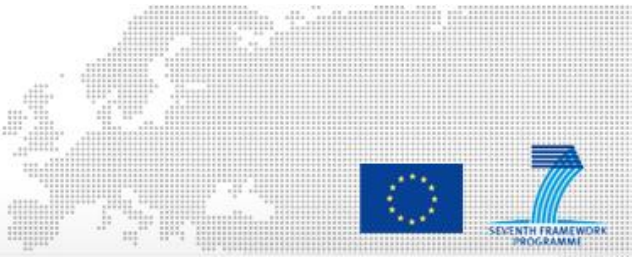
Report on the Test Methodologies and Metrics for the Real Context Tests

Deliverable 5.2

HP

Automatic Data relevancy Discrimination for a PRIVacy-sensitive video surveillance



Automatic Data relevancy Discrimination for a PRIVacy-sensitive videosurveillance

SEC-2010.6.5-2 - Use of smart surveillance systems, data protection, integrity and sharing information within privacy rules

D5.2 - Report on the Test Methodologies and Metrics for the Real Context Tests

Due date of deliverable: Month 23

Actual submission date: 09/04/2013

Start of project: 01 February 2011 Duration: 36 Months

Lead Contractor for this deliverable: HP

Revision: 1.0

Project co-funded by the European Commission within the Seventh Framework Programme		
Dissemination level		
PU	Public	
CO	Confidential, only for members of the consortium (including the Commission Services)	X

Revision History

Deliverable Administration and summary		
Project Acronym: ADDPRIV		Grant Agreement no: 261653
Document Identifier: ADDPRIV_API		
Leading partner: HP		
Report version: 1.0		
Report preparation date: 9/04/2013		
Classification: Confidential		
Nature: Report		
Author(s) and contributors: HP		
Status		Plan
		Draft
		Working
	X	**Final**
		Submitted
		Approved

Changes to the document are detailed in the change log table below.

Date	**Edited by**	**Status**	**Changes made**
06/03/2013	HP	Beta version	Initial version
12/03/2013	HP	Beta version	Positive and negative scenarios
19/03/2013	HP	Beta version	Improved testing methodology section
27/03/2013	HP	Beta version	Improved scenarios sections
09/04/2013	HP	Beta version	Updated according to Gdansk's feedbacks

Copyright

Citation

HP (2013). Deliverable 5.2: "Report on the Test Methodologies and Metrics for the Real Context Tests". ADDPRIV Consortium, www.addpriv.eu

Acknowledgements

The work presented in this document has been conducted in the context of the EU Framework Programme project with Grant Agreement 261653 ADDPRIV (Automatic Data relevancy Discrimination for a PRIVacy-sensitive video surveillance). ADDPRIV is a 36 months project started on February 1st, 2011.

The project consortium is composed by: Anova IT Consulting (ANOVA), Kingston University Higher Education Corporation (KU), Politechnika Gdanska (GDANSK), Lancaster University (ULANCS), Avanzit Tecnología, S.L. (AVANZIT), Hewlett Packard Italiana Srl (HP), Societá Per Azioni Esercizi Aeroportuali Sea SPA (SEA), Renfe Operadora (RENFE) and The Provost Fellows & Scholars Of The College Of The Holy And Undivided Trinity Of Queen Elizabeth Near Dublin (TCD).

More Information

Public ADDPRIV reports and other information pertaining to the project are available through ADDPRIV public website under www.ADDPRIV.eu

Table of contents

1. Introduction 7
2. Testbed environment and configuration 8
2.1 Hardware configuration and software configuration 8
3. Testing methodology and environment 10
4. Negative scenarios 16
5. Positive scenarios 19
6. Conclusion 26

1. Introduction

ADDPRIV provides a new solution to limit the storage of unnecessary data, proposing a solution for automatic discrimination of relevant data recorded by a multi-camera network, based on the automatic identification of security-relevant events.

The events that are considered relevant are:

- Left luggage
- Barrier crossing (intrusion in a forbidden area)
- Counter flow

Due to the increasing risk of terrorist attacks, an abandoned luggage represents a potentially dangerous event for public safety, especially in public places as railway stations and airports. Detection of barrier crossing events allows identifying two different, but related types of forbidden activities. The first is when an object/person enters a restricted area or walks into an unauthorized zone (intrusion); the second is related to existing barrier where people can be present at the both sides, but crossing is forbidden. Counter flow is triggered when an object/person moves in an opposite direction to the general flow.

The present deliverable provides a plan to perform functional testing of the integrated AddPriv system in one of the real environments for which it has been designed. Different test cases are planned to be executed in the form of both positive and negative scenarios. The document is directed both to people responsible for setting up the testing environment and people responsible for executing and observing the results of tests.

Section 2 reports the requirements and details about the hardware and software to be configured for testing. Section 3 describes the testing environment and includes information and indications about the testing methodology. Testing scenarios are defined in sections 0 and 0. Conclusions are reported in section 0.

2. Testbed Environment and Configuration

The environment in which real context tests will take place is Linate airport, where 9 of the security cameras can be used for the purpose of the project.

There are three types of cameras, with different resolution and bandwidth requirements: low resolution cameras (704x576 – 1Mb/sec), medium resolution cameras (1280x800 – 3Mb/sec) and high resolution cameras (2048x1536 – 3,6Mb/sec).

This section defines the typical requirements of each module in terms of hardware resources and software prerequisites. The following requirements will guide the setup of the infrastructure in the airport for executing the real context tests.

2.1 Hardware configuration and software configuration

Hardware requirement are divided by module and can be summarized as follows.

Requirements for the PE component are:

- **CPU cores**: 8
- **CPU arch.**: 64 bit
- **RAM**: 16Gb
- **HD**: 1Tb SSD per camera per day
- **SO**: Ubuntu Server 12.10

The storage space required for PE depends on both the number of cameras used for the test and on the expected amount of video history. For real operation of the system with nine cameras and one full day of video history more than 9Tb of storage will be needed. For the purpose of the tests the amount of video history to be kept online before deletion can be set to a smaller value, say 1 hour, to reduce storage requirements. A single disk of 500 GB will be more than enough to keep 1 hour of video history.

Requirements for the ED component are:

- **CPU cores**: 4
- **CPU speed**: 2 GHz
- **RAM**: 2 GB
- **SO**: Windows

Each node configured as indicated above will be able to process one high resolution video stream, or two at medium resolution or three at low resolution. For real operation at least four nodes of this type will be needed. The setup for the real context tests will depend on the available hardware and will possibly provide more data for sizing the real operation setup.

Requirements for MP are the following:

- **CPU cores**: 2 or 4
- **CPU arch.**: Intel (32 bit)
- **RAM:** 2 to 4 GB
- **SO:** Windows XP SP3

Requirements for RR are mostly on the CPU architecture and Operating System:

- **CPU arch.**: 32 bit
- **SO**: Windows XP SP3

In addition to that RR requires also a custom hardware component: the VIS212 video analytics server from www.ipsotek.com. Requirements for the UI component are the cheapest ones. The server can run on both Windows and Linux. For Linux it's enough

- **CPU cores**: 1
- **CPU speed**: 2 GHz
- **RAM**: 512 MB

The UI client runs in the most common browsers, such as Chrome, Firefox and IE.

3. Testing Methodology and Environment

Testing will be performed in the real world with real scenarios, in order to verify the correct operation of the system under real operating conditions. In particular it will be performed in one of the possible real environments where the system may be deployed: the Linate airport. The validation scenarios described in this document have been defined in collaboration with the interested end-user partner and will be played by end-user actors. The related behaviour of the system during the execution of each scenario will be observed and reported.

The map of the controlled area is reported in Figure 1 and Figure 2 along with the position of the installed cameras. Cameras indicated in Figure 1 are the old low resolution cameras (704x576), those represented in Figure 2 are the new high resolution cameras (from 1280x800 for cameras 3 and 4 to 2048x1536 for 1 and 2).

The maps reported in the following figures are borrowed from deliverable D2.2, in which more details are available about the cameras and the environment they operate in.

A single up-to-date map will be prepared before the real testing starts, both to include it in the UI and to keep it as a reference during the execution of tests.

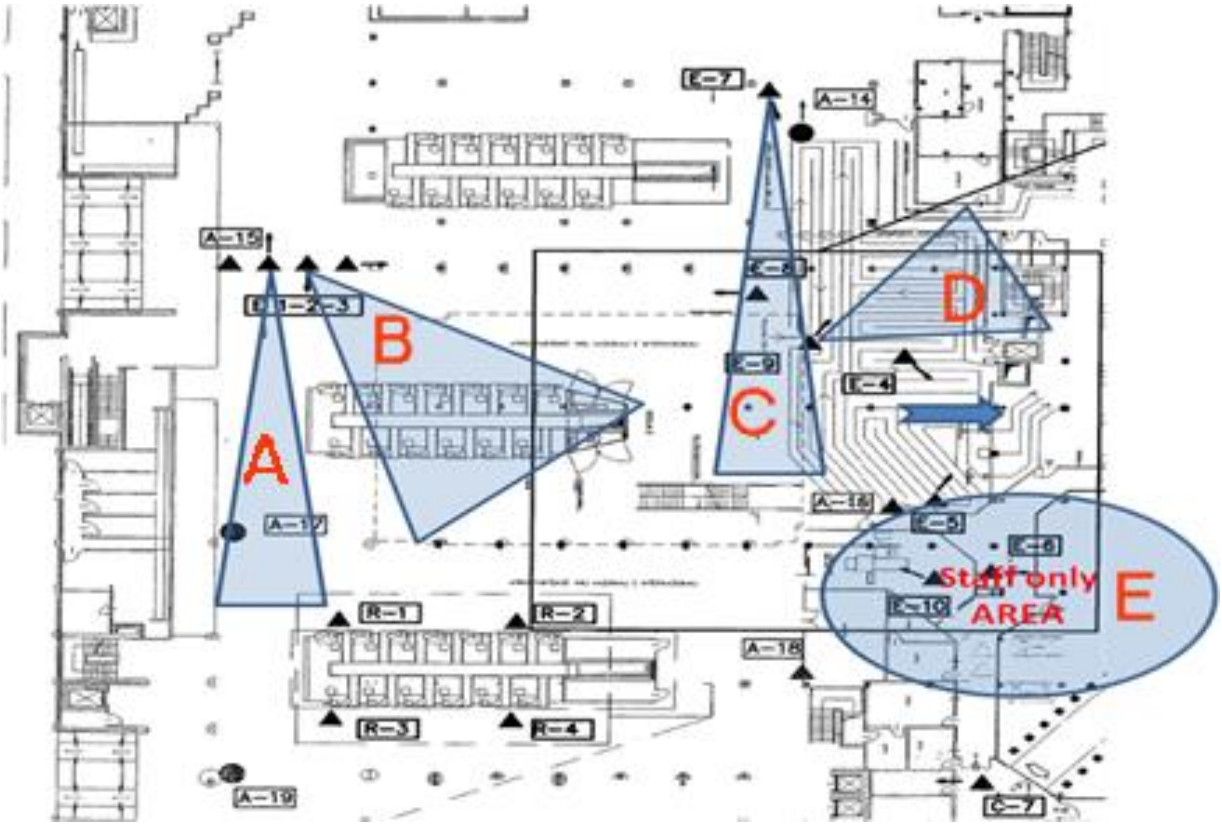


Figure 1 - Position of the old low resolution cameras

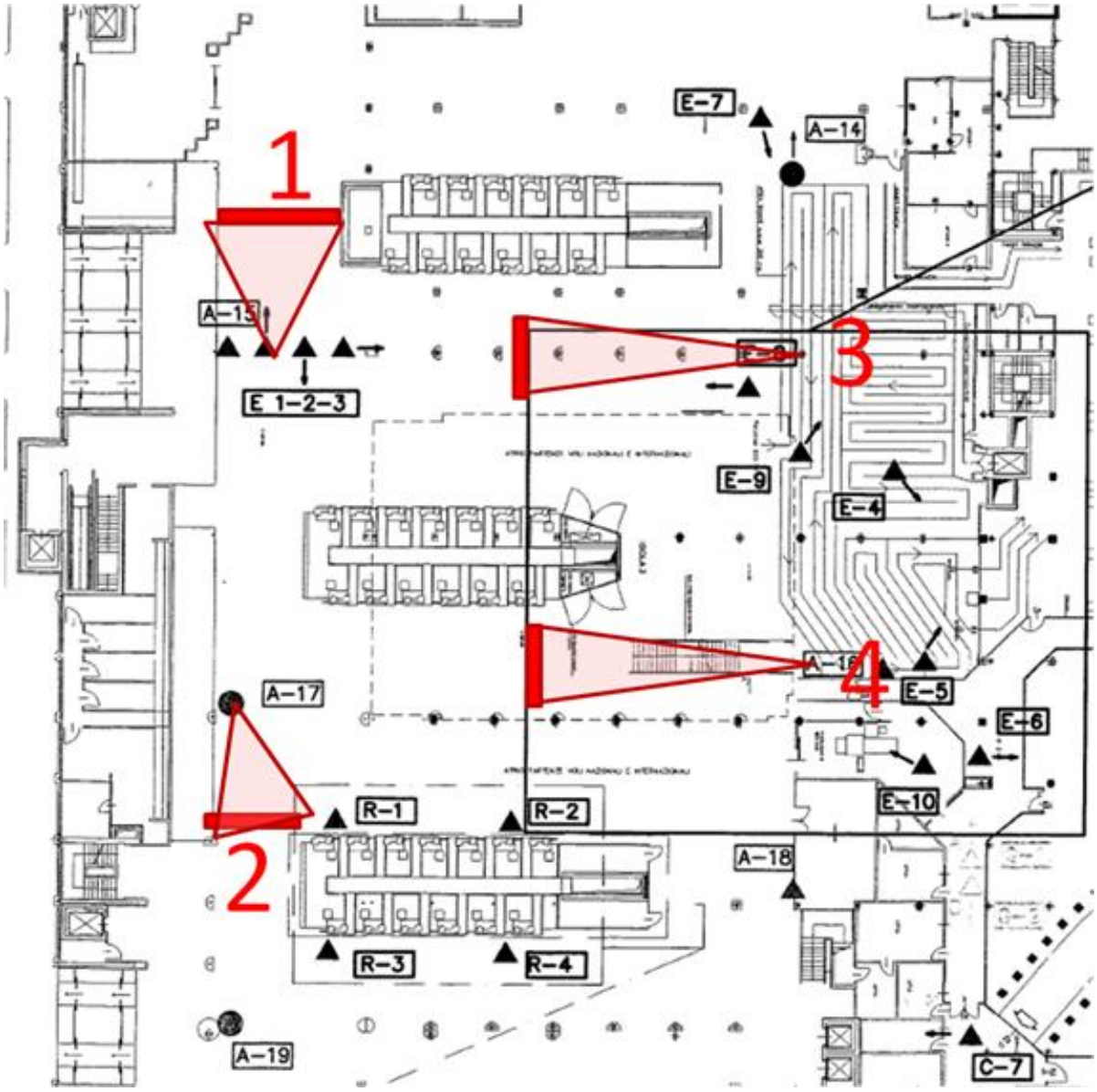


Figure 2 - Position of the new high resolution cameras

The events that the system is able to recognize are the following:

- Left luggage
- Barrier crossing (intrusion in a forbidden area)
- Counter flow

The tests planned in the next sections will be based on the above events and will include both negative and positive scenarios. Positive scenarios will represent actual situations testing specific system functionalities with the expectation to positively verify their

working. Negative scenarios will try to "fool" the system by representing borderline situations to test the absence of false positives.

Real world tests will be end-to-end tests and will involve the whole system with most or all its components. Such end-to-end scenarios will test both the correct working of each involved component and their proper integration.

During test execution a few constraints must be respected to ensure better event recognition performance:

- Luggage colour must not be similar to floor colour
- Size of the abandoned luggage should be at least equal to the standard carry-on size
- When it is abandoned the luggage must be visible (the luggage can't be overlapped by other objects)
- Luggage abandoned far away from the camera, event if within its field of view, can't be detected
- A person making an event must not be occluded by other persons during event duration

For more complete verification of the system each test scenario will be repeated under different conditions and each repetition will be evaluated and reported as a separate test. A crossing table will combine all scenarios with the planned variations in the environment conditions to enumerate all the tests to be performed. To ensure proper reporting of tests results, each single test in this table will be marked with a unique identifier.

Conditions to be considered for varying both positive and negative base scenarios are the following:

- Clothing of actors (flashy vs. gray)
- Resolution of the camera under which the event is represented (high vs low)
- Crowding of the environment (low, intermediate or high – see D2.2 sect. 5.1.2)

Results of the real world functional tests will be reported in the next deliverable D5.4[1] and will be further interpreted and evaluated according to the validation criteria and scoreboard defined in deliverable D5.1[2], whose results will be reported in deliverable D6.1[3].

For proper evaluation of the system according to criteria defined in D5.1 some details could be observed during real world functional tests and answers to the questions indicated in both the legal and ethical compliance scoreboards could be collected. Proper planning for observing and reporting legal and ethical compliance aspects is not in the scope of this deliverable: WP6 partners will take care of this in the context of T6.1, T6.2 and T6.3 tasks.

[1] Report on performance test results, in the application context, and improvements required in the different solutions
[2] Final Definition of the Evaluation Metrics and Scoreboard for Validation of System Compliance with Privacy
[3] Report analysing human rights, civil liberty and public participation potential of these solutions

Before executing some of the scenarios (mostly the positive ones) the Security Department of the Airport must be contacted and informed that the test activity involves simulating some actions that are normally identified as events triggering the intervention of security guards.

Table 1 lists all the possible tests that can be performed, taking into consideration the scenarios defined in sections 0 and 0 and the variations on executing conditions defined above. Please note that some variations cannot be applied to all scenarios; for example, in the current system setup, barrier crossing and counter flow events can be identified only under low resolution cameras. Each test in the table is numbered and identified by a unique ID generated by composing the type of scenario with the applied variations. For example N2-FHI identifies the test where the negative scenario number **N2** is performed by an actor with **F**lashy clothing, under **H**igh resolution cameras in an **I**ntermediately crowded environment.

Table 1 - Possible tests performed

Test Number	Test ID	Clothing of actors		Resolution of the camera		Crowding of the environment	
		Flashy	Gray	Low	High	Intermediate	High
T1	N1 – FLI	X		X		X	
T2	N1 - FHI	X			X	X	
T3	N1 - GLI		X	X		X	
T4	N1 - GHI		X		X	X	
T5	N2 - FLI	X		X		X	
T6	N2 - FHI	X			X	X	
T7	N2 - GLI		X	X		X	
T8	N2 - GHI		X		X	X	
T9	N3 - FLI	X		X		X	
T10	N3 - FHI	X			X	X	
T11	N3 - GLI		X	X		X	
T12	N3 - GHI		X		X	X	
T13	N4 - FLI	X		X		X	
T14	N4 - FHI	X			X	X	

T15	**N4 - GLI**		X	X		X	
T16	**N4 - GHI**		X		X	X	
T17	**P1 - FLI**	X		X		X	
T18	**P1 - FLH**	X		X			X
T19	**P1 - FHI**	X			X	X	
T20	**P1 - FHH**	X			X		X
T21	**P1 - GLI**		X	X		X	
T22	**P1 - GLH**		X	X			X
T23	**P1 - GHI**		X		X	X	
T24	**P1 - GHH**		X		X		X
T25	**P2 - FLI**	X		X		X	
T26	**P2 - FHI**	X			X	X	
T27	**P2 - GLI**		X	X		X	
T28	**P2 - GHI**		X		X	X	
T29	**P3 - FLI**	X		X		X	
T30	**P3 - FHI**	X			X	X	
T31	**P3 - GLI**		X	X		X	
T32	**P3 - GHI**		X		X	X	
T33	**P4 - FDB**	Operator gives feedback through the UI					
T34	**P4 - NOF**	Operator does not give feedback through the UI					
T35	**P5 – FLI**	X		X		X	
T36	**P5 - FHI**	X			X	X	
T37	**P5 - GLI**		X	X		X	
T38	**P5 - GHI**		X		X	X	
T39	**P6 - FLI**	X		X		X	
T40	**P6 - FHI**	X			X	X	
T41	**P6 - GLI**		X	X		X	

T42	**P6 - GHI**		X		X	X	
T43	**P7 - P1**						
T44	**P7 - P2**						
T45	**P7 - P3**						
T46	**P7 - P5**						
T47	**P7 - P6**						

4. NEGATIVE SCENARIOS

The possible scenarios that may occur and should therefore be analyzed are different. Some of the possible scenarios are defined as negative because they represent situations similar to those classified as events, but in reality they are not. The system must be able to not mistakenly identify them as events. Negative scenarios are borderline situations to test the absence of false positives.

The possible variations to be considered for each scenario are indicated in the description of the scenario and generate different tests according to the table reported at the end of section 3.

To collect data for scenario P4, defined in section 0, it is suggested that, before performing the scenarios indicated in this section, the time interval Td after which AddPriv must delete non security related video is set to a manageably short value, indicatively not greater than the time allocated to run all the scenarios.

Scenario N1
The person is near to the controlled barrier but is not crossing it
Purpose
Verify that the system does not mistakenly recognize as barrier crossing events some borderline situations such as: walking along the protected line without crossing it, crossing a line which is the continuation of the barrier but it's not the protected line.
Preconditions, environment and applicable variations
Event Cameras: C or D or E (see Figure 1).
Number of actors: one.
Actor walks around before and after the event: no.
Variations that can be applied: clothing of the actors and crowding of the environment.
Execution steps
An actor will reproduce the scenario. The event must be represented in the field of view of cam C or cam D or cam E, which are positioned in correspondence with a barrier. The actor will walk along the protected line without crossing it and will also cross a line which is the continuation of the barrier but it's not the protected line.
Expected behavior
The system should not recognize the performance of the actor as an event

Scenario N2
The person takes a couple of steps back but doesn't walk continuously in the opposite flow direction
Purpose
Verify that the system does not mistakenly recognize as counter-flow event the simple situation of a person taking just a couple of steps back with respect to the flow.
Preconditions, environment and applicable variations
Event Cameras: C or D (see Figure 1).
Number of actors: one.
Actor walks around before and after the event: no.
Variations that can be applied: clothing of the actors and crowding of the environment.
Execution steps
An actor will reproduce the scenario. The event must be represented in the field of view of cam C or cam D, which include the snake areas. The actor will enter the snake in the right direction, then will stop, will take a couple of steps back and then will go on again.
Expected behavior
The system should not recognize the performance of the actor as an event

Scenario N3
The person stoops, as if to pick up something, while in a queue
Purpose
Verify that the system does not mistakenly recognize as barrier crossing or counter-flow event the simple situation of a person stooping down, as if to pick up something.
Preconditions, environment and applicable variations
Event Cameras: C or D (see Figure 1).
Number of actors: one.
Actor walks around before and after the event: no.
Variations that can be applied: clothing of the actors and crowding of the environment.

Execution steps
An actor will reproduce the scenario. The event must be represented in the field of view of cam C or cam D, which include the snake areas. The actor will enter the snake in the right direction, then will stop, will stoop down as if to pick up something, and then will go on again.
Expected behavior
The system should not recognize the performance of the actor as an event

Scenario N4
The person is not too far from the suitcase, for a small period of time, and still keeping an eye on it
Purpose
Verify that the system does not mistakenly recognize as left luggage event the situation of a person leaving a bag alone to do something nearby for a small period of time.
Preconditions, environment and applicable variations
Event Cameras: any camera.
Number of actors: one.
Actor walks around before and after the event: no.
Variations that can be applied: clothing of the actors, crowding of the environment, and resolution of the camera under which the scenario is represented.
The timeout Te for triggering the left luggage event is known.
Execution steps
An actor will reproduce the scenario. The actor will enter the field of view of a camera, possibly on one side, with a bag. The actor will leave the bag unattended, while keeping an eye on it, and will move to the other side of the camera's field of view for a certain time interval T<Te before going back near the luggage.
Expected behavior
The system should not recognize the performance of the actor as an event

5. POSITIVE SCENARIOS

The positive scenarios represent situations in which an event occurs, where each event can be an abandoned luggage, or a person walking in the opposite direction to the normal path or a person crossing a barrier that prevents access to an area not accessible to the public.

The actor playing each scenario will first walk around across the field of view of most cameras, then will represent the event and then will walk around again to allow testing also the route reconstruction functionality. During the execution of each scenario, to properly check Route Reconstruction module, it will be needed to keep track of the cameras in which the actor shows up.

The possible variations to be considered for each scenario are indicated in the description of the scenario and generate different tests according to the table reported at the end of section 3.

To collect data for scenario P4 it is suggested that, before performing the first three scenarios indicated in this section, the time interval Td after which AddPriv must delete non security related video is set to a manageably short value, indicatively not greater than the time allocated to run all the scenarios.

Scenario P1
The person leaves her luggage unattended
Purpose
Verify that the system correctly recognizes the left luggage event
Preconditions, environment and applicable variations
Event Cameras: any camera.
Number of actors: one.
Number of operators: one.
Actor walks around before and after the event: yes.
Variations that can be applied: clothing of the actors, crowding of the environment, and resolution of the camera under which the scenario is represented.
The timeout Te for triggering the left luggage event is known.

Execution steps
An actor will reproduce the scenario. The actor will first walk around, with a bag, across the field of view of some cameras and another person (the operator) will keep track of the list of cameras in which the actor shows up. The actor will then enter, with the bag, the field of view of a camera and will leave the bag unattended. After leaving the bag the actor will walk around again across some other cameras, and will not go back near the luggage for at least a time interval T>2*Te. The operator again will keep track of the list of cameras in which the actor shows up.
Expected behaviour
The system should be able to recognize the performance of the actor as a left luggage event, and report it to the operator with a clear indication in the User Interface.

Scenario P2
The person crosses a forbidden barrier or enters into an area not allowed to the public
Purpose
Verify that the system correctly recognizes the barrier crossing event
Preconditions, environment and applicable variations
Event Cameras: C or D or E (see Figure 1).
Number of actors: one.
Number of operators: one.
Actor walks around before and after the event: yes.
Variations that can be applied: clothing of the actors, crowding of the environment.
The (virtual) lines that are configured in the system as barriers are known.
Execution steps
An actor will reproduce the scenario. The event must be represented in the field of view of cam C or cam D or cam E, which are positioned in correspondence with a barrier. The actor will first walk around across the field of view of some cameras and another person (the operator) will keep track of the list of cameras in which the actor shows up. The actor will then enter the field of view of one of the event cameras and will cross at least one of the lines defined as barriers. Then the actor will walk around again across some other cameras and the operator again will keep track of the list of cameras in which the actor shows up.

Expected behaviour
The system should be able to recognize the actor's performance as a barrier crossing event and report it to the operator with a clear indication in the User Interface

Scenario P3
The person walks in the opposite direction with respect to the correct flow
Purpose
Verify that the system correctly recognizes the counter-flow event
Preconditions, environment and applicable variations
Event Cameras: C or D or E (see Figure 1).
Number of actors: one.
Number of operators: one.
Actor walks around before and after the event: yes.
Variations that can be applied: clothing of the actors, crowding of the environment.
Execution steps
An actor will reproduce the scenario. The event must be represented in the field of view of cam C or cam D, which are positioned in correspondence with snake areas. The actor will first walk around across the field of view of some cameras and another person (the operator) will keep track of the list of cameras in which the actor shows up. The actor will then enter the field of view of one of the event cameras and will enter the snake queue in the correct direction. When near the end of the queue the actor will change direction and walk back again on her steps through the whole queue. Then the actor will walk around again across some other cameras and the operator again will keep track of the list of cameras in which the actor shows up.
Expected behaviour
The system should be able to recognize the performance of the actor as a counter-flow event, and report it to the operator with a clear indication in the User Interface

Scenario P4
The system will delete the videos which do not relate to security-relevant events
Purpose
Verify that the system correctly deletes video fragments which do not contain or are associated to recognized security-related events
Preconditions, environment and applicable variations
Event Cameras: any
Number of actors: N/A.
Number of operators: one.
Actor walks around before and after the event: N/A.
Variations that can be applied: the operator gives feedback or not to the events and routes
The time interval Td after which AddPriv must delete non security related video is known.
At least one of the scenarios N1-N4 and at least one of the scenarios P1-P3 have been executed
Execution steps
This scenario must be executed after the successful execution of at least one of the scenarios N1-N4 and at least one of the scenarios P1-P3. As soon as the prerequisite scenarios have been executed, the operator checks that the AddPriv system contains some recorded video which is associated to security-related events, let's call this video set V1, and some video which is not in any way associated to events, call it V2. After a time interval T>Td the operator will check again for the existence of video frames in the V1 set and in the V2 set.
Expected behaviour
ADDPRIV should delete all and only the saved video frames that are older than a configurable time interval and that are not marked as Events or part of a Route Reconstruction tree. Therefore, when the configured time interval is passed, video set V1 should not have changed, while video set V2 should be empty. Video which representing an event or a node in the RR tree related to that event should not be deleted even if feedback on the event and / or tree (see scenario P7) has not yet been provided by the operator.

Scenario P5
Two events occur at the same time under different cameras
Purpose
Verify that the system correctly recognizes two events occurring under at the same time under different cameras.
Preconditions, environment and applicable variations
Event Cameras: C or D or E (see Figure 1) plus any other camera.
Number of actors: two
Number of operators: one.
Actor walks around before and after the event: yes.
Variations that can be applied: clothing of the actors, crowding of the environment
The timeout Te for triggering the left luggage event is known
Execution steps
Two actors, A and B, will reproduce the scenario. Actor A will first walk around, with a bag, across the field of view of some cameras. Also actor B walks around across some cameras, but without a bag. Another person (the operator) will keep track of the list of cameras in which each of the two actors shows up. After leaving the bag actor A will walk around again across some cameras. After crossing the barrier actor B will walk around again across some other cameras. The operator will keep track of the list of cameras in which each of the two actors shows up.
Expected behaviour
The system should be able to recognize the performance of the actors as two events: a left luggage event and a barrier crossing event, and report them (separately) to the operator with a clear indication in the User Interface.

Scenario P6
Two events occur at the same time under the same camera
Purpose
Verify that the system correctly recognizes two events occurring under at the same time under the same camera

Preconditions, environment and applicable variations
Event Cameras: C or D or E (see Figure 1)
Number of actors: two
Number of operators: one.
Actor walks around before and after the event: yes.
Variations that can be applied: clothing of the actors, crowding of the environment
The timeout Te for triggering the left luggage event is known
Execution steps
Actor A will then enter, with the bag, the field of view of the event camera and will leave the bag unattended. At the same time, and under the same camera, actor B will cross at least one of the lines defined as barriers. After leaving the bag actor A will walk around again across some cameras. After crossing the barrier actor B will walk around again across some other cameras. The operator will keep track of the list of cameras in which each of the two actors shows up
Expected behaviour
The system should be able to recognize the performance of the actors as two events: a left luggage event and a barrier crossing event, and report them (separately) to the operator with a clear indication in the User Interface

Scenario P7
The operator can view the route travelled by the actor triggering an event
Purpose
Verify that the system correctly shows the event and allows the operator to view and give feedback on it and on the route travelled by the triggering person
Preconditions, environment and applicable variations
Event Cameras: depends on the selected scenario
Number of actors: depends on the selected scenario
Number of operators: one.
Actor walks around before and after the event: yes.
Variations that can be applied: different scenarios, other variations depend on the selected one
Other preconditions: see the selected scenario

Execution steps
One of the scenarios P1, P2, P3, P5, or P6 is successfully executed. The operator uses the UI to view the event(s) and to give feedback on it. Then the operator, again through the UI, views and gives feedback on the route taken by the actor(s) before and after the event(s)
Expected behavior
The operator can view the event and the route taken by the actor(s) and reconstructed by the system. The operator can provide feedback on both the event and the route and the system. Takes it into account by marking the related video segments as relevant

6. Conclusion

This document defines a plan for functional testing of the AddPriv system in a real environment. Several scenarios have been defined to validate typical situations in which the system is expected to work.

Variations to the scenarios have been also identified and the possible tests have been enumerated as the combinations of scenarios with all the possible variations.

All the combinations identifying single tests have been enumerated, in the table at the end of section 3, with the purpose of assigning a unique ID to each combination, representing a single possible test to be executed.

The number of tests that will actually be executed will depend on various factors, such as available time and the possibility of verifying all the possible variations. Anyway for a complete functional verification of the system at least one test for each scenario should be executed.

The results of the executed tests will be reported in the next deliverable D5.4.