Effective Shadow Detection in Traffic Monitoring Applications

Alessandro Bevilacqua, Member, IEEE
ARCES–DEIS (Department of Electronics, Computer Science and Systems)
University of Bologna, Viale Risorgimento, 2
ITALY (40136) Bologna
abevilacqua@deis.unibo.it

ABSTRACT

This paper presents work we have done in detecting moving shadows in the context of an outdoor traffic scene for visual surveillance purposes. The algorithm just exploits some foreground photometric properties concerning shadows. The input of the system is constituted by the blobs previously detected and by the division image between the current frame and the background of the scene. The method proposed is essentially based on multi-gradient operations applied on the division image which aim to discover the *most likely* shadow regions. Further on, the subsequent "smart" binary edge matching we devised is performed on each blob's boundary and permits to effectively discard those regions inside the blob which are either too far from the boundary or too small. We demonstrate the effectiveness of our method by using a gray level sequence taken from a sunny, daytime, traffic scene. Since no a priori knowledge is used in order to detect, and remove, shadows, this method represents one of the most general purpose systems to date for detecting outdoor shadows.

Keywords

binary edge matching, gradient analysis, shadows detection, motion analysis, motion detection, visual surveillance, background difference, image division, traffic monitoring.

1. INTRODUCTION

A common problem that one could encounter in motion estimation of daytime outdoor scenes is that of the detection of shadows attached to their respective moving objects. The detection of a shadow as a legitimate moving region may create confusion for the subsequent phases of motion analysis and tracking. Therefore, it is more than desirable to separate moving objects from their shadow. In order to develop a general purpose, yet effective, system, we use the least a priori information as possible. For example, we do not use color information (which could not be available in case of low light conditions) but only exploit some general properties concerning shadows. In addition, our test sequence shows a high depth of field which makes the task of shadow removal challenging. In fact, the depth of field could alter considerably the appearance of the same object (hence, of the same shadow) through the sequence, according to its position within frames (e.g. the area of a shadow ranges from few tens pixels up to thousands of pixels).

As far, in most of the scenes used to test existing methods reported in other research works, shadows are just outlined, and this due to cloudy weather. By the same way, often the systems coping successfully with sequences showing a high depth of field only deal with aerial images, where the depth of field does not produce any change in the target's shape, or appearance, through a sequence. A few other systems dealing with a high depth of field either fail in detecting distant objects as they become small or detect objects with both bad resolution and definition. Actually, most of the systems which have tried to deal with the above problems use prior knowledge, for example regarding with ground plane constraints, the shadow's direction or shape models. Although the systems using a priori knowledge have achieved some good results, they can always work limited in a particular environment and cannot adjust successfully to great changes either within the environment (except for the ones due to lighting

Journal of WSCG, Vol.11, No.1., ISSN 1213-6972
WSCG'2003, February 3-7, 2003, Plzen, Czech Republic.

conditions) or in the filming modality (such as the depth of field).

The method we setup deals with moving shadow detection and removal, stemming from moving regions (or *blobs*, made of objects with their attached shadows) previously detected ([Bev01a], [Bev02a]). Operations basically rely on the assumption that shadows also from different objects keep some properties across frames unchanged. The gradient-based technique we applied jointly to an effective *false positive reduction (FPR) step* we devised are suited to remove shadows where their *umbra* (a quite uniform region) is relevant. Besides, this work represents one of the most general systems to date for detecting outdoor shadows and it is consequently able to work on scenes coming from different perspectives and having shadows which can even be very marked.

In the next section, we review some other works dealing with shadows. In Section 3 the problem of shadow identification is treated. Our technique for moving shadow detection and removal is then thoroughly explained in Section 4. Finally, extensive experimental results are shown in Section 5 and Section 6 draws conclusions and future works.

2. PREVIOUS WORKS

Traffic monitoring, as well as visual surveillance applications, mostly rely in their first processing step on some kind of moving object detection. Whether these applications deal with traffic monitoring, a building's entrance or parking lots surveillance, only a few of them can successfully cope with the problem of moving shadows originated from the sun light. In fact, many outdoor applications assume that shadows in the sequences have been accounted for prior to their processing. Thus, we see a large number of works with experimental data devoid of shadows.

In the rough algorithm applied in [Scan90] a block-wise mean of the image is computed and stored in an array. Further, its median value is calculated. Pixels belonging to the blocks whose mean is less than the median value are considered as shadows and scaled to the median value by means of an iterative process. In this way, at the end of the process all the shadows should have been removed and the moving objects should be left unprocessed. Experiments show good results for monochromatic aerial images but only with regard to static objects.

The algorithm developed in [Rosin95] works on grey level images taken by a stationary camera. Firstly, authors perform background subtraction and setup automatic thresholding methods in order to segment foreground regions (or *blobs*, a sort of coherent connected regions, sharing common features). Secondly, the intensity ratio image between the current and the reference image is calculated for each pixel within the detected blobs. Besides, authors speculate on the photometric properties of the regions with shadows in the image division. They argue that the photometric gain with respect to the background image is less than the unity and roughly constant over the whole shadow region, except at the edges (*penumbra* region). Furthermore, authors state that objects in the scene which show similar properties are rare "accidents" in a sequence, thus not requiring any FPR step. A region growing algorithm is used to build likely shadow regions. After that, in the first instance shadow regions are selected on the basis that they should contain relatively homogeneous intensity ratio values. In addition, these values should always be less than unity. A more accurate selection is obtained by thresholding on the ratio of the boundary shared with the background against the total boundary length. Furthermore, a ratio between the area of each of the directly bordering regions and of the shadow is computed. By thresholding this value, shadows are definitely identified. Experiments have been accomplished by using only a few human figures in outdoor environments and a low depth of field. At last, some of their a priori assumptions regarding with shadow identification rules yield to detect only shadows with a quite large area with respect to the objects itself.

The algorithm outlined in [Horp99] deals with colour sequences grabbed by a stationary camera. First, the difference between the background image and the current image has been decomposed into brightness and chromaticity components (based on the human visual system). Further on, a suitable threshold is applied on the separate components. This yields a pixel classification into background, shadow or foreground categories. Experiments have been made for both indoor and outdoor scenes, with only one pedestrian.

The strategy in [Staud99] is applied to grey level sequences taken with a stationary camera. Authors use four assumption to detect shadows. First, image regions changed by moving cast shadows will be detected by a large frame difference. Second, the above regions are detected through static edges. At last, background is plane: this yields illumination changes due to moving cast shadows to be smooth. Illumination changes are measured directly from two frames using a physics-based signal model of the appearance of a shadow rather than by comparing two scenes one with and the other without shadows. Authors prepare two distinct modules to detect penumbra and shadows where penumbra is

neglected, respectively. The first module uses the two frame difference between subsequent frames as the input image. A linear luminance edge model is applied in order to detect likely shadow boundaries. For further refinements, a Sobel operator is measured perpendicularly to the borders and the results are thresholded using both the gradient outcome and the edge model. The second method computes the ratio between two subsequent images and thresholds on the local variance. The examples shown concern indoor environments with a static background and one moving person. This algorithm should be heavily adjusted as to work into outdoor scenes as well.

In the method presented in [Kaew01] to adapt the background to changes authors use a colour model similar to the one proposed in [Horp99]: if the difference in both chromatic and brightness components are within some thresholds, the pixel is considered as a shadow. The scenes used for the experiments show a high depth of field and blobs as well as shadows are small. Because of this reason it is difficult to appreciate the effectiveness of the method the authors developed.

The object recognition approach of the aerial traffic surveillance system achieved in [Zhao01] deals with cars and trucks aligned with road direction by using a 3-D model. Authors consider two sides of the rectangular outer boundary of the shadows and the intensity of the shadow area as features. Anyway, these are *ill-posed* features, since small changes in shape may result in large changes in value, due to the high depth of field.

3. SHADOW IDENTIFICATION

Recognizing shadows in a scene is generally a hard task. A person can reasonably recognize a shadow once the scene-geometry and the characterization of light throughout the scene is known. We must transfer this knowledge to the machine so that it can also confidently detect shadows.

This work on recognizing shadows began by asking us: "What can an observer achieve in detecting shadows from a sunny outdoor traffic scene?". Let's try to analyse some properties related to shadows. They result from the obstruction of light coming from a source of illumination. Therefore, we can identify two components: one geometric, the other photometric. The geometry of a shadow is mainly determined by the nature of the obstruction, even though the relative position of objects in a scene could be important. Photometric properties derive from the comparison of light intensity over the same area with and without obstruction.

Another important consideration concerns the partial obstruction of the light source. In daytime outdoor environments, the outer portion of a shadow results from the partial obstruction of the sun. This portion, lighter than the inner side, is a sort of *penumbra*. The inner side of a shadow represents the umbra, where the sun is completely obstructed. Our algorithm mainly addresses shadows where penumbra is almost negligible with respect to the umbra region.

We can distinguish two kinds of shadows: *self* (or *attached*) *shadows* and *cast shadows*. In attached shadows, there is no "free" space between the obstruction and the shadows. It is the opposite for cast shadows. Here we always refer to moving self shadows, without giving relevance to their shape.

Many algorithms detecting shadows take into account a priori information, such as the geometry of the scene or of the moving objects and the location of the light source. We aim to avoid using such a knowledge in detecting shadows. Nevertheless, we exploit the following sources of information:

- as said above, moving shadows in each frame are attached to their respective obstruction object for the most time - this involves spatial information;
- *transparency*: a shadow always makes the region it covers darker - this involves the appearance of single pixels;
- *homogeneity*: researches in [Rosin95] state that the ratio between pixels when illuminated and the same pixels under shadows can be roughly linear - this also involves the appearance of single pixels.

These three criteria are combined by heuristic rules resulting in a binary mask indicating image regions changed by moving shadows.

4. THE ALGORITHM

The algorithm outline is described in Fig.1. All the operations refer to those parts of frames corresponding to previously detected blobs. Namely, before applying this algorithm the previous step consists in detecting moving blobs made of objects with self shadows. Therefore, the algorithm described in the present paper allows detecting the region of each blob corresponding to a shadow in order to further subtract it to the whole blob.

The first step starts by applying a denoising (smoothing) operator both to the background reference image $B(x)$ [Bev02b] generated starting from a cluttered scene (Fig.2) and to the current frame $F(x)$ (Fig.3). After a subsequent image division operation $D(x)=B(x)/F(x)$, the outcome is smoothed in its turn and multiplied by a prefixed factor k. Here $D(x)$ points out this last outcome. Some researches

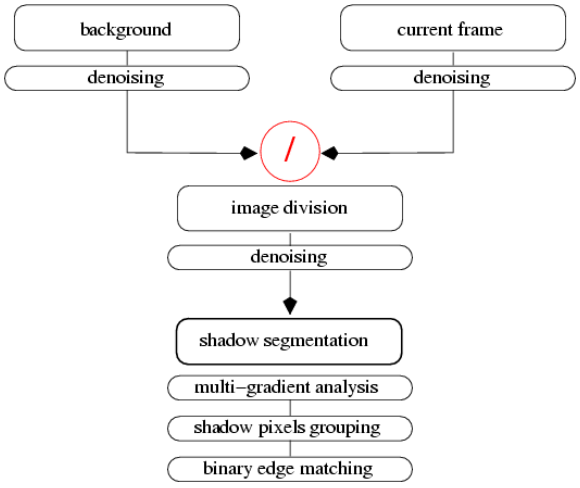

Figure 1. General scheme for the shadow detection algorithm

(e.g. Rosin[95]) state that the ratio between pixels when illuminated and the same pixels can be *roughly* linear. We experimentally found this value ranging from 1 to 2.5. Multiplying the image D by k allows increasing the scale sensitivity of the results in order to make the further threshold operation more reliable

Figure 2. The background scene extracted from the test sequence

Figure 3. A sample (current) frame extracted from the test sequence

and easy to perform. This factor value is not sensitive and actually k has empirically set to *50*. We do not show the division image since it is useful only for computation purposes. Nevertheless, it highlights the homogeneity feature of shadows.

Multi-Gradient Analysis

Actually, before starting the multi-gradient analysis; a *relaxed* threshold operation is performed directly on the division image (Eq.1) within the area defined by the blobs previously detected by a motion detection algorithm. By studying the histograms of intensity values for cars freed of shadows, shadows alone and blobs made of car with shadow, we saw that intensities values for shadows are rather well defined ranging from T_m=*50* to T_M=*80*. However, even though the classes considered are not separable at all and this thresholding operation is error prone whatever the threshold is, it allows selecting *likely* shadow regions *R(x)*:

$$R(x) = \begin{cases} 1 & \textbf{if} \quad T_m < D(x) < T_M \\ 0 & \textbf{else} \end{cases} \qquad (1)$$

Now let us define $S = \{x \in D \mid R(x) = 1\}$. Further on, in those regions defined by $S \subset D$ we calculate multiple gradients in order to find more and more uniform regions. In fact, gradient operations enhance regions with some contrast and let pixel values in uniform regions almost unchanged. For this purpose we use the masks in Fig.4, which define first derivative filters. Sometimes, for computational simplicity the magnitude $|G_i|$ is computed as:

$$|G_i| = |Gx_i| + |Gy_i| \qquad (2)$$

where *i=1* or *i=2*. Masks $|Gx_i|$ in (a) and (b) are able to detect vertical edges while $|Gy_i|$ are sensible to

Figure 4. Convolution kernels for vertical (Gx_1,Gx_2) and horizontal (Gy_1,Gy_2) edge detection

horizontal edges. We use them according to the Eq.2, thus obtaining two couples. When coupled (Fig.4 (a) and (b)), these masks allow detecting edges in horizontal, vertical and oblique directions as well. The shadows we cope with stem from the sun light at the left of the camera. Therefore, all of them are oriented from left to right. It is worth remarking that we do not want to set up a method that works at best only for this kind of shadows. Therefore, in spite of a sort of redundancy, the masks of Fig.4 yield gradients acting in a slightly different way mainly for

oblique edges. As a matter of fact, the couple (a) detect top-left oblique edges better than the couple (b) and the latter is better for detecting top right edges. Since final result improves when using both (a) and (b), we keep both the couples.

In addition, we also compute the Robert Cross gradient by using the masks of Fig.5. It has been shown that it responds best to diagonal edges, rather than to vertical and horizontal. The operator estimates the total gradient by summing it in the diagonal directions. The filter is implemented as the addition of the two convolutions with masks Gd_1 and Gd_2. We then perform three hard thresholding operations (Eq.3), one for each gradient *g*, that yields

+1	0
0	−1

Gd_1

0	+1
−1	0

Gd_2

Figure 5. Roberts Cross convolution kernels

the ultimate binary image *G* of Fig.6, regarding with this first stage:

$$G(x) = \begin{cases} 1 & \textbf{if} \quad \exists g | S(x) < T_g \\ 0 & \textbf{else} \end{cases} \qquad (3)$$

where T_g is the threshold value for the gradient type *g*. Fig.6 shows the outcome of the thresholding performed on the smoothed image *D* and on the three gradients described above. Actually, this multiple threshold operation is the same as to perform OR thresholding. In fact, having *at least* one direction for which the gradient value is bounded has been considered to be enough. In fact, we could have performed an AND operation in order to reduce the amount of false positive signals. Nevertheless, we would also have reduced the amount of true signals, mainly along the shadow borders. We want to stress that one of our main goals is not to loose true signals at all. Further tasks will take care of reducing false signals.

Figure 6. Thresholding on the smoothed image *D* and on the three gradients yields these *likely* shadow pixels

Shadow Pixel Grouping

Fig.6 shows a lot of not connected regions, but the previous threshold operations have preserved mainly coarse areas. Hence, morphological operations performed on *G* give good results, as it was predictable (Fig.7). Here we want to remark that this

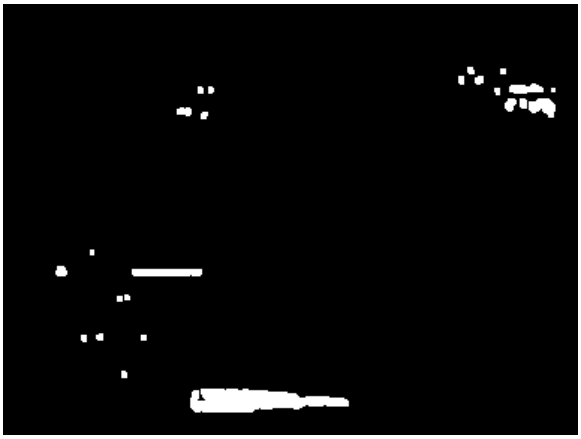

Figure 7. *likely* shadow regions after they have been segmented into blobs by means of morphological operations

method aims to find *coarse* uniform regions. Therefore morphological operations (see [Bev02c]) essentially perform a dilate and a connection operation among yet coarse agglomerate of pixels, without putting the attained definition on the first level. Nevertheless, blob's definition keeps high, thanks to the procedure described above.

Binary Edge Matching

Once the likely shadow regions have been connected, it is necessary to find out the *true* shadows. This step is also known as the *False Positive Reduction* (FPR) step. In fact, we must clean those regions that in the previous steps had the same appearance of true shadows. These, usually, are inner parts of blobs, like regions *B*, *C*, *D* of Fig.8, that exhibit same photometric properties of true shadows (region *A*). Therefore, the FPR is essentially based on geometric considerations. All regions are inner to the blobs. Two kinds of regions are discarded: the ones far from the blob's boundary, and the smallest ones. The basic assumption is that a true shadow must be a large shadow-like region near the boundary. Regions removal is accomplished by computing the percentage of the blob's border (red border lines) shared with the boundary of the homogeneous regions just selected (dark blue areas). The percentage *p* has been established to be about *50%* with respect to the total perimeter (red solid and dash lines) length. Namely, the homogeneous regions which share the boundary with the blob's border for more than *50%* are considered to be shadows. However, this value may change during the day.

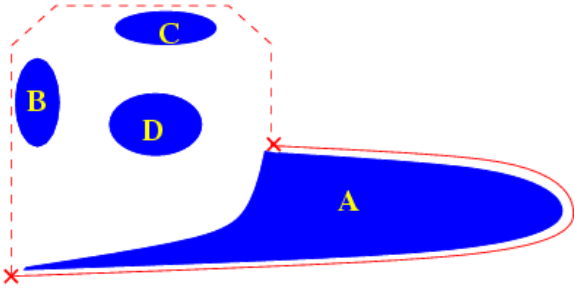

Figure 8. Silhouette of the blob previously detected (red solid and dash line) and likely shadow regions (dark blue areas). The solid line points where the blob's border matches with the shadow's one

Actually, the candidate shadows of Fig.7 are usually a little smaller than their real size, due to the last morphological operations. Therefore, "shared" means "close enough", where in the current implementation the target distance *d* is 5 pixels. Hence, shadows are primarily selected by thresholding on the percentage of the shared borders. By looking at the example of Fig.8, the region *D* is discarded because it lies more than *d* pixels from border. Regions *B* and *C* are excluded because the segment of border near to the dash red line is less than *p*. Only the region *A* will pass this thresholding phase. Shadows are then removed by considering the outer shadow border (red solid line of Fig.8) and following the inner shadow border joining the two cross signs.

The outcome of this method is shown in the image of Fig.9. To conclude, we state that the FPR step does not afflict either absolutely large shadows, like the one on the bottom side of Fig.9, or absolutely small shadows but large enough with respect to the whole blob, like the one on the top right side of Fig.7. The method thus results in detecting shadows, independently from the depth of field, where penumbra is negligible with respect to the umbra region.

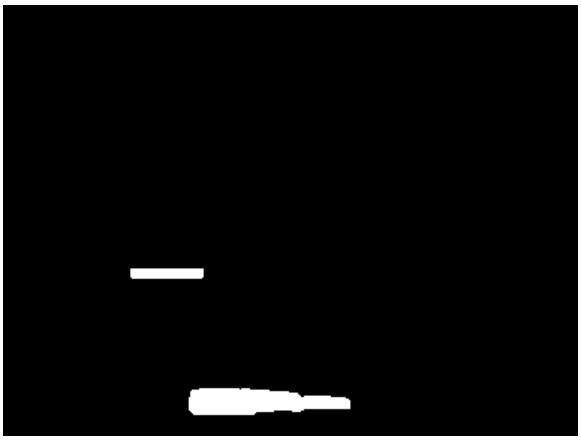

Figure 9. Definite shadow regions after the FPR step

5. EXPERIMENTAL RESULTS

Our study focuses on outdoor grey level image sequences taken by one camera, with a fixed focal length and a high depth of field. Images are 8-bit, with resolution of 384x288. The camera was mounted on a tripod placed on a bridge, so the background is static, even though not completely stationary, because of waving tree phenomena. The scene is quite cluttered: cars, campers, motorcycles and pedestrians are present. Actually, our overall motion detection system is able to detect up to 20 blobs at 4 fps on an entry-level PC. The algorithm has been fully written in ANSI C and works under Windows, Solaris and Linux OS's.

The video used to test the algorithm contains a daytime traffic sequence which has been sampled at 10 Hz and is of 100 frames. In order to better assess the performance of the algorithm, we split our set into five couples (test and training) of equally sized subsets. That is, each set contains 50 frames and the averaged values is finally considered, even though to this analysis we only consider the average result attained on the test sets. Besides, in order to be able to give absolute measures, we did obtain accurate ground-truth in our experiments by manually segmenting the frames of the sequence.

	TOTAL	TRAINING	TEST
blobs	1122	654	468
f-shad	108	64	44
b-shad	88	56	32
NOP_b	977324	709572	267752
NOP_s	95344	75930	19414

Table 1. The ground-truth values related to the whole sequence (second column), to the average of the five training sets (third column) and of the five test sets (fourth column)

Table 1 shows the most relevant information about the ground-truth inherent to the training sets and the test sets we use. The values are reported in terms of *number of entities* (whether they are blobs with shadow or just shadows) and of *number of pixels*. In this last case, the pixels belong either to blobs without shadows (NOP_b) or to shadows only (NOP_s).

As for shadows, we must define two different typology of shadows. *f-shad* ("foreground" shadows) indicate all the moving shadows detected below the nearest (from bottom) pedestrian crossing of Fig.10. Oppositely, all the shadows above that pedestrian crossing are "background" shadows (*b-shad*).

Before analyzing the overall performance of the motion detection algorithm, we briefly outline the measures utilized in the subsequent analysis. Let *H (Hit)* indicate the number of detected objects that really move, *M (Miss)* the number of moving targets that will be classified as non-moving. Let $K=H+M$ be the total number of the actual known objects. Based on the above definitions, we can define $DR=H/K$ (*Detection Rate*) and $MR=M/K$ (*Miss Rate*). In addition, the *number of shadow pixels* is studied. As far, *False Alarm (FA)* has not been considered since most of the wrongly detected shadows lie within the area of the objects (not of the blobs) and there could be many reasons for this blob fragmentation. Therefore, even though at the moment it is quite difficult to find out which regions are erroneously detected as shadows, FA will be soon introduced.

	H	DR	M_a	MR_a	M_i	MR_i
f-shad	43	97.7%	1	2.3%		
b-shad	18	56.3%	14	43.7%		
NOP_s	17136	88.3%	1994	10.2%	284	1.5%

Table 2. Values for the most significant quality parameters related to the number of *f-shad* and *b-shad* and to the number of the overall shadow pixels (NOP_s).

Fig.10 shows a significant output frame where shadows have been removed. Table 2 reports the result the average results for the test sets achieved by the algorithm on the basis of the number of shadows, accordingly to their typology, and to the overall number of shadow pixels. Here, M and MR have been used to indicate two different kinds of missed pixels. M_a and MR_a measure the number of missed pixels that anyway remain attached to the blob, after the shadows have been removed. These do not afflict the overall performance (DR, MR) concerning the number of blobs detected. At most, these missed pixels could worsen the *definition* of blobs. However, the outer shadow pixels sometimes may be not detected. Or more precisely, sometimes the "point" of the shadow (in this sequence, at our right) is not detected. After removing shadows, the point of the shadows may get detached from the object it belongs to thus constituting a FA in terms of a wrongly detected blob. M_i and MR_i measure the missed pixels that remain isolated from the blob.

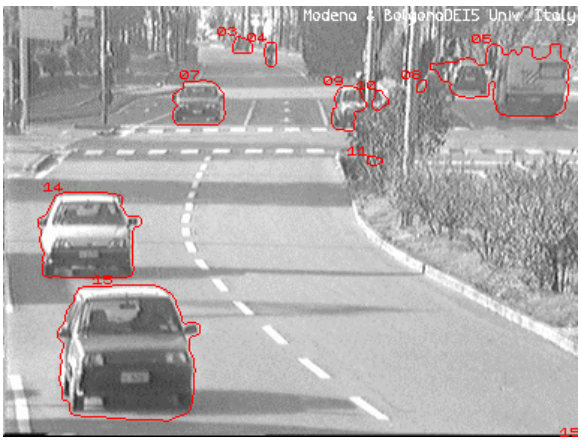

Figure 10. An output frame of the system where the moving objects (freed of shadows) have been contoured

As regards to the number of shadows, we see from Table 2 that *all* of the *f-shad*, but one, are detected (and this removed!) while a large percentage of the thinner *b-shad* is not detected. Basically, the algorithm has been devised quite for *f-shad*, where the penumbra region is negligible with respect to the umbra. In addition, this kind of shadows is easier to detect because they are larger and well defined. The system also detects the *f-shad* which "visually" join different objects (blob ID's 13 and 14 in Fig.11). Frequently, removing a shadow (attached to blob ID 13 in Fig.11) allows one compound blob to correctly separate into different objects, thus increasing the overall performance of the motion detection and tracking system.

As for the detection of *b-shad*, at first sight the outcome attained by the presented method could be considered quite poor. However, they should be yet more appreciated when considering the objective difficulty for those shadows to be detected. In fact, mostly they refer to far away vehicles whose overall shape is not yet fully visible. On the other side, this bad visibility makes sure this loss in shape definition does not cause too heavy a visible consequence.

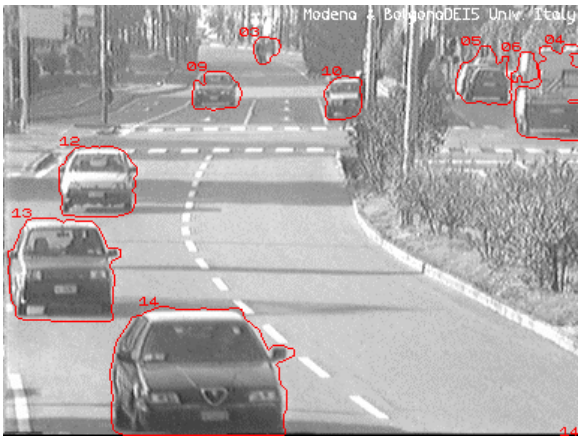

Figure 11. An output frame of the system where removing a shadow (attached to blob ID 13) allows one blob to separate into two distinct objects (freed of shadows) have been contoured

As far as NOP_s is concerned, we can see that a few pixels remain attached to the blobs (10.2%) and still fewer get detached (1.5%). With the area threshold we actually use in the overall motion detection algorithm (50 pixels), MR_i (284) will contribute at most to 5 wrongly detected (FA) blobs. As a matter of fact, these "islands of shadow" appear only for a very few frames and soon disappear. Therefore, we can conclude by stating that the FA introduced by the shadow detection module are practically negligible for the subsequent tracking stage.

6. CONCLUSIONS

A novel method to detect attached shadows has been presented. The original contribution of this work mainly consists of a sequence of quite simple operations (from a computational point of view) which allows obtaining an effective shadow detection method in spite of their simplicity. In particular, the false shadows reduction method, based on binary edge matching, is original by itself.

Besides, this work represents one of the most general systems to date for detecting outdoor shadows. In fact, no color information has been exploited in order to detect them and the sequence used to assess the algorithm's performance also shows a high depth of field.

Experimental results show how the shadows removal module we prepared enhances the performance of the overall detection system, in terms of quality of detection, by entirely removing the shadows where penumbra is negligible. It is also worth remarking how the simplicity of the image processing routines utilized does not lower significantly the overall timing performance of the motion detection system.

As directions for further researches, we are quantifying how much the regions erroneously detected as shadows affect the objects integrity. In addition, a module apt to cope with thin and bad-defined shadows is being developed.

7. ACKNOWLEDGMENTS

We wish to thank Prof. Giorgio Baccarani and Prof. Riccardo Rovatti for their interest in the present work and for useful discussions. We also desire to thank Prof. Luigi Di Stefano for having offered the sequence to study.

8. REFERENCES

[Bev01a] Bevilacqua, A., and Roffilli, R. Robust Denoising and Moving Shadows Detection in Traffic Scenes, in IEEE CVPR 2001 Technical Sketches conf. proc., Kauai, Hawai, pp.1-4, 2001

[Bev02a] Bevilacqua, A. A System for Detecting Motion in Outdoor Environments for a Visual Surveillance Application. PhD thesis, Department of Electronics, Computer Science, Systems, Bologna, Italy, 2002

[Bev02b] Bevilacqua, A. A Novel Background Initialisation Method in Visual Surveillance, in MVA 2002 conf. proc., Nara, Japan, pp.614-617, 2002

[Bev02c] Bevilacqua, A. Effective Object Segmentation in a Traffic Monitoring Application, in ICVGIP 2002 conf. proc., Ahmedabad, India, pp.125-130, 2002

[Scan90] Scanlan, J.M., Chabries, D.M., and Christiansen, R. W. A Shadow Detection and Removal Algorithm for 2-D Images, in IEEE ICASSP 1995 conf. proc., Albuquerque, New Mexico, pp.2057-2060, 1990

[Rosin95] Rosin, P.L., and Ellis, T. Image Difference Threshold Strategies and Shadow Detection, in the 6th BMVC 1995 conf. proc., Birmingham, UK, pp.347-356, 1995

[Staud99] Stauder, J., Mech, R. and Ostermann, J. Detection of Moving Cast Shadows for Object Segmentation. IEEE Transactions on Multimedia, Vol.1(1), pp.65-76, 1999

[Horp99] Horprasert, T., Harwood, D., and Davis, L.S. A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection, in the 7th IEEE ICCV 1999 Frame-rate Workshop conf. proc., Kerkyra, Greece, 1999

[Kaew01] KaewTraKulPong, P., and Bowden, R. An Improved adaptive background mixture model for real-time tracking with shadow detection, in the 2nd AVSS European Workshop proc. conf., Kingston upon Thames, London, UK, Vol.1, pp.149-158, 2001

[Zhao01] Zhao, T., and Nevatia, R. Car Detection in Low Resolution Aerial Image, in the 8th IEEE ICCV conf. proc., Vancouver, BC, Canada, Vol.I, pp.710-717, 2001