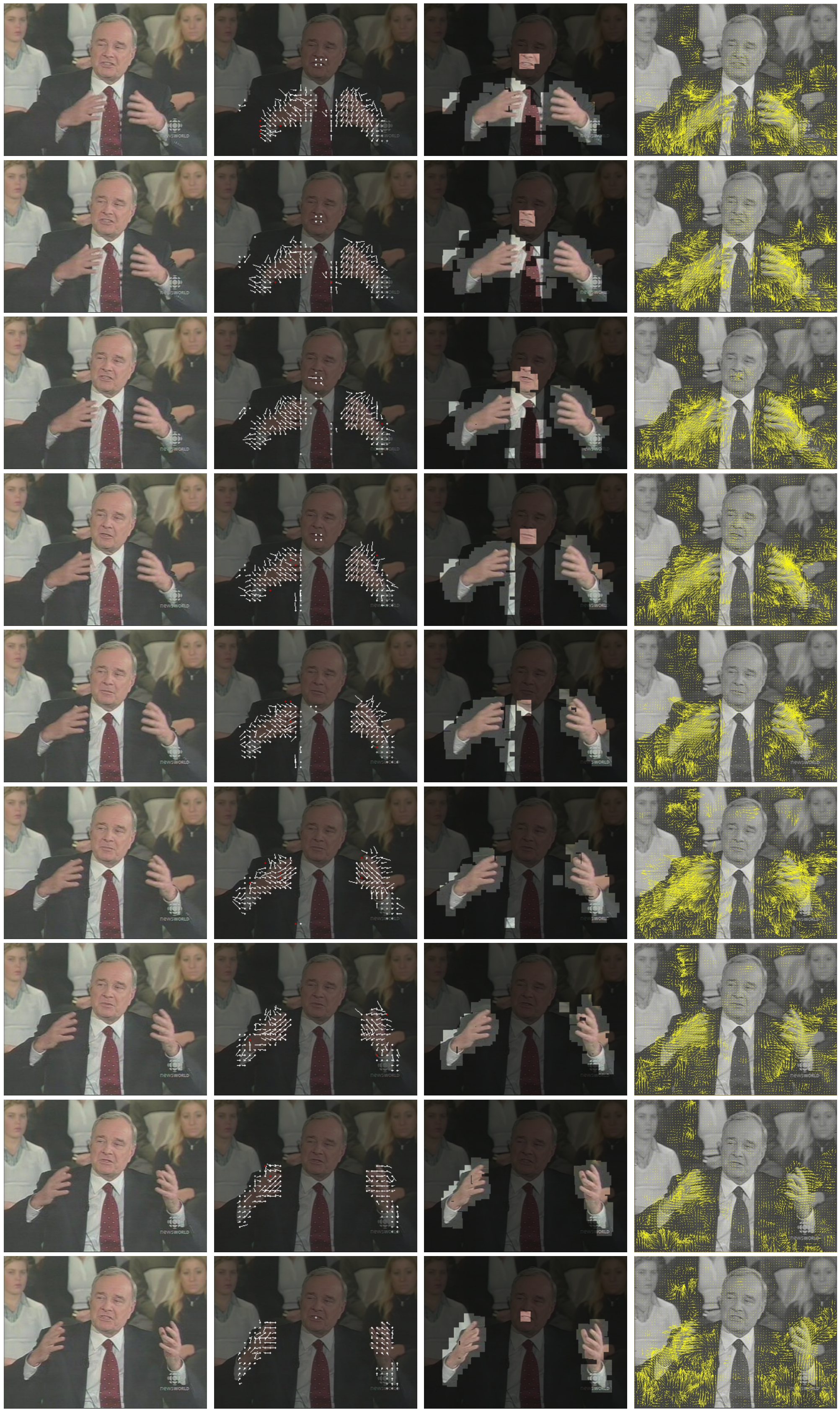

Figure 2. A video sequence of Priminister Paul Martin, original (left), motion vectors superimposed (2nd col.), crude motion segmentation (3rd col.) and optical flows superimposed (right).

Figure 3. A video sequence of an ice hockey game, original (left), motion vectors superimposed (2nd col.), crude motion segmentation (3rd col.) and optical flows superimposed (right).

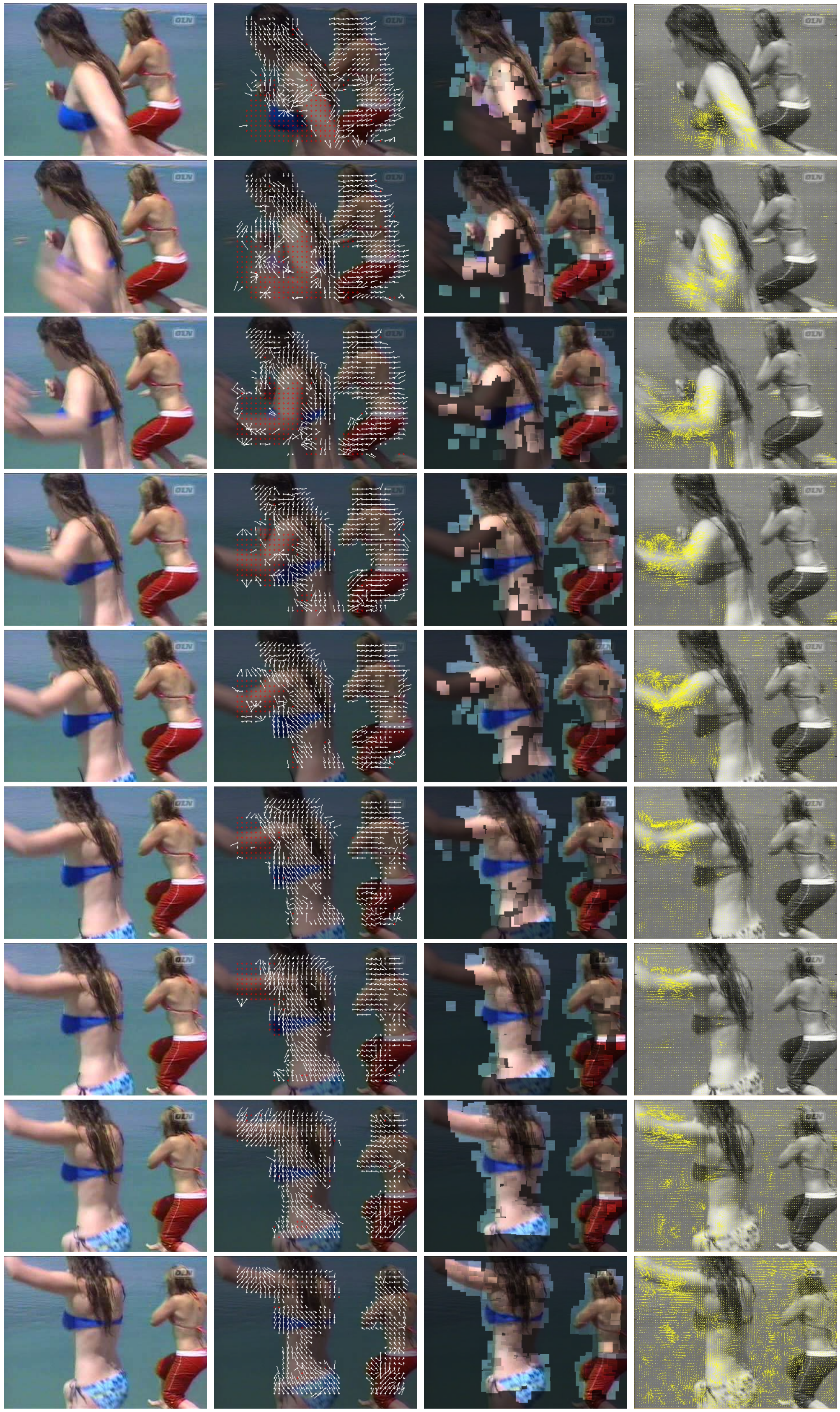

Figure 4. A video sequence of diving off the cliff, original (left), motion vectors superimposed (2nd col.), crude motion segmentation (3rd col.) and optical flows superimposed (right).

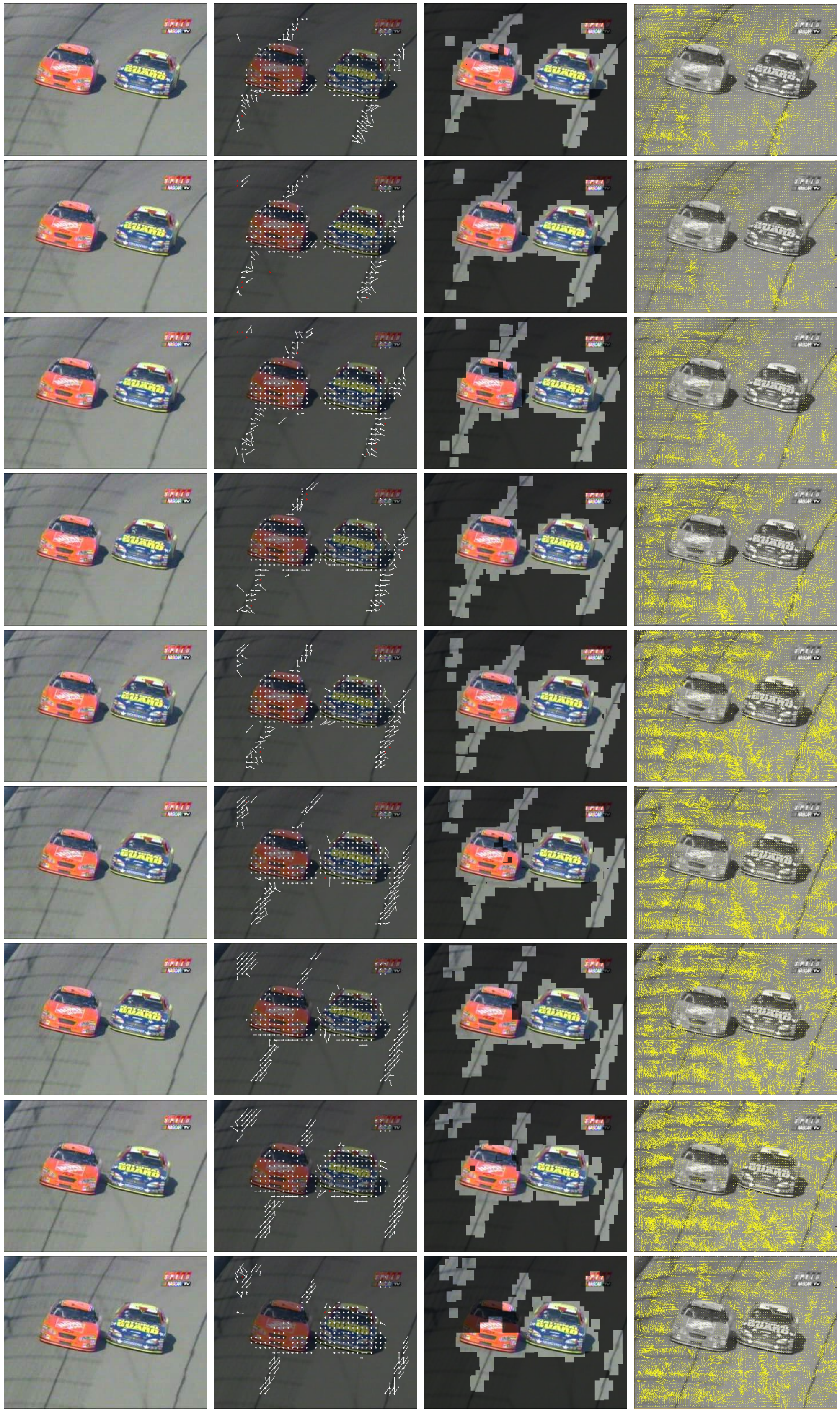

Figure 5. A video sequence of racing cars, original (left), motion vectors superimposed (2nd col.), crude motion segmentation (3rd col.) and optical flows superimposed (right).