Evaluating the Effects of MJPEG Compression on Motion Tracking in Metro Railway Surveillance

Angelo Cozzolino[2], Francesco Flammini[1], Valentina Galli[1], Mariangela Lamberti[1],
Giovanni Poggi[3], and Concetta Pragliola[1]

[1] Ansaldo STS, Via Argine 425, Naples, Italy
{francesco.flammini,valentina.galli,
mariangela.lamberti}@ansaldo-sts.com
[2] Nexera Scpa, Centro Direzionale Isola A/3, Naples, Itlay
acozzolino@nexera.it
[3] University of Naples Federico II, Via Claudio 21, Naples, Italy
giovanni.poggi@unina.it

Abstract. Video content analytics is being increasingly employed for the security surveillance of mass-transit systems. The growing number of cameras, the presence of legacy networks, the limited bandwidth of wireless links, are some of the issues which highlight the importance of evaluating the performance of motion tracking against different levels of video compression. In this paper, we report the results of such an evaluation considering false-negative and false-positive metrics applied to videos captured from cameras installed in a real metro-railway environment. The evaluation methodology is based on the manual generation of the Ground Truth on selected videos at growing levels of MJPEG compression, and on its comparison with the Algorithm Result automatically generated by the Motion Tracker. The computation of reference performance metrics is automated by a tool developed in Matlab. Results are discussed with respect to the main causes of false detections, and hints are provided for further industrial applications.

Keywords: performance evaluation, motion tracking, MJPEG codec, intelligent video surveillance, mass-transit systems.

1 Introduction

Many transit systems can be spread through hundreds of kilometers and require thousands of employees for daily operations. A complete deployment of visual surveillance to cover a system of this magnitude requires thousands of cameras, which makes human-based surveillance unfeasible. Detecting specific activities almost completely relies on costly and scarce human resources. Manual analysis of video is labor intensive, fatiguing, and prone to errors. Additionally, psychophysical research indicates that there are severe limitations in the ability of humans to monitor simultaneous signals. Thus, it is clear that there is a fundamental contradiction between the current surveillance model and human surveillance capabilities. The ability to monitor real-time footage provides dramatic capabilities to transit agencies. Software-aided real-time video content analytics (VCA) considerably alleviates the

J. Blanc-Talon et al. (Eds.): ACIVS 2012, LNCS 7517, pp. 142–154, 2012.

human constraints, which currently are the main handicap for analyzing continuous surveillance data [4].

Past experiences (see e.g. [11] for Madrid Metro) using state-of-the-art systems reported poor performance, with up to 1700 false alarms per camera per day, literally overwhelming central operators. Those results lead to the conclusion that the video-analytics technology was not yet mature to be adopted in real mass-transit environments. Our experience proved instead that, though very initial results can be disappointing, after careful testing and optimization a huge improvement could be achieved, making the technology usable in practice. In fact, we developed and succesfully adopted in real installations (e.g. Metrocampania Nord-Est [3]; see Fig. 1) a methodology based on rigorous testing procedures for ‘black-box’ performance evaluation of VCA in the specific contexts (camera type and position, scene, external noise, weather, indoor/outdoor surrounding environment, etc.). Hence, key parameters of the algorithms are modified according to the results of performance assessment, and the evaluation is repeated until satisfactory results are achieved. In order to speed-up the process, those parameters (including area of interest, size/speed of the objects, alarm latencies, inhibition times, etc.) are grouped considering camera categories which are homogenous in terms of installation and lighting conditions (e.g. platform cameras, tunnel cameras, etc.). In a few iterations, the fine-tuning methodology converges to the optimal trade-off between false/nuisance alarm rate and detection probability.

In this paper we present a method to go a step forward with respect to our previous experience: the aim here is to evaluate the performance of the motion tracker without using filters on the speed and size of the objects, like we did when performing black-box testing. In such a way, the tracking of any object of any size is taken into account and therefore it is possible to investigate more precisely on the causes of false detections (positive or negative). However, that obliges to employ lower level metrics, that we borrowed from the past research in this field. Furthermore, we wanted to evaluate the performance of motion tracking with respect to the MJPEG video compression, which is notoriously much less efficient than more recent codecs like H264 [9], but still widespread especially in legacy installations. The results we achieved allow to fine tune the compression level to obtain the optimal trade-off between bandwidth occupation and VCA performance, when video quality is not required to be 100% (e.g. because of redundant coverage with other standard, megapixel or PTZ cameras).

Fig. 1. Control room for the security management system

2 Reference Metrics

Several metrics have been proposed in literature to evaluate VCA performance. Those metrics usually require a comparison of the Algorithm Result (AR) with optimal results stored in the so called Ground Truth (GT) [1]. Therefore the first step towards performance evaluation is to build a valid Ground Truth [8].

In this paper, the method used for ground truthing is one in which objects are manually bounded by geometric shapes, typically rectangles; unique IDs are assigned to individual objects and are consistently maintained over subsequent frames.

A variety of annotation tools exist to generate GT data manually, such as Anvil, VideoAnnex, ViPER-GT. Though more time-consuming with respect to possible automatic or semi-automatic methods, manually generated GT are obviously more reliable; this is the reason why this is still the most widespread method for GT generation. Hence, a typical GT consists of a text file including information about the labels and the coordinates of the bounding boxes for each object present in the scene. In order to automate the evaluation of metrics, GT and AR files should include coherent information.

The tracking method has been used for the definition of reference metrics. Tracking is defined as the problem of estimating the spatial extent of the non-background objects for each frame of a video sequence. The result of tracking is a set of tracks for all non-background objects [14]. Tracking is based on the fact that objects are present in the scene for a certain time frame; hence, objects can be characterized spatially by their positioning information (i.e. the up-left and down-right coordinates of their bounding box) and temporally by the number of frames in which they are present, that is their track.

To quantify the level of matching between GT and AR tracks, both in space and time, it is necessary to define the concepts of spatial and temporal overlap between tracks. The spatial overlap is the bounding box overlapping *A(Gi, ARj)* between *Gi* and *ARj* tracks in a specific frame *k*:

$$A(G_{ik}, AR_{jk}) = \frac{Area(G_{ik} \cap AR_{jk})}{Area(G_{ik} \cup AR_{jk})} \tag{1}$$

The temporal overlap associates AR tracks to GT tracks according to the following condition in order to find candidates for GT and AR tracks association:

$$\frac{L(G_i \cap AR_j)}{L(G_i)} \geq T_{ot} \tag{2}$$

where *L(Gi ∩ Ai)* is the number of frames of the intersection between GT track *i* and AR track *j*, *L(Gi)* is the number of frames of GT track *i* and *Tot* is an appropriate threshold.

In the following we introduce the basic metrics used in this paper to evaluate the performance of the motion tracker.

- ***False Negative (FN) o Track Detection Failure (TDF)***: a GT track will not be considered detected (i.e. track detection failure), if it satisfies any of the following conditions.

  1) a GT track *i* has temporal overlap smaller than *Tot* with any AR track *j*:

$$\frac{L(G_i \cap AR_j)}{L(G_i)} < T_{ot} \qquad \forall j \tag{3}$$

  2) although a GT track *i* has enough temporal overlap with AR track *j*, it has insufficient spatial overlap with any AR tracks (smaller than *Tos*):

$$\frac{\sum_{k=1}^{N} A(G_{ik}, AR_{jk})}{N} < T_{OS} \qquad \forall j \tag{4}$$

- ***False Positive (FP) o False Alarm Track (FAT)***: an AR track will be not associated with any GT tracks (i.e. false alarm), if the AR track meets any of the following conditions:

  1) a AR track *j* has temporal overlap smaller than *Tot* with any GT track *i*:

$$\frac{L(G_i \cap AR_j)}{L(AR_i)} < T_{ot} \qquad \forall i \tag{5}$$

  2) a AR track *j* does not have sufficient spatial overlap with any GT track *i* although it has enough temporal overlap with GT track *i*:

$$\frac{\sum_{k=1}^{N} A(G_{ik}, AR_{jk})}{N} < T_{OS} \qquad \forall i \tag{6}$$

The above listed metrics have been validated in previous studies on performace evaluation of artificial vision using publicly available datasets (like PETS, i-LIDS, ETISEO, etc.).

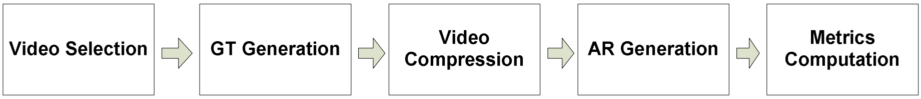


Fig. 2. Steps of the evaluation method

3 Evaluation Method

The main steps of the method used for performance evaluation are shown in Fig. 2.

In the first step (i.e. 'Video Selection') we had to collect a set of uncompressed video recordings from cameras representing a comprehensive picture of the main metro railway scenarios, that is:

- Concourse
- Platform
- Turnstiles
- Tunnel portal

Those scenarios are very diverse, going from possibly crowded areas featuring quick movements (especially near turnstiles) or almost stationary conditions (mainly in platform), to situations (i.e. tunnel portals) in which human presence is rare but false alarms can be generated by the light change of passing trains. More specifically, real footage has been selected for the duration of 1 minute, featuring:

- Concourse, 7 objects in the scene;
- Platform, simulation of object left behind;
- Turnstile, 7 objects in the scene;
- Tunnel portal, train passing.

All cameras are analogue featuring 4CIF resolution and 25FPS. The camera watching tunnel portal features an IR lamp to be able to see in very low light conditions.

In the second step (i.e. 'GT Generation') the GT has been generated using an appropriate criterion for the organisation of the information (ID and box coordinates) in the text file including for each frame the list of manually detected objects.

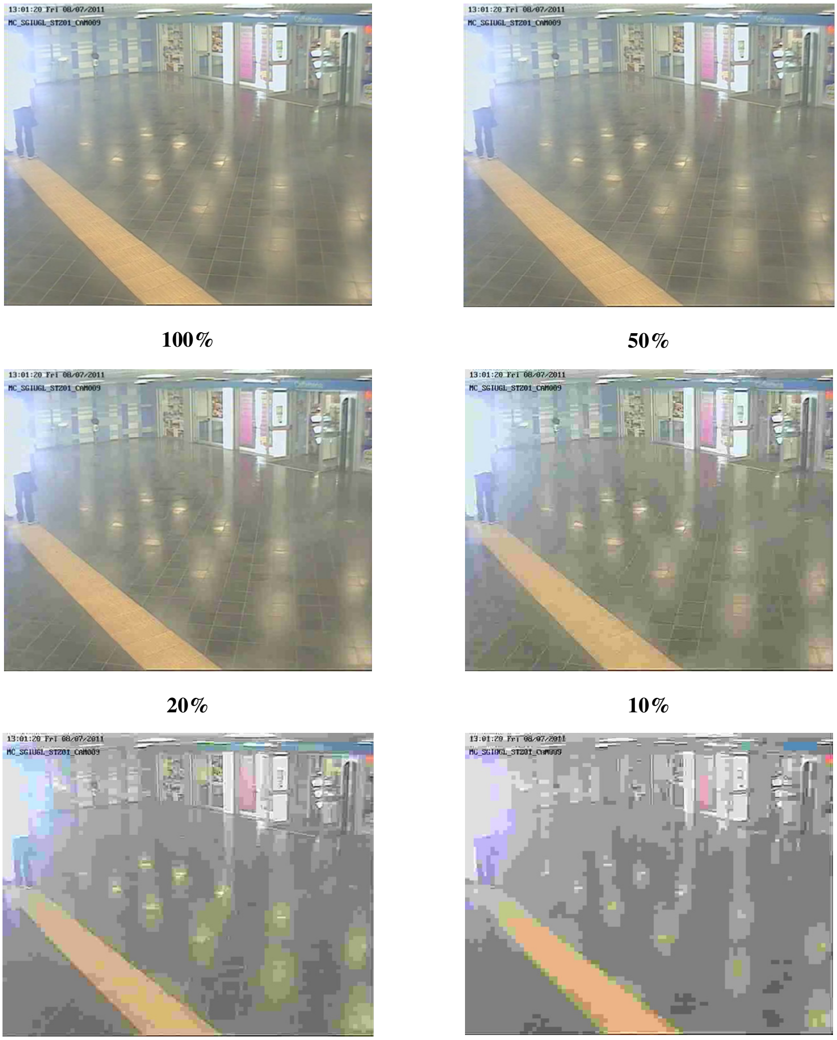
In the third step (i.e. 'Video Compression') the 1500 frames of the selected videos have been MJPEG compressed using the following quality levels (in percentage) and subsequent compression factors ('C'): 100% (C = 1); 50% (C ≈ 5); 20% (C ≈ 10); 10% (C ≈ 15); 5% (C ≈ 20); 1% (C ≈ 25) (see Fig. 3).

In the fourth step ('AR Generation'), videos have been analyzed by a Motion Tracker [10] identical to the one installed in the metro-railway but without using filters for alarm generation. The Motion Tracker has generated for each compression level an AR text file with detected objects, whose information was structured coherently with the ones included in the GT.

In the fifth step ('Metrics Computation'), we have applied the SW tool developed in Matlab to automatically compute the FN and FP metrics introduced in Section 2. The tool organizes its input data (GT and AR) in cell array whose number of rows is equal to the number of objects while the number of columns is 5, that is:

- The list of frames in which the object is present (i.e. the track), that is a vector whose length is equal to the number of frames of the track;
- Top-left and bottom-right coordinates of the bounding-boxes (2 vectors of length 2)

By comparing GT and AR cells, the algorithm computes the FN and FP metrics verifying conditions on temporal and spatial overlaps (*Tot* and *Tos* thresholds), as discussed in Section 2.



100%

50%

20%

10%

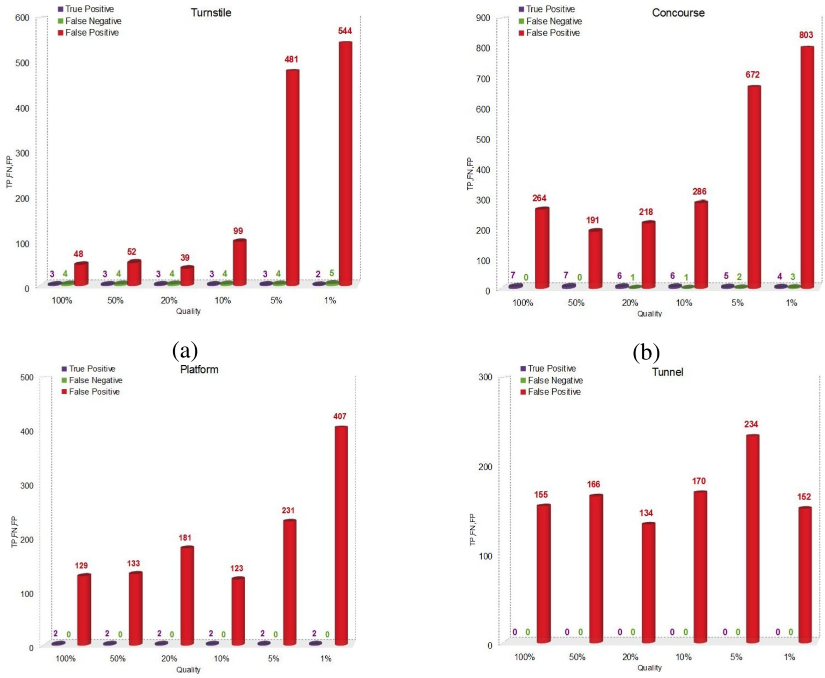
5%

1%

Fig. 3. Example MJPEG compression at different quality levels

4 Discussion of Results

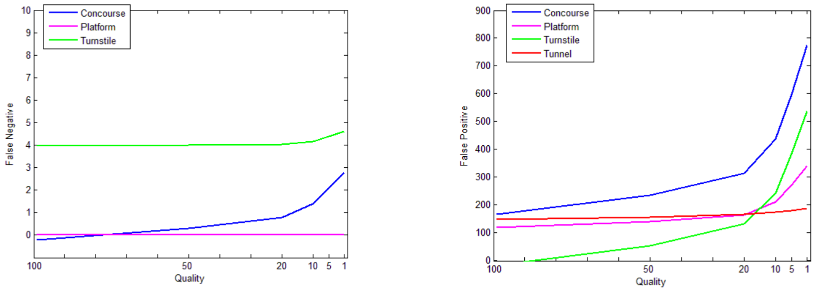
In Fig. 4 we report the numerical results (represented by bar diagrams) of FN and FP evaluation against video compression quality, in the different scenarios, while in Fig. 5 the same results are shown by means of smoothing functions in order to highlight the trends; in fact, since algorithm adaptive thresholds are variable depending on scene characteristics (e.g. objects size, ambient light, etc.), slight unpredictable fluctuations of results around an average are possible, until the effect of compression starts predominating the results. Furthermore, and this especially evident in case of Tunnel FP, the ‘filtering’ effect of the compression can possibly counterbalance the negative effect of quality degradation, by reducing the number of detectable objects.



(a) (b)

(c) (d)

Fig. 4. TP and FP evaluation: (a) turnstile; (b) concourse; (c) platform; (d) tunnel



(a) (b)

Fig. 5. False Negative (a) and False Positive (b) trends w.r.t. quality levels

As expected, tracking performance degrades generally with quality, and this has a much relevant impact at higher levels of compression, in particular when the image quality threshold is lower than 20%, that is at compression ratios higher than 10 (corresponding approximately to 4 Mbps bandwidth occupation).

Starting from those results, a more detailed analysis allowed us to discover the causes of FN and FP and their relevance at higher compression levels.

For FN, the main causes appear to be *tiling* (see Fig. 6) and *occlusions* (see Fig. 7), preventing the tracker to 'hook' the objects in the scene, and thus to track their trajectory, since their IDs change frequently as they were different objects.

For FP, there can be several possible causes (see Fig. 8), including:

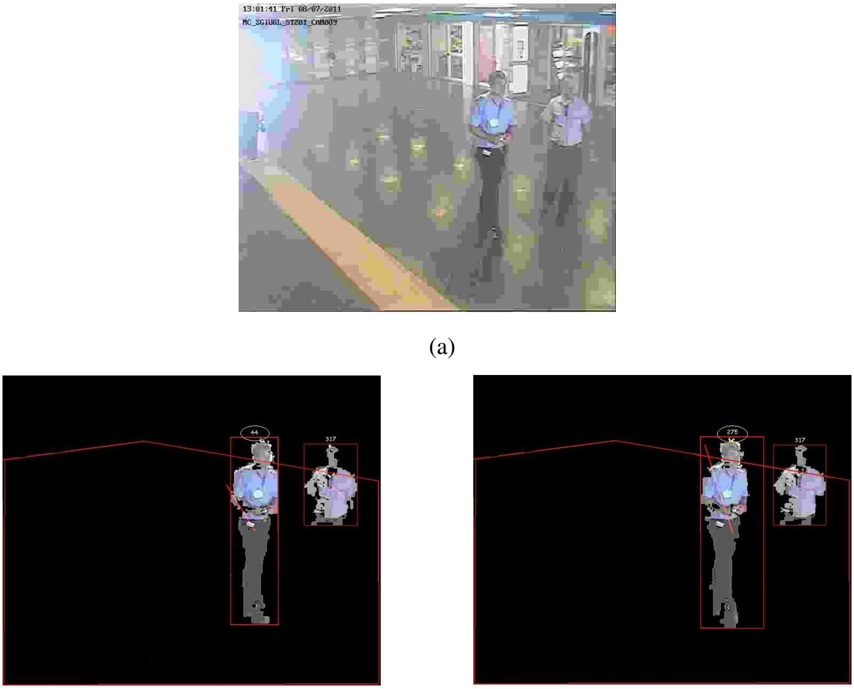
- *Glare*: a strong light source saturates a certain area of the camera sensor causing charge leaks in adjacent pixels; when an object moves, the light reaching the sensor decreases suddenly and so do charge leaks, modifying the appearance of the light source even when it is not covered by any objects.
- *Light Change*: the movement of an object in an area which has a light level that is different from the rest of the scene (e.g. natural light) causes a light variation in the same area, and hence a variation in chromatic components that the Motion Tracker can associate to a new object.
- *Reflection*: such a phenomenon happens when the image of an object is reflected on the floor, generating a variation on chromatic components in that area; the consequent effect is the detection of a 'phantom' object.
- *Camouflage*: it happens when the chromatic components of object parts melt into the background so that the object is no more identified with a single box but it is partitioned into blocks featuring different identifiers.
- *Large Artefacts*: it happens when a group of adjacent pixels undergo a relevant variation of chromatic components in areas in which there is no object movement, caused by reflections or light variations generating tiling artefacts detected by the algorithm as objects.

More specifically, as shown in Fig. 9, in the Concourse all FP causes (especially glare) increase considerably with compression, while in Platform and Turnstiles the effects of artefacts is largely predominant with respect to other causes, which, however, continue to be relevant.

Tunnel FP are not reported since they feature a singular unpredictable behavior, as already shown in Fig. 4d: since there is no real object moving in the scene, they show up only at train passage due to the light change in the scene; furthemore, the absence of most chromatic components with respect to other standard cameras (IR cameras only provide greyscale images) reduces the numerosity of FP causes varying with compression levels.

Using deblocking as a pre-processing tool prior to VCA may improve the results, therefore we are going to experiment this filter on the test-set in the future. We will also add more granularity in the 20%-50% range to highlight possible non-linear behaviors.

Finally, it is important to state that the results reported above are largely conservative, since the areas of interest configured in the real installation are usually smaller; moreover, most FP are filtered at a higher level by scene calibration and by setting object size/speed thresholds to configure the VCA alarms actually active, like: person in restricted area, object left behind, platform line crossing, etc.

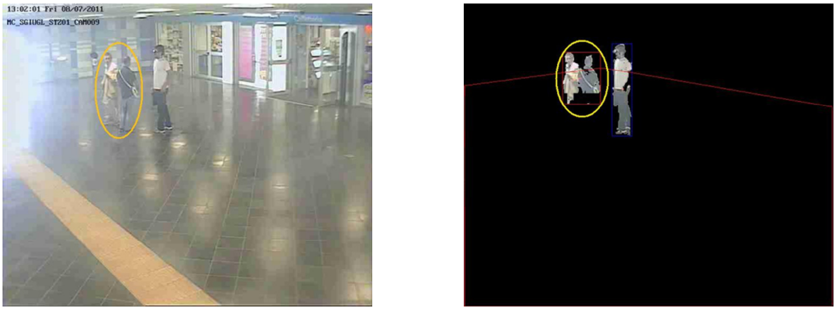


(a)

(b)

(c)

Fig. 6. Example of tiling



(a)

(b)

Fig. 7. Example of occlusion

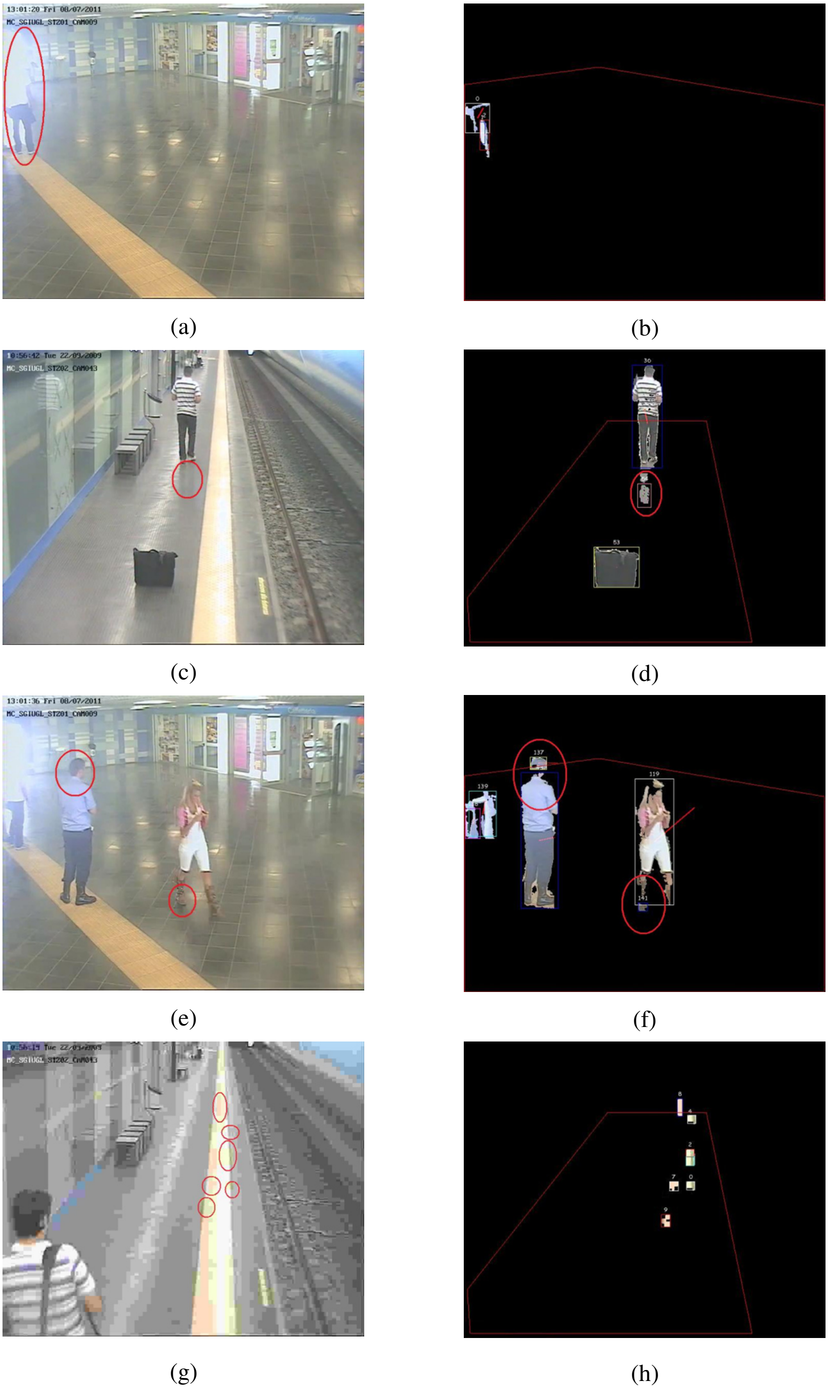
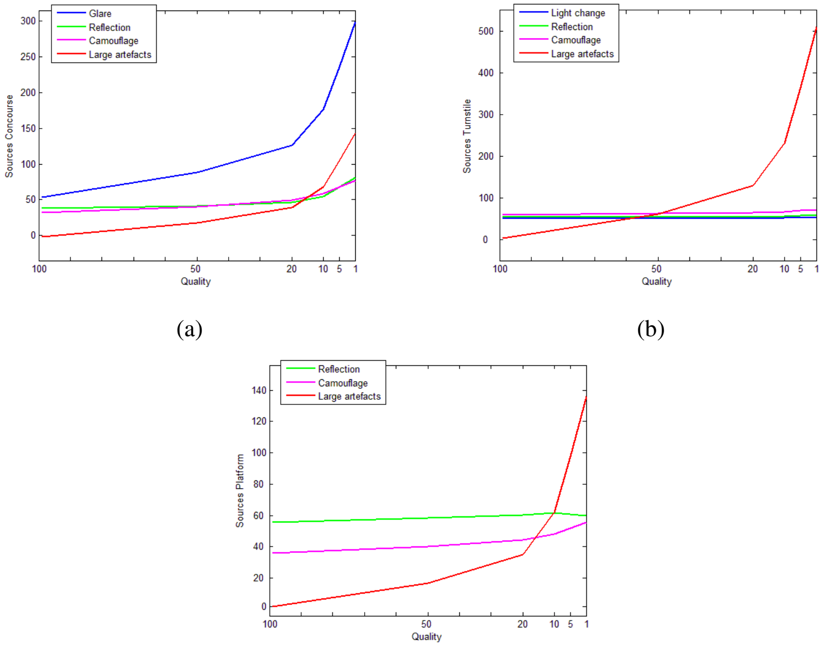


Fig. 8. FP sources: (a)(b) glare; (c)(d) reflection; (e)(f) camouflage; (g)(h) large artefacts



(a)

(b)

(c)

Fig. 9. Relevance of FP sources at growing compression levels in different scenes: (a) concourse; (b) turnstiles; (c) platform

5 Conclusions

The research results presented in this paper have shown the relationship between the performance of a Motion Tracker and the MJPEG video compression level in a real mass-transit installation. By using reference metrics already validated in the scientifc literature and automatically computed by a specifically developed tool, it has been possibile to evaluate a performance degradation, which is critical when passing from a 20% till a 1% quality level of compressed videos, whereas a 50% reduction on image quality representes a very acceptable trade-off (corresponding to ≈ 7 Mbps bandwidth occupation). In all the cases in which it is required to go over that ‘conservative’ ratio, it is necessary to evaluate how the error sources are affected in the correct detection of the objects, according to the specific features of each scene (motion density, light sources, camera shots, type of background, etc.). On this regard, the results achieved can provide some guidelines which can be applicable in similar scenarios (technologies and contexts), e.g. using more efficient codecs like HEVC.

Generally speaking, using the evaluation method described in this paper, it is possible to fine-tune the video compression level against scene characteristics or other factors influencing motion tracking, for each camera. This allows to support the design of a surveillance system, in which it is necessary to concurrently optimize a set of parameters, including the bandwidth of transmitted videos, that are the most relevant contribution in practical applications, since:

- the number of security cameras is ever growing, especially in transit applications
- VCA requirements are increasingly demanding in terms of events to be detected and expected performance

The method and tool used for the analysis provided in this paper are suitable to other evaluations, e.g. considering other quality factors (sensitivity, resolution, frame rate, etc.) or noise factors (vibrations, electro-magnetic interference, chromatic distortions, etc.). Therefore, it is possible to envisage several useful applications in industrial settings considering any other domains, like distributed urban surveillance which can be based on low-band wireless networks [6].

References

[1] Baumann, A., Boltz, M., Ebling, J., Koeing, M., Loors, H.S., Merkel, M., Niem, W., Warzelhan, J.K., Yu, J.: A Review and Comparison of Measures for Automatic Video Surveillance Systems. EURASIP Journal on Image Video Processing (June 2008)

[2] Black, J., Velastin, S.A., Boghossian, B.: A real time surveillance system for metropolitan railways. In: IEEE Conference on Advanced Video and Signal Based Surveillance, pp. 189–194 (2005)

[3] Bocchetti, G., Flammini, F., Pappalardo, A., Pragliola, C.: Dependable integrated surveillance systems for the physical security of metro railways. In: Proc. 3rd ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC 2009), Como (Italy), August 30-September 2, pp. 1–7 (2009)

[4] Candamo, J., Shreve, M., Glodgof, D.B., Sapper, D.B., Kasturi, R.: Understanding Transit Scenes: A Survey on Human Behavior-Recognition Algorithms. IEEE Transactions on Intelligent Transportations Systems 11(1) (March 2010)

[5] Chang, J.-Y., Liao, H.-H., Che, L.-G.: Localized Detection of Abandoned Luggage. EURASIP Journal on Advances in Signal Processing, Article ID 675784 (2010)

[6] Flammini, F.: Critical Infrastructure Security: Assessment, Prevention, Detection, Response. WIT Press (2011)

[7] Grabner, H., Roth, P., Grabner, M., Bischof, H.: Autonomous Learning of a Robust Background Model for Change Detection. In: Proc. 9th IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, pp. 39–46 (2006)

[8] Manohar, V., Soundararajan, P., Raju, H., Goldgof, D.B., Kasturi, R., Garofolo, J.S.: Performance Evaluation of Object Detection and Tracking in Video. In: Narayanan, P.J., Nayar, S.K., Shum, H.-Y. (eds.) ACCV 2006. LNCS, vol. 3852, pp. 151–161. Springer, Heidelberg (2006)

[9] Marpe, D., Wiegand, T., Sullivan, G.J.: The H.264/MPEG4 Advanced Video Coding Standard and its Applications. IEEE Communication Magazine (August 2006)

[10] Nexera Motion Tracker, `http://www.nexera.it/files/VMT_110426.pdf`

[11] Piñero, J.C.: Intelligent Video Results of testing 4 technologies on Madrid Metro. In: Procs. Joint UITP-CUTA International Security Conference, Montreal, Canada, November 11-12 (2009)

[12] Räty, T.: Survey on Contemporary Remote Surveillance Systems for Public Safety. IEEE Transactions on Systems, Man and Cybernetics-Part C 40(5) (September 2010)

[13] Spirito, M., Regazzoni, C.S., Marcenaro, L.: Automatic detection of dangerous events for underground surveillance. In: IEEE Conference on Advanced Video and Signal Based Surveillance, pp. 195–200 (2005)

[14] Yin, F., Makris, D., Velastin, S.A., Orwell, J.: Quantitative evaluation of different aspects of motion trackers under various challenges. In: Quantitative Evaluation of Trackers, Annual of the BMVA, vol. 2010(5) (2010)