1. Histogram characterization

In order to identify a single person detected by different cameras or one person in a group, we need to characterize the tracked people. Based on the results presented in [Perez02], [Nummiaro03], [Qingshan02] and [Schiele96], we have chosen to use *colour histograms*. To realize this *histogram characterization*, we have to build a colour histogram on each received *segment*. A *segment* is the mask obtained after the background subtraction and the connexion of close components of pixels. Such zones are considered to be in the same *segment* in case of overlapping of the bounding boxes and should represent one person.

Below we describe this development in more details.

The histogram characterization is used to search a *target* in the following cases:

- When someone has left the field of view of a camera and appears in the field of view of a neighbour camera.
- Right after two persons have crossed each other in the field of view of a camera.
- When someone reappears after a partial or total occlusion by a stationary object in the field of view of a camera.

In these three cases, the tracking cannot be performed by only using the position of a person. A comparison between the colour histogram of the person built from the view of the camera and the histogram contained in a list of *targets* established before will allow such tracking.

During the previous ViBIH project, we had already done histogram characterization. Nevertheless, it was not robust enough to achieve the requirement of the current project, particularly for the multi-camera tracking, since it used only the histogram of the last image for comparison. To improve the robustness of the characterization, we propose:

- An adaptive colour histogram based on several sequential images of the *target* and not only the last one.
- A *histogram model*, which comprises several histograms, related to different person's positions (i.e., since the same person seen from the front and from the back are not always composed by the same colours, the histograms built from the front and back views could be different).

1.1 Description of the histogram model

The proposed histogram is based on the colour vector of a defined colour space (RGB, HSV...). The entire colour space is discretised into a number of bins, each one defined by its central colour vector C(R,G,B) or C(H,S,V). The discretisation along the 3 axis can be different in order to give, for example, more weight to the hue in a HSV histogram. Each bin is normalized according to the number of computed pixels. In this way, a bin represents the ratio of the image pixels which have a colour close to its colour and the sum of all bins of a histogram is equal to 1.

Each *target* contains a *histogram model* composed by a predefined number of histograms. Each histogram in the model has a weight which can vary in the time and which indicates how much this histogram is representative of the detected person compared to the other histograms of the model.

The Figure 1 shows an example of a histogram model composed by 3 histograms in the HSV colour space.

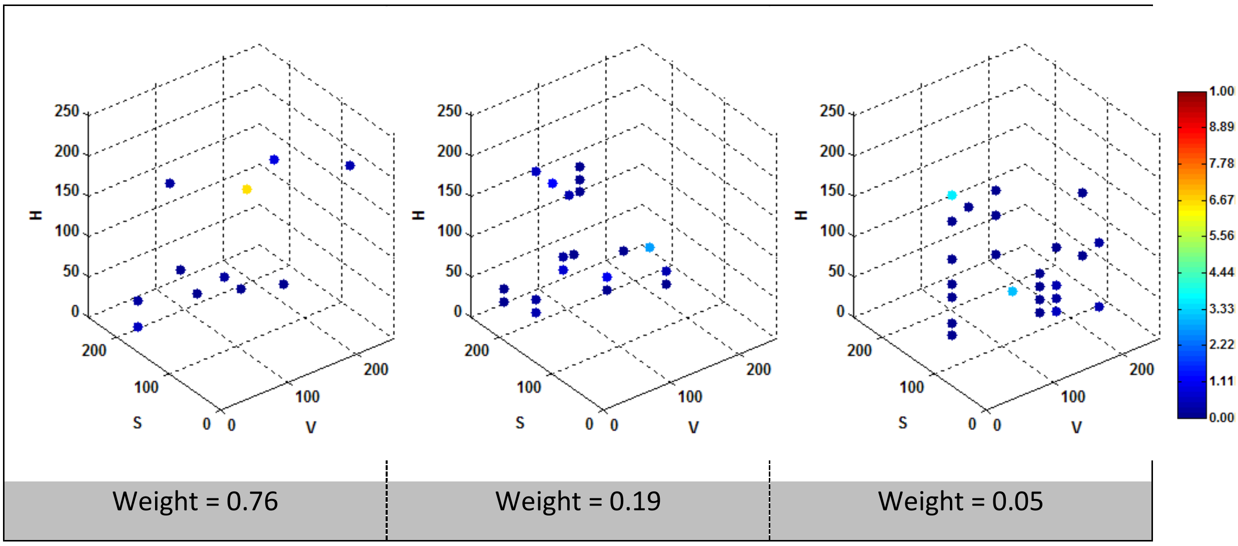

Figure 1: Representation of an example of a histogram model composed by 3 histograms in HSV colour space (255x255x255 discretised to 16x8x4). The non-empty bins are represented as coloured dots, being each colour the value of the bin is indicated in the colour bar in the right.

1.2 Histogram update method

To be continuously adapted, the histogram model must be updated for each new image which contains the detected person. The update is done as follow: A single histogram is computed for the new image and then compared to the histograms of the model. If the result of the comparison is accurate enough (i.e., a histogram in the model is similar to the new one), the most similar histogram of the model is updated with the new histogram and its weight is increased. On the other case, the histogram with the lowest weight is replaced by the new one and its weight is set to zero (see Figure 2).

To update a single histogram with a new one, each bin value b_i of the model histogram is updated with the value of the new bin n_i, according to the following formula:

$b_i = (1-\alpha)b_i + \alpha \cdot n_i$, where α is the adaptive factor of the update method.

The weights of the histogram in the model are the representativeness of the detected person by this histogram. The more a histogram is updated (i.e., selected as the most similar histogram), the more its weight will grow. The new weight of an updated histogram is:

$w = (1-\beta)w$, where w is the weight and β the *update weight factor*

or $w = (1-\beta)w + \beta$, in case the histogram was already updated.

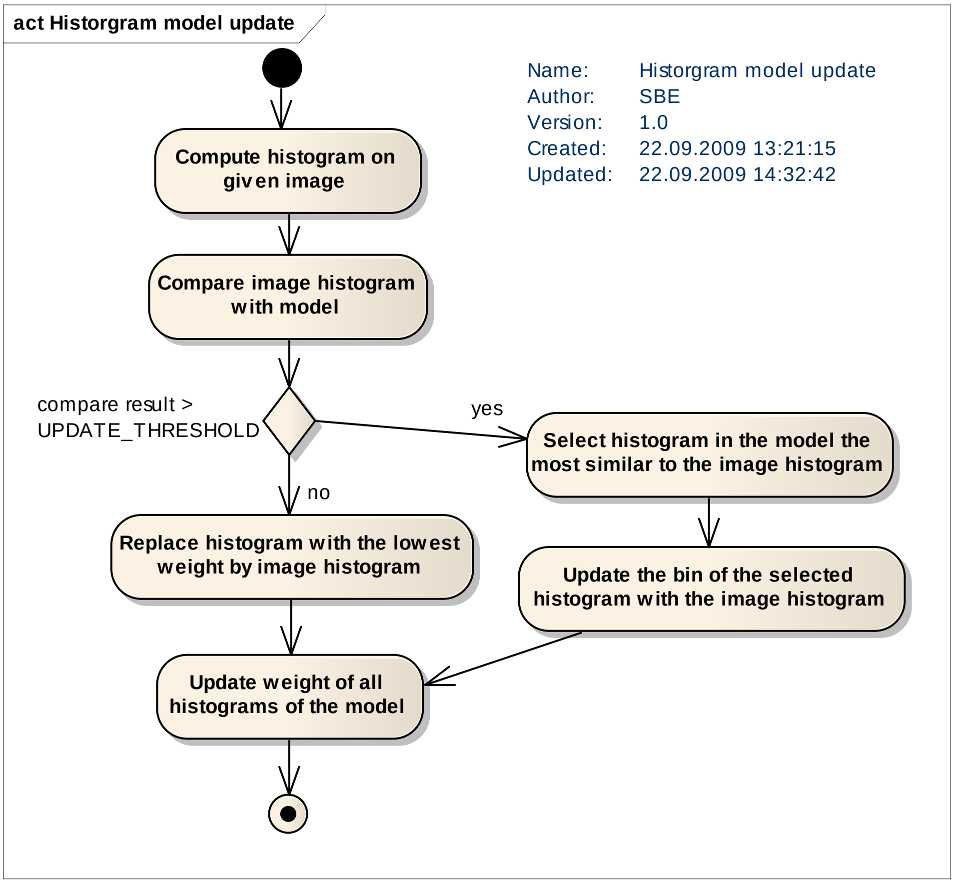

Figure 2: Activity diagram of the update method on histogram model

1.3 Histogram comparison method

To compare two different histograms we use the method proposed by Swain and Ballard in [Swain91]. This is a simple sum of the intersections of two histograms which returns a value between 0 and 1. This result is 1 for identical histograms and 0 for completely different one.

We also need a method to compare a single histogram with a *histogram model*. This is simply done by making the comparison between the single histogram with all the histograms of the model and returning the greater result.

1.4 Histogram implementation

In our implementation, we have a *target* object for each different tracked people. Each *target* contains, among other data, the *histogram model*. A *histogram model* is composed by several adaptive colour histograms of a defined colour type. Two possibilities were implemented: one based on the RGB colour space and the other on the HSV colour space. However, the model could easily be extended to other colour spaces. Our implementation uses the OpenCV library for the histogram and images structures.

The main class of the histogram implementation is *HistogramModel.* Each target has an object of this type for its description. This class contains a defined number of histogram, each one is an object of the abstract class *Histogram* which is overwritten here by to subclasses: *HistogramRGB* and *HistogramHSV* for the two colour spaces implemented (see Figure 3).

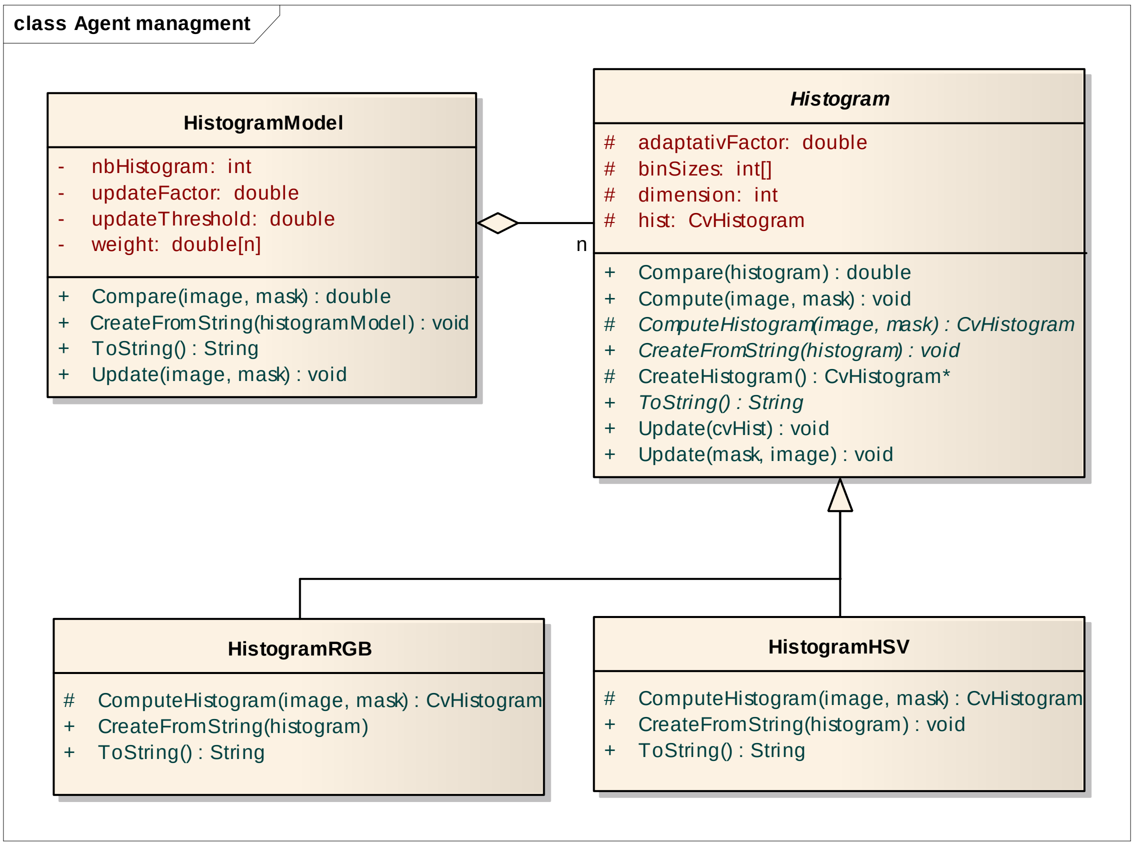

Figure 3: Class diagram of the histogram model of a target

The class *HistogramModel* contains a method to update a class instance with a new *segment*. To compare a model to the histogram of a detected person, the class *HistogramModel* provides the method *Compare,* which has the same parameters as the *Update*. This method returns a double value in the range [0:1] which indicates the similarity between the histogram built from the given image and the current model. This class contains also a method to serialize the histogram into a string or to create a new *histogram model* based on a serialized one. ~~Those two methods are mainly used to send a *target* from a camera to another one.~~

1.5 Target Identification

Sometimes a *target* is present in the system of a camera even though if nobody is visible by the camera. This happens when:

- The person is occluded by a piece of furniture.
- The person is leaving or has left the field of view of a neighbour camera and is supposed to arrive in the field of view of the current camera.

For each new image, the position and the colour histogram are computed for each visible person. It is still needed to link each point in the image to a target. This is the goal of the identification part.

The identification part has, on one hand, a list of the current people found in the image represented by their head position and their histogram, and on the other hand, a list of *targets* composed by the last head position and *histogram models*. The Figure 5 shows an example of the arrays *Current people list* and *Current targets list*.

The identification process comprises two main steps. The first step compares the Euclidian distance between the head position of all current people with the last head position stored in the *Current targets list*. This method is detailed in the Figure 4. All pairs of person-target obtained by this method are accepted only if the computed distance is lower than a threshold. See the example of Figure 5, array *Distance head position*.

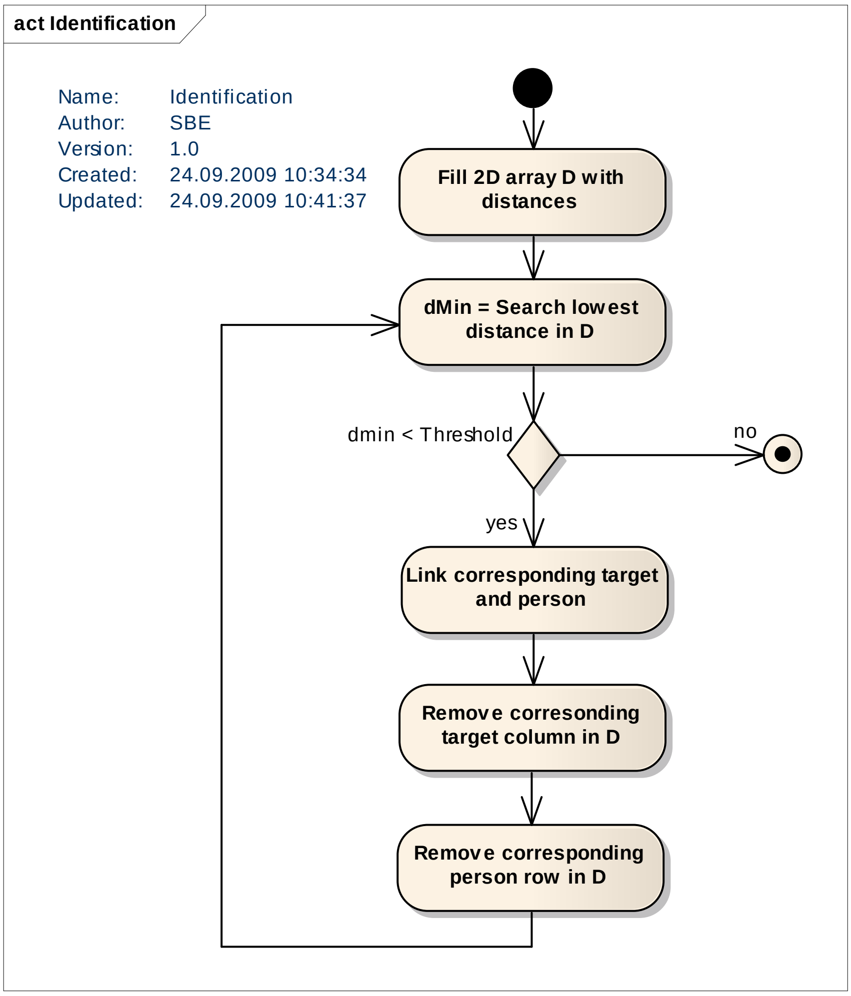

Figure 4: Activity diagram of the identification step using distance between head positions

The second step measures the similarity between the colour histogram of *Current people list* and the colours *histogram models* that are in the *Current targets list*. All pairs person-target obtained by this method are accepted only if the comparison is higher than a threshold. See the example of the Figure 5, array *Histogram comparison*.

The identification is done by these two first steps. Some *targets* are linked to detected persons found in the image while others are not. Each *target* contains a timestamp which indicates the last time it has been seen by the camera. This timestamp is updated for each target found in the image. For the other targets, if their timestamp is too old, the *target*s are deleted. So, people without *target* are considered as new one. For them, a new *target* is created using their head position and their colour histogram.

Current people list

ID	Head pos.	Histogram
1	(x,y)	$\mathcal{H}$(h,s,v)
2	(x,y)	$\mathcal{H}$(h,s,v)
3	(x,y)	$\mathcal{H}$(h,s,v)
4	(x,y)	$\mathcal{H}$(h,s,v)

Current targets list

ID	Prev. head pos.	Histogram model
A	(x,y)	M($\mathcal{H}$,$\mathcal{H}$,$\mathcal{H}$)
B	(x,y)	M($\mathcal{H}$,$\mathcal{H}$,$\mathcal{H}$)
C	(x,y)	M($\mathcal{H}$,$\mathcal{H}$,$\mathcal{H}$)
D	(x,y)	M($\mathcal{H}$,$\mathcal{H}$,$\mathcal{H}$)
E	(x,y)	M($\mathcal{H}$,$\mathcal{H}$,$\mathcal{H}$)

Distance head position

	A	B	C	D	E
1	95	18	151	232	197
2	143	182	107	128	130
3	125	136	206	64	177
4	216	108	198	265	249

Histogram comparison

	A	C	E
2	32%	86%	26%
4	16%	29%	46%

Results

Target	Person
A	
B	1
C	2
D	3
E	
	4

Figure 5: Identification process.
The first line shows the list of detected person and the list of person description (*targets*). The second line shows the matching process which compare 1) the head distance (threshold=100) and 2) the histogram similarities (threshold=60%). In the results array targets linked to a person are in green, targets not linked in red and new targets in blue.

2. Results and analysis for the extended algorithms

This section presents and analyses the outcomes of the implementation of the fall detection application for multi-camera systems.

2.1 Background subtraction method improvement results

To test the new background subtraction, the mask and the computation time were compared for the two methods, the original (full Lab images) and the extended one (only the luminance L). To measure the quality of the computed mask, a background subtraction has been first performed "by hand" on an image sequence, which provide us with an ideal mask. Comparing this mask to the one computed by the two methods, we can extract:

- False foreground pixels: pixels expected in the background set in the foreground.
- False background pixels: pixels expected in the foreground set in the background.
- False pixels: the sum of the false foreground pixels and the false background pixels.

This test has been done on a sequence of 250 images at a resolution of 320x240 pixels representing someone walking across a hall (see Figure 6).

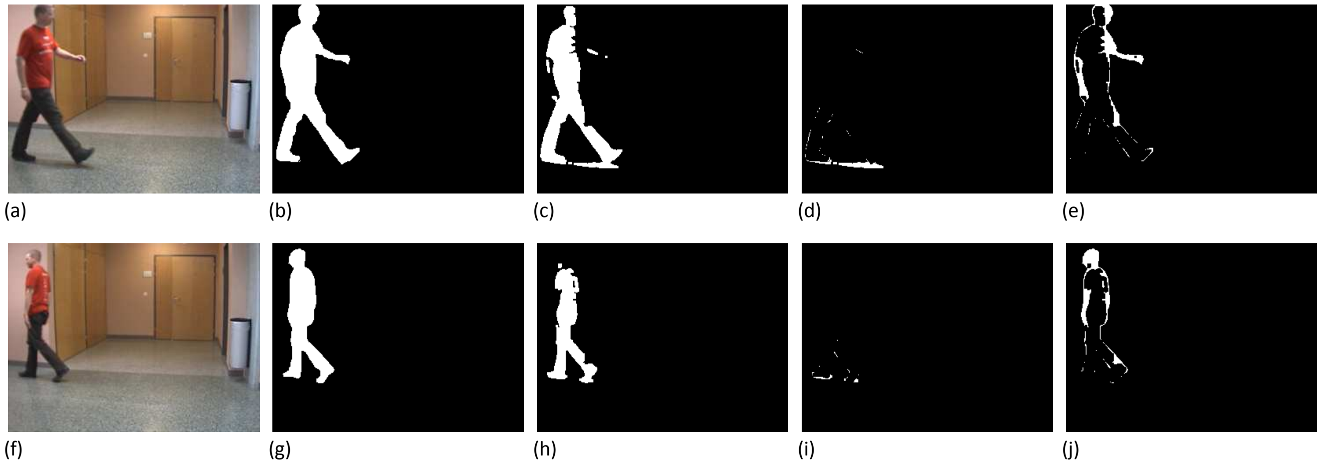

Figure 6: Two images of the background subtraction test sequence
(a) (f) Camera view, (b) (g) Ideal mask, (c) (h) Computed mask, (d) (i) False foreground pixels (in white), (e) (j) False background pixels (in white)

The results were then normalized according to the total number of pixels in foreground or in background. The processing time of the background subtraction methods have also been measured. The results are presented in Table 1.

Method used	Full Lab	Only L
False foreground pixels AVG	4.82%	6.66%
False background pixels AVG	1.24%	0.73%
False pixels AVG	1.33%	0.88%
Frame per second	15.79	26.83

Table 1: Results of the comparison of the original (Full Lab) and the extended (Only L) background subtraction methods

The original method performs better when looking only at the false foreground pixels. This is probably due to a better subtraction of shadows which mainly change the luminance. In contrast with false foreground, we obtain better results with the extended method for the false background. The latter results are more important for our application, since false background pixels can reduce the rectangle built by the segmentation. For this reason, we can claim that the extended method performs better for our purposes. Moreover, the extended method considerably speeds up the application, being able to process more frames per second.

2.2 Histogram characterization tests and results

To know if our characterization solution based on colour histogram is suitable for the goals of our project, we have carried out several tests:

- Luminosity change.
- Partial occlusion of a person.
- Multi-camera histogram comparison.

These tests and their results are presented in the following paragraphs.

2.3 Test of luminosity change

The luminosity of the field of view for every camera placed in the same environment can be different. For this reason, when someone traverses from one camera to another one, it can be captured under a different lighting. In such a case, does the histogram comparison give us enough good results? That is the question we try to answer with the luminosity test presented here.

To carry out this luminosity test, we have filmed 15 persons under two different lighting conditions: bright and dark. Then the histogram models have been computed l for each sequence. Finally the histogram models for the different lightings have been compared. The results are presented in Table and Table 3.

		bright														
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
dark	1	**35%**	22%	20%	14%	20%	36%	13%	40%	7%	5%	40%	9%	15%	13%	14%
	2	43%	**62%**	27%	40%	60%	37%	39%	34%	27%	0%	1%	3%	3%	23%	38%
	3	0%	1%	**26%**	10%	4%	2%	3%	1%	1%	0%	0%	0%	1%	0%	1%
	4	0%	0%	29%	**9%**	2%	2%	1%	5%	0%	0%	4%	0%	4%	0%	0%
	5	17%	21%	46%	29%	**21%**	24%	16%	11%	13%	0%	1%	3%	3%	4%	10%
	6	43%	29%	22%	15%	26%	**46%**	16%	25%	7%	13%	19%	12%	21%	19%	14%
	7	21%	27%	46%	33%	28%	34%	**27%**	13%	14%	0%	1%	6%	7%	11%	11%
	8	26%	30%	31%	37%	37%	24%	29%	**49%**	23%	1%	27%	6%	14%	9%	23%
	9	9%	14%	37%	28%	20%	14%	15%	9%	**24%**	12%	1%	3%	3%	2%	9%
	10	6%	0%	4%	0%	0%	0%	0%	13%	5%	**5%**	13%	1%	6%	0%	0%
	11	22%	1%	1%	1%	1%	15%	2%	39%	1%	17%	**60%**	20%	35%	16%	10%
	12	35%	14%	4%	5%	15%	36%	21%	20%	2%	18%	37%	**30%**	34%	29%	11%
	13	20%	3%	4%	4%	4%	13%	1%	28%	0%	13%	37%	10%	**32%**	7%	5%
	14	55%	38%	11%	13%	44%	53%	23%	27%	10%	13%	21%	16%	21%	**31%**	24%
	15	43%	49%	13%	20%	45%	37%	33%	44%	25%	13%	27%	22%	25%	43%	**61%**

Table 2: Comparison array of the 15 bright sequences
Red: 1st result, orange: 2nd result, green 3rd result, bold: expected best result

		dark														
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
bright	**1**	**35%**	43%	0%	0%	17%	43%	21%	26%	9%	6%	22%	35%	20%	55%	43%
	2	22%	**62%**	1%	0%	21%	29%	27%	30%	14%	0%	1%	14%	3%	38%	49%
	3	20%	27%	**26%**	29%	46%	22%	46%	31%	37%	4%	1%	4%	4%	11%	13%
	4	14%	40%	10%	**9%**	29%	15%	33%	37%	28%	0%	1%	5%	4%	13%	20%
	5	20%	60%	4%	2%	**21%**	26%	28%	37%	20%	0%	1%	15%	4%	44%	45%
	6	36%	37%	2%	2%	24%	**46%**	34%	24%	14%	0%	15%	36%	13%	53%	37%
	7	13%	39%	3%	1%	16%	16%	**27%**	29%	15%	0%	2%	21%	1%	23%	33%
	8	40%	34%	1%	5%	11%	25%	13%	**49%**	9%	13%	39%	20%	28%	27%	44%
	9	7%	27%	1%	0%	13%	7%	14%	23%	**24%**	5%	1%	2%	0%	10%	25%
	10	5%	0%	0%	0%	0%	13%	0%	1%	12%	**5%**	17%	18%	13%	13%	13%
	11	40%	1%	0%	4%	1%	19%	1%	27%	1%	13%	**60%**	37%	37%	21%	27%
	12	9%	3%	0%	0%	3%	12%	6%	6%	3%	1%	20%	**30%**	10%	16%	22%
	13	15%	3%	1%	4%	3%	21%	7%	14%	3%	6%	35%	34%	**32%**	21%	25%
	14	13%	23%	0%	0%	4%	19%	11%	9%	2%	0%	16%	29%	7%	**31%**	43%
	15	14%	38%	1%	0%	10%	14%	11%	23%	9%	0%	10%	11%	5%	24%	**61%**

Table 3: Comparison array of the 15 dark sequences
Red: 1st result, orange: 2nd result, green 3rd result, bold: expected best result

It can be observed that 11 results over 30 are the expected one, 15 if the 2nd best result is took and 18 if the 2nd and the 3rd best results are took. These tests were extended to also cover darker sequences (fast night), but the comparison results were quite bad. Therefore it can be concluded that the luminosity of the field of view has an important effect on the histogram characterization, but it can even though be used as differentiated characteristic as long as the environment is not too dark. This effect can be explained by the fact that the colour appears in greyscale when it is not lighted up enough.

2.4 Test of partial occlusion of a person

Another goal pursued by the histogram characterization is to retrieve the person's tracking after his occlusion. To test if our histogram model can achieve this goal, we have used the bright sequence of the single camera luminosity change test. For each of these 15 videos, we have built two histogram models: one computed with the entire rectangles and the other one built with only the upper-half part of the rectangle (which is representative of the majority occlusions that can occur in a flat). Finally, the different histogram models were compared.

		half occluded														
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
entire	**1**	**49%**	29%	10%	3%	38%	45%	6%	16%	0%	1%	21%	11%	30%	25%	24%
	2	29%	**61%**	14%	14%	43%	46%	6%	3%	0%	1%	3%	13%	16%	35%	55%
	3	11%	21%	**49%**	37%	9%	21%	6%	3%	0%	2%	3%	12%	19%	5%	14%
	4	13%	34%	23%	**52%**	13%	17%	4%	3%	0%	2%	3%	14%	21%	7%	27%
	5	20%	49%	16%	27%	**57%**	49%	7%	3%	0%	2%	3%	14%	18%	39%	48%
	6	28%	21%	15%	6%	38%	**57%**	13%	10%	0%	2%	10%	18%	33%	28%	19%
	7	15%	41%	12%	24%	23%	26%	**31%**	3%	0%	1%	3%	35%	14%	20%	37%
	8	31%	45%	7%	14%	21%	20%	1%	**41%**	0%	0%	42%	9%	29%	19%	44%
	9	14%	33%	9%	18%	11%	13%	1%	3%	**42%**	42%	3%	9%	11%	9%	31%
	10	13%	0%	0%	2%	0%	0%	1%	7%	54%	**54%**	8%	1%	18%	0%	0%
	11	29%	1%	1%	2%	1%	1%	5%	54%	0%	0%	**61%**	5%	32%	1%	1%
	12	15%	3%	3%	1%	4%	6%	38%	11%	0%	0%	12%	**42%**	32%	6%	3%
	13	20%	2%	3%	6%	3%	4%	8%	19%	0%	1%	23%	9%	**43%**	1%	2%
	14	23%	18%	2%	1%	44%	32%	14%	9%	0%	0%	9%	16%	23%	**61%**	27%
	15	29%	57%	7%	14%	33%	25%	1%	8%	0%	0%	8%	9%	16%	36%	**73%**

Table 4: Comparison array of the 15 normal sequences against half occluded one
Red: 1st result, orange: 2nd result, green 3rd result, bold: expected best result

The results, presented in Table 4, indicate that 13 results over 15 are the expected one. Therefore, it can be concluded that using histogram characterization to solve occlusion is an accurate solution.

2.5 Test of multi-camera histogram comparison

The main goal of the histogram characterization is to recognize people when they traverse from the field of view of a camera to a neighbour one. To test this scenario, 14 sequences of different persons who are walking and crossing the field of view of both cameras were filmed with two cameras. Then, the histogram models were computed for each sequence. Finally all these histogram models were compared. The results are presented in Table 4.

	1A	1B	2A	2B	3A	3B	4A	4B	5A	5B	6A	6B	7A	7B	8A	8B	9A	9B	10A	10B	11A	11B	12A	12B	13A	13B	14A	14B
1A	x	**58**	36	37	48	59	45	44	41	45	33	37	36	38	4	2	5	1	4	3	5	9	30	37	4	3	17	20
1B	**58**	x	45	43	58	66	48	52	53	58	41	43	42	44	0	1	1	0	0	4	7	18	39	45	1	3	27	32
2A	36	45	x	**63**	43	46	46	45	42	41	44	43	43	41	2	3	5	4	2	5	6	13	39	41	30	26	12	16
2B	37	43	**63**	x	40	42	41	45	40	43	44	49	36	45	3	6	11	10	3	5	6	14	35	41	27	33	13	17
3A	48	58	43	40	x	**71**	53	47	49	49	41	39	41	41	0	1	0	0	0	0	1	11	40	43	0	0	8	13
3B	59	66	46	42	**71**	x	55	51	53	54	43	42	42	43	0	1	1	0	0	2	5	13	40	45	1	2	12	16
4A	45	48	46	41	53	55	x	**64**	49	45	46	42	43	42	14	11	15	11	25	24	14	23	47	49	14	12	19	24
4B	44	52	45	45	47	51	**64**	x	44	48	43	48	42	46	12	14	14	13	22	26	14	22	45	52	13	15	22	26
5A	41	53	42	40	49	53	49	44	x	**67**	40	40	39	42	0	1	1	0	0	5	31	33	36	42	1	3	17	21
5B	45	58	41	43	49	54	45	48	**67**	x	37	42	37	43	0	1	1	0	0	5	23	40	36	44	1	3	20	26
6A	33	41	44	44	41	43	46	43	40	37	x	**74**	40	39	7	10	42	40	8	9	7	15	39	41	9	18	13	17
6B	37	43	43	49	39	42	42	48	40	42	**74**	x	38	44	5	10	41	43	7	10	7	14	37	44	9	19	15	18
7A	36	42	43	36	41	42	43	42	39	37	40	38	x	**60**	34	30	4	2	3	4	5	11	36	37	4	3	11	14
7B	38	44	41	45	41	43	42	46	42	43	39	44	**60**	x	23	28	5	3	2	5	6	12	35	42	3	6	13	15
8A	4	0	2	3	0	0	14	12	0	0	7	5	34	23	x	**59**	41	33	45	41	41	32	35	30	40	34	39	34
8B	2	1	3	6	1	1	11	14	1	1	10	10	30	28	**59**	x	38	39	37	37	33	30	33	31	35	37	33	33
9A	5	1	5	11	0	1	15	14	1	1	42	41	4	5	41	38	x	**72**	47	47	41	32	38	30	47	47	39	34
9B	1	0	4	10	0	0	11	13	0	0	40	43	2	3	33	39	**72**	x	39	41	35	31	33	30	39	46	33	34
10A	4	0	2	3	0	0	25	22	0	0	8	7	3	2	45	37	47	39	x	**60**	42	32	37	30	48	37	39	34
10B	3	4	5	5	0	2	24	26	5	5	9	10	4	5	41	37	47	41	**60**	x	50	37	39	31	48	39	50	44
11A	5	7	6	6	1	5	14	14	31	23	7	7	5	6	41	33	41	35	42	50	x	**53**	39	32	40	36	54	49
11B	9	18	13	14	11	13	23	22	33	40	15	14	11	12	32	30	32	31	32	37	**53**	x	41	41	32	34	49	52
12A	30	39	39	35	40	40	47	45	36	36	39	37	36	35	35	33	38	33	37	39	39	41	x	**65**	38	35	45	46
12B	37	45	41	41	43	45	49	52	42	44	41	44	37	42	30	31	30	30	30	31	32	41	**65**	x	30	31	39	43
13A	4	1	30	27	0	1	14	13	1	1	9	9	4	3	40	35	47	39	48	48	40	32	38	30	x	**56**	39	34
13B	3	3	26	33	0	2	12	15	3	3	18	19	3	6	34	37	47	46	37	39	36	34	35	31	**56**	x	35	35
14A	17	27	12	13	8	12	19	22	17	20	13	15	11	13	39	33	39	33	39	50	54	49	45	39	39	35	x	**69**
14B	20	32	16	17	13	16	24	26	21	26	17	18	14	15	34	33	34	34	34	44	49	52	46	43	34	35	**69**	x

Table 5: Comparison array of the 14 sequences filmed with 2 cameras (A and B)
Results in percent, red: 1st result, orange: 2nd result, green 3rd result, bold: expected best result

The results illustrated in Table 5 indicate that 25 results on 28 are the expected one, and the rest 3 mismatches are the 2nd or the 3rd best result. It is important to remark that the test sequences have been taken in similar lighting conditions; in other cases, the results are expected to be worse. Therefore, it can be concluded that, in good lighting conditions, the histogram characterization can be successfully applied for multi-camera tracking.