# ETISEO, performance evaluation for video surveillance systems

A. T. Nghiem, F. Bremond, M. Thonnat, V. Valentin
Project Orion, INRIA - Sophia Antipolis France

## Abstract

*This paper presents the results of ETISEO, a performance evaluation project for video surveillance systems. Many other projects have already evaluated the performance of video surveillance systems, but more on an end-user point of view. ETISEO aims at studying the dependency between algorithms and the video characteristics. Firstly we describe ETISEO methodology which consists in addressing each video processing problem separately. Secondly, we present the main evaluation metrics of ETISEO as well as their benefits, limitations and conditions of use. Finally, we discuss about the contributions of ETISEO to the evaluation community.*

## 1. Introduction

In this paper we present the evaluation results of ETISEO, a project on performance evaluation of video surveillance systems, sponsored by the French government. The fast increase of the computational power has enabled to build complex video surveillance systems that process video streams in real-time. With the development of these systems, the performance evaluation stage becomes crucial because it is not limited to the identification of system weaknesses but also it enables to determine the conditions of use for a given system.

Concerning video surveillance systems, there are several evaluation projects which have their own purpose and view point. For instance, VACE programs [6] have a wider spectrum including the reliable processing of meeting videos, broadcast news and ground videos. Thus, they pay special attention to efficient tasks such as text detection, face detection, person position detection etc. Meanwhile, PETS workshops [2] primarily focuses on advanced and original algorithms and so evaluate other tasks such as multiple object tracking and event recognition.

ETISEO also addresses surveillance systems evaluation. However, unlike PETS which stands on the algorithm point of view, ETISEO studies the dependencies between video characteristics and algorithms. These studies aim at identifying the suitable scene characteristics for a given algorithm and to highlight algorithm weaknesses for further improvements. ETISEO has achieved these objectives by adopting a principled evaluation methodology, collecting appropriate video sequences, and defining various metrics that help to analyze the diverse aspects of video surveillance systems.

The paper is organized as follows. In section 2 we present an overview of the related work in the evaluation domain. Section 3 then describes in details the ETISEO methodology. In section 4, we detail ETISEO metrics, their strengths and weaknesses dedicated to analyze algorithm performance. Finally, in section 5, we talk about ETISEO contributions to the evaluation domain.

## 2. Related work

There are many individual works on the evaluation of some aspects of video surveillance systems. For instance, [1] characterizes object detection algorithms using the metrics concerning correct detection, detection failures, number of splits, merges and matching area. In [9], the authors propose some recommendations to use ROC and F-measure techniques for (i) system parameters selection and (ii) comparison of performances of multiple algorithms, within the trade-off constraints for a specified end-user application scenario. Nevertheless, these works have a limited influence on the video surveillance community because they do not constitute a whole evaluation platform enabling a new algorithm to be evaluated. Moreover, their data set is not big enough to achieve reliable evaluation results.

Therefore, to answer the need of having a publicly available set of annotated video sequences, many evaluation programs such as CAVIAR [3], VACE [6], CREDS [4], CLEAR [5] and workshops (PETS [2]) have been created. These research programs provide video sequences at various global "difficulty levels" together with associated ground truth. However, the same global "difficulty levels" may be constituted by different individual video processing problems (e.g. shadows, weak contrast). Consequently, the evaluation process does not enable to gain some insight into each video processing algorithm. Specifically, for a given algorithm, the evaluation does not indicate which video processing problems that it have to pay attention to, which improvement is the most crucial and in what conditions this algorithm can achieve satisfactory performance.

ETISEO [7], one of the latest evaluation projects (ended in December 2006), has tried to address these issues. Unlike

VACE, CREDS or CLEAR which stand more on the user point of view, ETISEO tries to help algorithm developers to identify their weaknesses by underlining the dependencies between algorithms and their conditions of use.

# 3. ETISEO project

ETISEO aims at evaluating video processing algorithms given a video processing task (i.e. object detection, classification, tracking and event recognition), a type of scene (e.g. road) and a global difficulty level (e.g. contrasted shadows). The ultimate goal is to study the dependencies between a video processing task and video characteristics (e.g. shadows), which are called in the following, video processing problems. The methodology of ETISEO is as follows:

Firstly, ETISEO addresses separately each video processing problem that have been accurately defined and classified. For instance, handling shadows can be studied within at least three different problems: (1) shadows at different intensity levels (i.e. weakly or strongly contrasted shadows) with uniform non color background, (2) shadows at the same intensity level with different types of background images in terms of color and texture and (3) shadows with different illumination sources in terms of source position and wavelengths.

Secondly, ETISEO collects video sequences illustrating only a given problem. The video sequences were intended to illustrate the video processing problem at different difficulty levels. For instance, for the problem of shadows and intensity levels, we have selected video sequences containing shadows at different intensity levels (more or less contrasted). On these selected sequences, the appropriate part of the ground truth is filtered and extracted to isolate video processing problems. For instance, for the object detection task, we can evaluate the algorithm performance relatively to the problem of handling occluded objects by considering only the ground truth related to the occluded objects.

Thirdly, ETISEO computes three types of associated data for each video sequence. The first one is the ground truth (e.g. object bounding box, object class, event etc.) given by human operators using VIPER tool [8] at each level of the four video processing tasks. The second one is the general annotation on the video sequences concerning video processing difficulties (e.g. weak shadows) or concerning recording conditions (e.g. weather conditions such as sunny day). The third information is the camera calibration and contextual information about the empty scene describing the topology of the scene (e.g. zone of interest).

Fourthly, ETISEO has defined various metrics to evaluate the performance of a video surveillance system for every video processing task (object detection, tracking, object classification and event recognition). The ETISEO metrics are described in the next section.

Finally, ETISEO provides a flexible and automatic evaluation tool to accurately analyze how a given algorithm addresses a given problem.

# 4. Evaluation metrics

This section briefly describes ETISEO metrics and concentrates more on their characteristics. For detailed metric description, see [7].

ETISEO testing data set contains 40 multi-camera scenes corresponding to 85 sequences. We will use evaluation results on these sequences to discuss about the advantages as well as the weaknesses of the ETISEO metrics. In ETISEO each participant algorithm is assigned with a unique number to ensure the anonymity during the evaluation. Therefore, from here on, we will use these numbers to refer to the corresponding algorithms.

*Matching functions:* to compute most of the metrics, we need to match area or time intervals of objects in the reference data to objects detected by algorithms. If the value of the matching measure is higher than a predefined threshold, we consider that the reference data matches the detected objects. After trying various matching functions, we found that, the choice of matching functions does not greatly affect the evaluation results. Hence, in this paper, we only present the Dice coefficient function D1. D1 is defined as $2 \times card(RD \cap C)/(card(RD) + card(C))$. Here $RD$ is the time interval or area of a Reference Data, $C$ is the corresponding of a Candidate data (detected by an algorithm).

*Performance ratios:* the ETISEO metrics often use the performance ratios such as Precision ($GD/OD$), Sensitivity ($GD/RD$) or F-Score ($(2 \times Precision \times Sensitivity)/(Precision + Sensitivity)$). Here GD is the number of good detections (i.e. True Positive), OD is the total number of detections (i.e. TP + FP), and RD is the number of reference data (i.e. TP + FN).

## 4.1. Metrics for object detection

For the object detection task, there are one main metric (number of objects) and three other complementary ones (object area and split/merge metrics).

The **metric "number of objects"** evaluates the number of detected objects (called blobs) matching reference objects using their bounding boxes. Its main advantage is that it does not favor large blobs like pixel based metrics. When we are using pixel based metrics, if the characteristics of the pixels inside large blobs are homogeneous, the evaluation results will be biased towards this particular sequence and they will not reflect the real performance of the algorithm.

Since the metric "number of objects" matches the detected and reference objects based on a thresholded value of the matching function D1, it cannot measure the precision of the detection. For instance, with a certain value of the

threshold, it cannot distinguish one algorithm which overly detects 120% of the area of a given object with another algorithm which exactly detects 100% of the object area. To evaluate algorithm precision, we have to use the metric "object area".

The **metric "object area"** evaluates the number of pixels in reference data that have been detected.

Table 1: *Performance results using the metric "number of objects" on sequence ETI-VS2-BE-19-C1*

| **Algorithm** | 9 | 1 | 14 | 28 | 12 | 13 | 32 |
|---|---|---|---|---|---|---|---|
| **F-Score** | 0.49 | 0.49 | 0.42 | 0.4 | 0.39 | 0.37 | 0.37 |
| **Algorithm** | 8 | 19 | 20 | 17 | 29 | 3 | 15 |
| **F-Score** | 0.35 | 0.33 | 0.32 | 0.3 | 0.24 | 0.17 | 0.11 |

Table 2: *Performance results using the metric "object area" on sequence ETI-VS2-BE-19-C1*

| **Algorithm** | 1 | 13 | 9 | 32 | 14 | 12 | 20 |
|---|---|---|---|---|---|---|---|
| **F-Score** | 0.83 | 0.71 | 0.69 | 0.68 | 0.65 | 0.65 | 0.64 |
| **Algorithm** | 19 | 28 | 17 | 3 | 29 | 8 | 15 |
| **F-Score** | 0.64 | 0.59 | 0.55 | 0.54 | 0.51 | 0.5 | 0.3 |

Tables 1 and 2 show the evaluation results using the metrics "number of objects" and "object area" on sequence ETI-BE-19-C1. On this sequence, there is a large object (a car) which is quite easy to detect and several small objects (people) which are more difficult to detect. Because the metric "object area" is biased towards the car, the misdetection of the people does not affect much the evaluation results. If we use the metric "number of objects", the evaluation shows that the algorithm 9 has better results than the algorithm 13. This reflects the fact that algorithm 9 can detect more human blobs than algorithm 13. However, using the metric "object area", the evaluation shows that algorithm 13 has better results than algorithm 9. It means that algorithm 9 has lowered its threshold to detect more blobs.

The **split metric** qualifies the fragmentation of detected objects. Particularly, it computes the number of blobs per reference object, using a specific matching function D2 ($D2 = card(RD \cap C)/card(C)$). Normally, if only parts of the object are detected (splitted objects), the bounding boxes of these parts are much smaller than the object bounding box in the reference data. Therefore, using the matching function D1, all the small parts detected by a given algorithm will be eliminated. Thus we have defined the matching function D2.

During ETISEO evaluation most algorithms have good performance with the split metric and could not be discriminated by this metric. However this metric was useful to detect some error cases of over-detection (e.g. a person is detected as two objects). Thus this metric can be used to correct this type of errors.

The **merge metric** qualifies the overlapping of detected objects. It computes the number of reference bounding boxes per detected object, using the bounding box overlap constraint. However, for the videos in ETISEO the distance between objects is usually important enough so that most of the algorithms do not merge objects. Moreover when two objects are close to each other, if one objects is not detected, there will still not be any merge. Therefore, to measure the algorithm performance using this metric, we should extract video clips in which objects are well contrasted and close to each other.

The split/merge metrics explore the algorithm capability in handling split and merge situations in specific and short parts of video sequences. Therefore the evaluation results on the whole sequence using these two metrics are close to 100% for most of the algorithms. With appropriate data, these metrics can be well adapted to correct specific errors of split and merge.

## 4.2. Metrics for object localisation

The **metrics "2D/3D - distance"** measure the average of the 2D/3D distance between gravity centers of detected objects and corresponding reference objects. These metrics help to determine the detection precision, similarly to the metric "object area". Unlike the metric "object area", the localisation metrics are not biased towards big objects. Nevertheless, for a fair evaluation, these localisation metrics should be applied on the same set of detected objects for all the algorithms. If not, the evaluation results of good algorithms may be affected by difficult objects (e.g. far from the camera) which are not detected by other algorithms. Therefore collecting appropriate data is a difficult issue for these metrics. A solution could be the collection of specific types of test videos containing for example far, well contrasted and not occluded objects. Besides that these metrics are based on the gravity center computation. Thus detection errors on the outline are not evaluated by these metrics. Moreover, there is no consensus to compute the 3D - gravity center of some objects like cars.

Table 4 shows the evaluation results using the metric "2D distance" on sequence ETI-VS2-RD-10-C4. This table illustrates one of the disadvantages of the metrics that we discuss above. For instance, in this table, the results of algorithm 17 is quite good, but in table 3, its evaluation results using the metric "number of objects" is nearly the worst. It means that, this algorithm can handle well the objects which are easy to detect but not the difficult ones. Thus, the algo-

Table 3: *Performance results using the metric "number of detected objects" on sequence ETI-VS2-RD-10-C4*

| **Algorithm** | 14 | 13 | 1 | 19 |
|---|---|---|---|---|
| **F-Score** | 0.54 | 0.43 | 0.33 | 0.32 |
| **Algorithm** | 20 | 17 | 29 | 12 |
| **F-Score** | 0.25 | 0.23 | 0.23 | 0.23 |

Table 4: *Performance results using the metric "2D - distance" (from gravity center) on sequence ETI-VS2-RD-10-C4*

| **Algorithm** | 14 | 13 | 17 | 19 |
|---|---|---|---|---|
| **Distance** | 6.99 | 7.31 | 7.7 | 8.08 |
| **Algorithm** | 20 | 29 | 12 | 1 |
| **Distance** | 8.15 | 8.87 | 9.33 | 9.88 |

rithm 17 obtains a high score for the metric based on gravity center.

The metric "2D-distance", similarly to the "object area" metric, provides complementary information to the "number of objects" metric for assessing the detection task performance. For instance, we can see that algorithm 1 has good results using the metric "number of object" but bad results using the metric "2D distance". It means that this algorithm has lowered its detection precision to have a high detection rate.

## 4.3. Metrics for object tracking

For the tracking task, there are one main metric (tracking time) and two other complementary ones (object ID persistence / confusion).

The **metric "tracking time"** measures the percentage of time during which a reference data is detected and tracked. The match between a reference data and a detected object is done with respect to their bounding box. This metric gives us a global overview of the performance of tracking algorithms. Yet, it suffers from the issue that the evaluation results depend not only on the tracking algorithms but also on the process of object detection. In other words, we can track only the objects that we have detected.

The complementary metrics qualify the tracking precision. The **metric "object ID persistence"** helps to evaluate the ID persistence. It computes over the time how many tracked objects are associated to one reference object (ID persistence). However, it favors under-detection. For instance, using this metric, an algorithm tracking a reference object during a small period of time has higher evaluation results than another algorithm tracking the same reference object during two periods of time but with different IDs. In the contrary, the **metric "object ID confusion"** computes the number of reference object IDs per detected object. An example of confusion (exchange of IDs) may be due to two people meeting. The drawback of this metric is that it favors over-detection. In particular, if an algorithm detects several objects for one reference object, which is an erroneous case, then it will get a high score with the metric "object ID confusion" because each detected object matches with only one or zero reference object. Therefore an algorithm having a high score with these two metrics does not mean that it is good at tracking. To evaluate the performance of tracking algorithms, these metrics must be used together with the main metric. The following example will illustrate the use of these metrics to analyze the tracking evaluation results.

Table 5: *Performance results using the metric "tracking time" on sequence ETI-VS2-BE-19-C1*

| **Algorithm** | 1 | 9 | 14 | 12 | 19 | 13 | 32 |
|---|---|---|---|---|---|---|---|
| **Tracking time** | 0.33 | 0.32 | 0.28 | 0.27 | 0.25 | 0.24 | 0.24 |
| **Algorithm** | 28 | 17 | 8 | 20 | 29 | 3 | 15 |
| **Tracking time** | 0.21 | 0.19 | 0.18 | 0.17 | 0.16 | 0.14 | 0.11 |

Table 6: *Performance results using the metric "object ID confusion" on sequence ETI-VS2-BE-19-C1*

| **Algorithm** | 19 | 20 | 28 | 17 | 3 | 29 | 14 |
|---|---|---|---|---|---|---|---|
| **persistence** | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **Algorithm** | 8 | 15 | 32 | 13 | 9 | 12 | 1 |
| **persistence** | 0.95 | 0.94 | 0.9 | 0.9 | 0.88 | 0.83 | 0.77 |

Table 5 shows the evaluation results using the tracking time metric and table 6 shows the evaluation results using the metric "object ID confusion" on sequence ETI-VS2-BE-19-C1. If we use only the main tracking metric "tracking time", algorithm 1 has the best results. However, if we use the metric "object ID confusion", algorithm 1 results are not so good. Further analysis shows that it has assigned the same ID to at least two reference objects.

## 4.4. Metrics for object classification

For the object classification task, there is one metric: object types using bounding box. This metric computes in each frame the number of correctly classified detected objects matching a reference object using their bounding box. The evaluation results using this metric are quite reliable, even if only a small number of object types were available.

### 4.5. Metrics for event recognition

For the event recognition task, we use mainly one metric (number of correctly recognised events with the constraint of time). We may define more strict constraints that takes into account also the objects involved into the event but because of the algorithm results and event simplicity, we use only the time constraint. For instance, in ETISEO we did not observe errors due to the wrong involvement of objects in an event. This task is difficult to evaluate because the challenge depends strongly on the events to recognize. For instance, it is much easier to detect an intrusion in a zone of interest than a person opening the door.

## 5. ETISEO Contributions

In ETISEO, although that some videos were very challenging, most algorithms had high performance results and it was difficult to establish a global ranking on the whole video set. Some algorithms were performing better in some situations and worst in others. However we have observed that the algorithms with higher evaluation rates were often the ones combining region tracking using background subtraction with local descriptors tracking (e.g. HOG or SIFT descriptors, KLT tracker).

### 5.1. ETISEO advantages

To ensure meaningful evaluation results, ETISEO has applied several good practice rules.

First of all, ETISEO has provided a large set of data and metrics to evaluate video processing algorithms. For each task, we have indicated a main metric to assess the global algorithm performance and several complementary metrics to qualify the algorithm precision. The previous sections have clarified the value of ETISEO metrics with respect to the video characteristics and the evaluation objectives and it can be seen as a manual to efficiently use these materials to get an efficient evaluation of a given algorithm.

Secondly ETISEO has defined two ontologies to facilitate the communication between all participants in this domain: researchers, software developers and end-users (e.g. administrations, companies). The first one describes technical concepts used in the whole video interpretation chain (e.g. a blob, an individual trajectory) as well as concepts associated to the evaluation (e.g. reference data). The second one describes concepts of the application domains (e.g. opening a door event).

Thirdly, ETISEO automatic evaluation tool ensures a fair and quantitative comparison between algorithm results and reference data. Users can interactively select parameters to perform different types of evaluation. For instance, filters have been provided to select and evaluate a particular data type (e.g. stationary objects). This tool also enables to visualize algorithm results compared to the ground truth and to browse through the whole video set.

Fourthly, ETISEO has divided the evaluation into two phases. During the first phase, ETISEO participants have tested their algorithms on a sample data set. This phase was aiming to help participants to get used to data and clarify the evaluation requirements. Moreover, it has also helped the evaluators to adjust their evaluation protocol according to the participant feedbacks. The results of the second phase was the final evaluation results.

Fifthly, ETISEO has enabled to evaluate video processing algorithms in challenging situations (e.g. crowd scenes) and up to the recognition of events of interest (e.g. abandoned luggage).

Finally, because of its large dataset, ETISEO had to tackle the problem of having too few algorithm results on a desired sequence by giving priority for some video sequences which are representative of typical types of scene. Therefore, these sequences have been processed by most of participants. Only few participants have complained that priority sequences have narrowed the evaluation scope.

### 5.2. ETISEO limitations

ETISEO had to face several shortcomings.

Firstly, there were still inconsistencies among participants, particularly in defining the objects and events of interest. For instance, several participants processed the stationary objects differently. Some participants considered the objects not moving for a certain period of time as part of the background and eliminated them from the algorithm results while others detected these objects up to the end as it was requested. Therefore it was difficult to compare the algorithm results of these participants. The solution was to create a filter that removes these objects from both the ground truth and the algorithm results. After applying the filter, the algorithms were ranked differently. This filter has enable us to distinguish two different problems: handling stationary objects which were previously mobile and mobile object detection.

Secondly ETISEO did not set up a limit on the processing time to satisfy the real-time requirement. Hence some participants have applied sophisticated algorithms with a learning stage and have obtained good evaluation results. Moreover, ETISEO did not require the participants to keep the same algorithm parameters for all the video sequences or at least for each type of scene. Consequently, they have tuned their algorithms to achieve better results on each video sequence. Then, the evaluation results do not reflect the algorithm performance in real conditions which change arbitrarily but rather the partner involvement in ETISEO. To mitigate the performance results, the participants were asked to fill up questionnaires indicating the algorithm requirements (e.g. how many parameters have been tuned).

ETISEO has set up a workshop to demonstrate the real time capability and the dynamic configuration of the systems.

Thirdly, the evaluation results communicated through numbers and curves mostly help to compare the algorithm performance between themselves. In the user point of view, it is difficult to answer the question of how significant are these values. For instance, using metric "number of objects", is the F-Score value equal to 0.8 good enough? is the difference of 0.1 between two algorithms significant? There is no absolute answer to these questions because the answer depends on the specific application. We should perform also an end user evaluation on a selection of applications to establish the significance of the evaluation results (i.e. numbers).

Finally, although ETISEO has tried to estimate the difficulty levels of the video processing problems in each sequence, this estimation is still very rough. For instance, ETISEO uses the terms "normal" or "dark" to describe the intensity levels of video sequences. Therefore, the selection of video sequences in ETISEO according to their difficulty levels is not sufficient because the comparison among video sequences is subjective and imprecise. Moreover, the prediction of algorithm performance on new scenes based on these evaluation results is difficult because we have to compare these new scenes with the ETISEO video sequences. To solve this problem, we are currently working on defining objective and quantitative metrics to measure automatically the difficulty levels of video processing problems [10].

# 6. Conclusion

This paper presents the main contributions of ETISEO, a video processing evaluation project which is centered on the algorithm developers. First of all, its principled evaluation methodology helps the algorithm developers to determine the dependency between their algorithms and the video characteristics. Besides that, by isolating each video processing problem, this evaluation project helps for a given algorithm to concentrate on a selection of problems with priority levels for future improvements. Moreover, the video processing evaluation community can reuse the ETISEO evaluation tool and its metrics which are dedicated to each video processing task. In particular, the strengths and limitations of these metrics have been identified and validated through the ETISEO evaluation results.

In the future, we will extend the evaluation tool to be more flexible by creating filters dedicated to other video processing problems. For instance, we will develop the filter to select only the occluded objects to evaluate algorithm ability in processing occlusions (static vs dynamic and partial vs total occlusions). Moreover, we will add objective metrics to quantify automatically the difficulty levels of video processing problems in videos to facilitate the video characterization and selection. These metrics will enable to generalize the evaluation results for new scenes as described in [10]. After that we will study the interdependency between video processing problems to extend the ETISEO results to a combination of problems.

# References


[1] Jacinto Nascimento and Jorge Marques, "Performance evaluation of object detection algorithms for video surveillance", *IEEE Transactions on Multimedia 8*, pp. 761-774. 2006.

[2] *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS)*, http://www.pets2006.net/

[3] *CAVIAR: Context Aware Vision using Image-based Active Recognition*, http://homepages.inf.ed.ac.uk/rbf/CAVIAR/

[4] *CREDS: Call for Real-Time Event Detection Solutions (CREDS) for Enhanced Security and Safety in Public Transportation*, http://www-dsp.elet.polimi.it/avss2005/CREDS.pdf

[5] *CLEAR: Classification of Events, Activities and Relationships - Evaluation Campaign and Workshop*, http://www.clear-evaluation.org/

[6] *VACE: Video Analysis and Content Extraction*, http://www.ic-arda.org/InfoExploit/vace/index.html

[7] *ETISEO: Video understanding Evaluation*, http://www.silogic.fr/etiseo

[8] *ViPER-GT, the ground truth authoring tool*, http://viper-toolkit.sourceforge.net/docs/gt/

[9] N. Lazarevic-McManus et al, "Designing Evaluation Methodologies: The Case of Motion Detection", *In Proceedings of 9th IEEE International Workshop on PETS*, pages 23-30, New York, June 18, 2006.

[10] A.T. Nghiem, F. Bremond, M. Thonnat, R.Ma, "A New Evaluation Approach for Video Processing Algorithms", *In Proceedings of IEEE International Workshop on Motion and Video Computing, Austin, Texas* 23-24 February 2007.