Real-Time High Density People Counter Using Morphological Tools

Antonio Albiol, *Member, IEEE*, Inmaculada Mora, and Valery Naranjo

***Abstract*—This paper deals with an application of image sequence analysis. In particular, it addresses the problem of determining the number of people who get into and out of a train carriage when it's crowded, and background and/or illumination changes. The proposed system analyzes image sequences and processes them using an algorithm based on the use of several morphological tools, which are presented in detail in the paper.**

***Index Terms*—Computer vision, morphological operations, people counting, real-time.**

I. Introduction

THE PURPOSE of the work presented here was to provide a tool for the Spanish Railway Company to be able to determine the number of people getting into and out of a train. Currently, this work is done manually by people standing on the platforms surveying a certain train door and counting the number of passengers who get in and out. This is done once a year due to the high personnel cost. Some of the system requirements were as follows.

- *On board system*. All components of the system, namely the cameras and processing unit, should be on the train, in order to be able to use the train as a *traffic probe.* This option has the advantage of enabling the possibility of measuring the traffic in different railway lines by simply changing the probe train to the desired line. Another advantage, is that the camera is always placed at the same relative location to the door (above the door to avoid occlusions). We ruled out the alternative of placing the camera at the station, mainly for the need of an infrastructure and the rigidity of measurements in the network. Besides the variability of the position of the door in the scene should have been solved. In spite of these drawbacks, this option had the advantage of acquiring the images with not such a wide lens, (thus obtaining less geometric distortion) and allowing the same camera to survey several doors. Finally, after some preliminary trials, the railway company imposed that the cameras should be on the train.
- *Minimum train modification*: Another requirement was that minimum intervention should be performed on the carriages. The proposed approach only requires the installation of the cameras (one per door) in the door mechanism box and the cables from the cameras to the computer. The computer determines from the images if the doors are closed or open.
- *Accuracy*: The railway company imposed an accuracy of less than 5% of error. In our system this can be easily verified since video recordings of the sequences are available in order to check the counts. Verification is impossible in the case of today's human counters.
- *No special illumination* or marking should (if possible) be required on the train and/or station.
- *Real time*: The processing time should be smaller than the time needed by the train to make its journey. The storage requirements should be kept at a minimum.
- *Cost*: The desired cost should be approximately that of a PC and some acquisition hardware. No special hardware should be required to process the image sequences. The cameras should also be low cost.
- *Image quality* might be low. In fact, throughout the development phase, we have used VHS recordings of a camera placed above a train door.
- *At each station*, the number of people getting in and out should be provided as output.

A. Previous Work

Before we started our work, the railway company had done a market survey of commercially available systems. Some of the things that they found were as follows:

- *Mechanical counters* such as those used to validate the tickets. Although they are accurate, they involve an obstacle at the carriage door.
- *Light beams*. They are not appropriate to determine the direction of passing and in the case that two people are passing at the same time.
- *Differential weight system*s. Another approach that was suggested was to fit load cells at the carriage suspension that could detect the weight variations of the carriage when someone gets in or out. This was discarded mainly for the severe modifications of the carriage that were required.
- *Sensitive carpet*. This was one of the options more seriously considered. However sensitive carpets are prone to wear, and the determination of the direction of passing and the number of people (one person could tread with one or both feet on the sensitive area) are not straightforward at all.
- *Vision based systems* intended for commercial areas, where the density of people needed to be low. In fact, sample video recordings were supplied to a company which markets these systems and they admitted that their

Manuscript received April 10, 2000; revised April 9, 2001. The Associate Editor for this paper was H. Takahashi.

A. Albiol and V. Naranjo are with the Departamento de Comunicaciones, Universidad Politécnica de Valencia, 46022 Valencia, Spain (e-mail: aalbiol@dcom.upv.es; vnaranjo@dcom.upv.es).

I. Mora is with the Departamento de Tecnología de las Comunicaciones at University Carlos III, 28911 Leganés, Spain (e-mail: inmoji@tsc.uc3m.es).

Publisher Item Identifier S 1524-9050(01)10750-7.

commercial system was unable to cope with the density at the carriage door.

We made a survey of research work elsewhere and the most similar thing that we found was the work by Bartolini [1] for buses. The objective is the same, to count people and to determine the direction of passing, but the environment and the problem are different. Their main problems are sudden illumination changes and the vibrations of the bus while, in our case, the train is perfectly steady and the illumination almost constant when the train is stopped. However they do not seem to have the problem of possible crowds. They also use a high gradient line in order to find out if someone is passing or not.

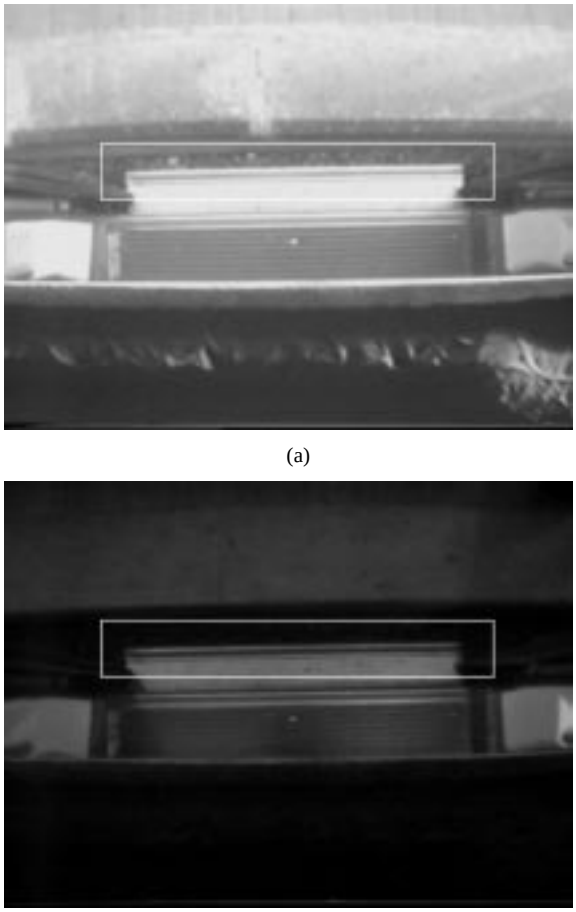
Recently, Terada *et al.* [2] proposed a system for counting people using a stereo camera. Their approach has some similarities with ours—they use zenital cameras, they also built space-time images (see below), and use a pair of lines in order to determine the direction. There are however many differences between their work and ours; they use a stereo camera, and try to obtain a 3-D image in order to make the people/background segmentation while we use a gradient line to do the same thing. To determine the direction they use a pattern-matching approach while we use an optical flow scheme. The density of the people that is shown in their examples is much lower than in our case, and all people appear not in contact with anyone else, and the results that they provide are based in about 40 people passing (counted with no error); they also provide some figures on the error in estimating height of people, which is not applicable in our case.

B. System Overview

Our system is conceptually similar to the human counter: it analyzes images of a door. But rather than doing it from the platform, it does the surveillance from a zenital position above the train door. The camera is placed in the door mechanism box, and the acquired images are monochrome. An example of the camera view is depicted in Fig. 1. However, the trains stop at some underground stations and some surface ones. In these cases the appearance of the images can change. An example of this is shown in the same figure. Moreover, surface stations can produce different background images depending on the hour of the day and weather conditions. The placement of the camera above the door has the significant advantage that no occlusion occurs. Fig. 2 shows three sample frames of people passing with different densities.

Since a person usually takes more than one frame to cross the door, some time memory must be included in the algorithm. In order to reduce the memory storage requirements and to allow noncausal processing, the only thing that we perform each time we acquire a new frame is to store certain lines of the frame onto what we have called *stacks of lines*, which will be described with more detail in the next section. These stacks of lines contain all the required information to count the people. The counting process actually starts after the doors are closed. At that time, the computer disables the image acquisition and processes the mentioned stacks. An overview of the system is depicted in Fig. 3.

The time required to process the stacks is considerably shorter than the time needed to acquire them, and much shorter too than the time needed for the train to reach the next station. In this sense we claim that our process is real-time. Strictly speaking we could say that it is a *delayed-real-time* algorithm.



(a)

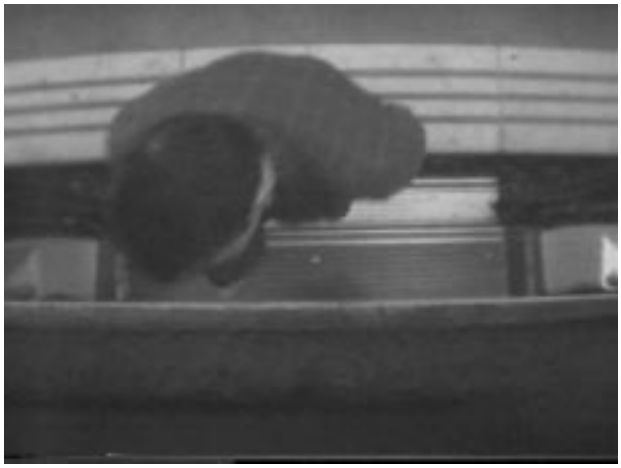
(b)

Fig. 1. Example of full camera views at different stations. (a) Outdoor station. (b) Badly illuminated underground station. The region of interest used by the algorithm is also shown.

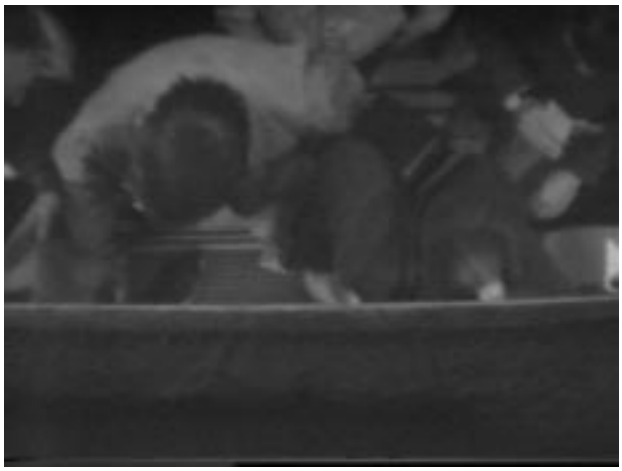
A stack of lines can be considered itself as an image, where the horizontal dimension corresponds to the horizontal axis in the original image, while the vertical one corresponds to time in the original sequence.

Then, our algorithm can be at different states.

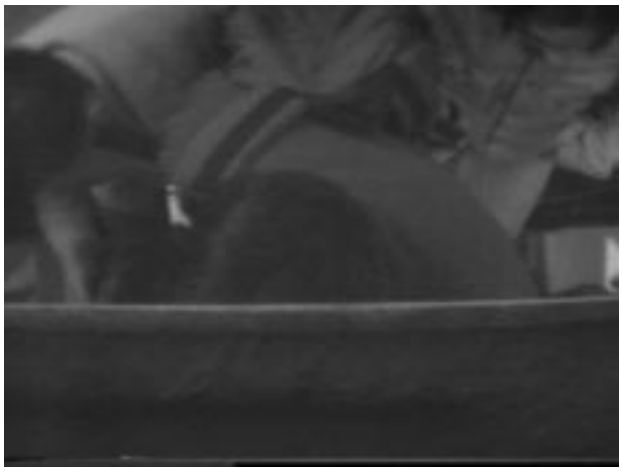
1) When doors are closed the camera lens is covered and the acquired image is completely black. We will assume that this is the initial state, that we have called *Closed-Doors State.* When the train arrives at a station and opens the door, a sudden increase in the brightness happens. During the Closed-Doors-State, every frame is analyzed in order to detect this sudden brightness increase. When this occurs we change into the next state.
2) When the door starts to open, the storage of certain lines of each frame starts. We call this, *acquisition state.* We leave this state when we see a fully dark image for a certain time (approximately 2 s), that means that doors are closed.
3) The next thing to do is to analyze the stacks in order to count the people and write the results onto a file. During



(a)



(b)



(c)

Fig. 2. Example of full camera views with people. (a) Isolated person. (b) and (c) Crowded situations.

this *Counting State* we ignore the video input, since there is enough time to finish the processing before reaching the next station. The full processing of a station takes between 5–10 s on a 233-MHz Pentium. After finishing the counting, we return to the Closed-Doors State to wait for following station.

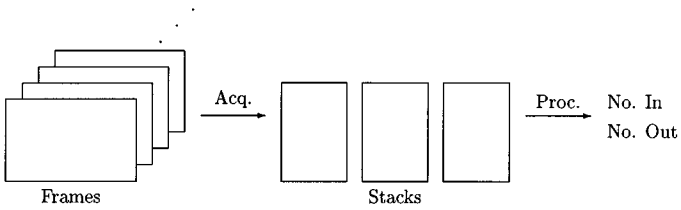


Fig. 3. Overview of the system.

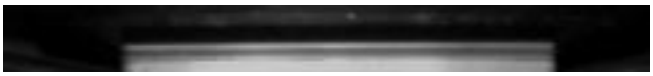


Fig. 4. Decimated ROI. The scale is the same as Fig. 1.

Fig. 5. Position of the black, gradient, and white lines used to build the stacks.

The rest of the paper is organized as follows. In Section II we explain how to build the stacks. Section III gives an overview of the whole processing. In Section IV we explain how to segment people from the background. Section V treats of the separation of the individual persons. Finally we address the determination of direction in Section VI. To conclude the paper we will give some results obtained on a large number of train stops.

II. Image Acquisition

We acquire the images using a standard frame grabber at a rate of 25 images/s. For sparse people this rate can be excessive but for crowds we have found it to be essential.

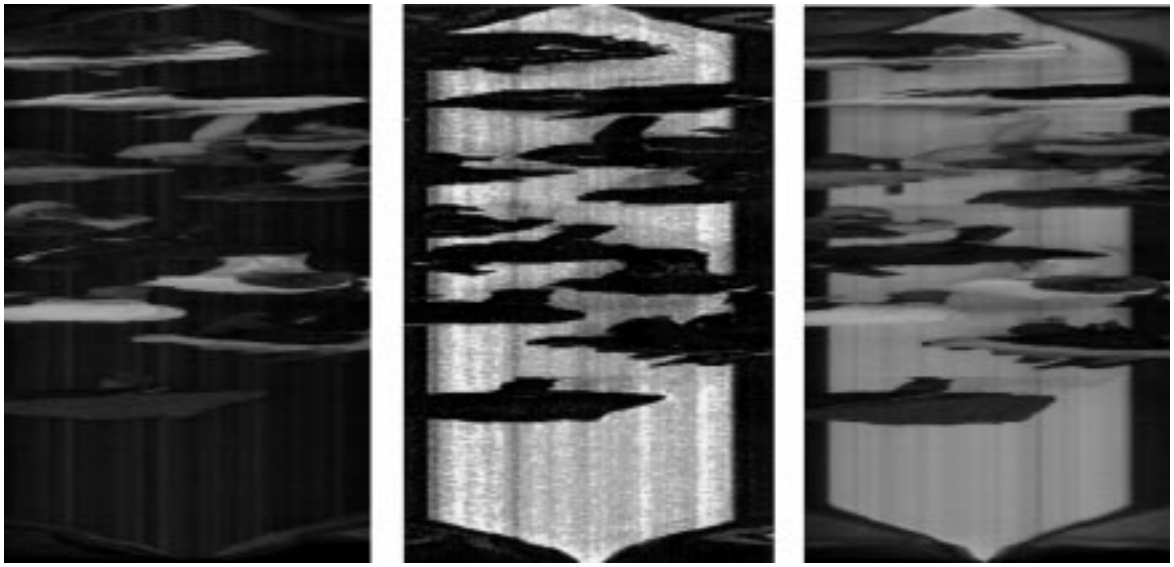
A. Frame Preprocessing

Images are acquired with a size of 756 $\times$ 576 pixels. A region of interest (ROI) is defined at installation time. The ROI contains the interesting portion of the frames in Fig. 1. In other words, the algorithm ignores the image content outside the ROI.

Since a person is *something large,* in order to reduce the computational and memory requirements the first thing that we perform is a low pass filtering and decimation. We have used a horizontal moving average of size 7 decimating by 2 horizontally at the same time.

Vertical decimation by 2 is also carried out. This is done to avoid problems with interlaced video (even and odd lines have a time offset of 1/50th of a second). The filtering is restricted to the area of interest. Fig. 4 shows the result of this phase. Notice that only a small fraction of the image is processed (the size of image in Fig. 4 is 3% of those in Fig. 1).

The next step is to store three lines of this image onto a stack. The lines are shown in Fig. 5. We will call the first line (the topmost one in the figure) the *black line* and it corresponds to



(a) (b) (c)

Fig. 6. These figures correspond to a (relatively) well illuminated underground station. At the top and bottom the characteristic pattern produced by the doors opening and closing can be observed. (a) Black stack. (b) Gradient stack. (c) White stack.

the gap.[1] The intermediate line, we will call it the *gradient line*, should coincide with the upper edge of the train step. The bottom line is located on the step and will be termed *white line*, because it is brighter than the *black* one.

Black and white lines simply contain the gray level of the original image. For the gradient line, if we denote as $f[m, n]$ the pixel intensity at row m and column n, we compute the following thresholded vertical gradient

$$v_g[m, n] = \begin{cases} f[m, n] - f[m+1, n], & f[m, n] > f[m+1, n] \\ 0, & f[m, n] \leq f[m+1, n] \end{cases} \quad (1)$$

and this is the value that we store.

Black and white stacks are primarily used to determine the direction. Gradient line is used to segment people from the background. Since the appearance of the background is different at each station, and people may be of the same gray level as the background, background substraction is not useful in order to segment people from the background. However, we have realized that high gradient values occur rarely within a person, and when they happen they tend to have a very short duration at the precise position of the step. Thus, the use of the stack of gradient values at the step becomes essential in order to differentiate people from background.

Each of these three lines is stacked on separate images where each row corresponds to a frame. At the end we will have three stacks: one for black lines, one for gradient lines, and one for white lines, which we will denote, respectively, as black, gradient, and white stacks. In Fig. 6 we can see an example of stacks. The vertical axis corresponds to time (increasing downwards) while the horizontal axis is the horizontal dimension of the original images.

[1]In surface stations it is not necessary black, but for convenience we will always call it this way.

Since the camera is fixed to the train, the position of the lines in the images does not change from station to station, and is configured (automatically) off-line during the installation.

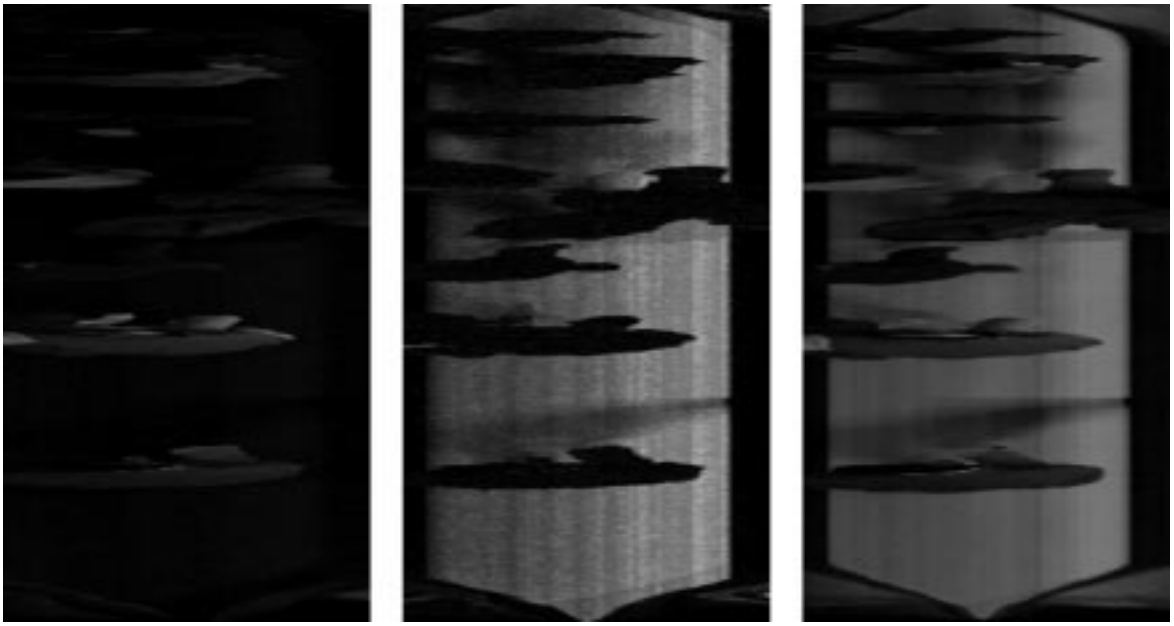
Figs. 7–9 show additional examples of the stacks that appear in practice. Stacks like these will be the input to the actual processing algorithm. We could say that preprocessing and stacking act like an information condenser, passing from a sequence to three images per station.

From the stacks it is quite obvious for a human observer (after a certain amount of training) to see how many people have passed through the door. It suffices to count the number of blobs. At the top and bottom of the stacks a characteristic pattern due to opening and closing of doors can be seen. In the following sections will explain how to do the counting from the stacks automatically.

III. Processing Overview

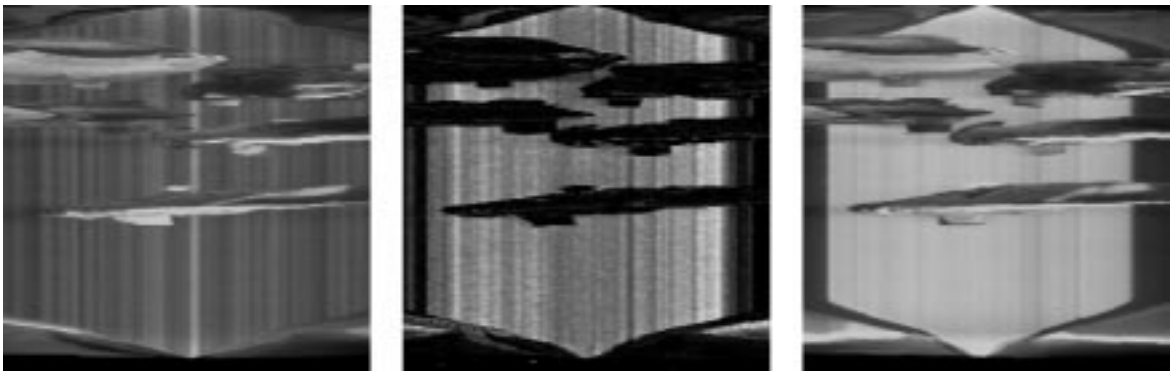
Once we have the stacks of lines, we must analyze them in order to find out how many people they contain. The processing is performed in different stages.

1) *Presence detection*: The aim of this point is to create a binary image (with the same size of the stacks) that indicates when and where the *border line* is being occupied by someone or something passing. The *border line* is the position of the frame that must be crossed in order to make a count. It corresponds to the position of the gradient line.
2) *Segmentation*: The purpose will be now to segment the presence image into individual prints corresponding each one to a single person. It should be noted here that although it is desirable to make a good segmentation to ease the estimation of the direction, this is not the objective of the system; we are interested only in the number of people crossing. Then, we will consider that we have a good segmentation simply if the number of people is



(a) (b) (c)

Fig. 7. These figures correspond to a badly illuminated underground station. (a) Black stack. (b) Gradient stack. (c) White stack.



(a) (b) (c)

Fig. 8. These figures correspond to a surface station. (a) Black stack. (b) Gradient stack. (c) White stack.

well estimated, and the segmentation is accurate enough to allow proper direction determination.

3) *Direction estimation*: After segmentation, we will have an *image* which will be null when nobody is passing and will contain a different label for each portion of the stacks corresponding to a different person. The objective is now to estimate the direction of passing at each of these *labels* of the stacks of lines.

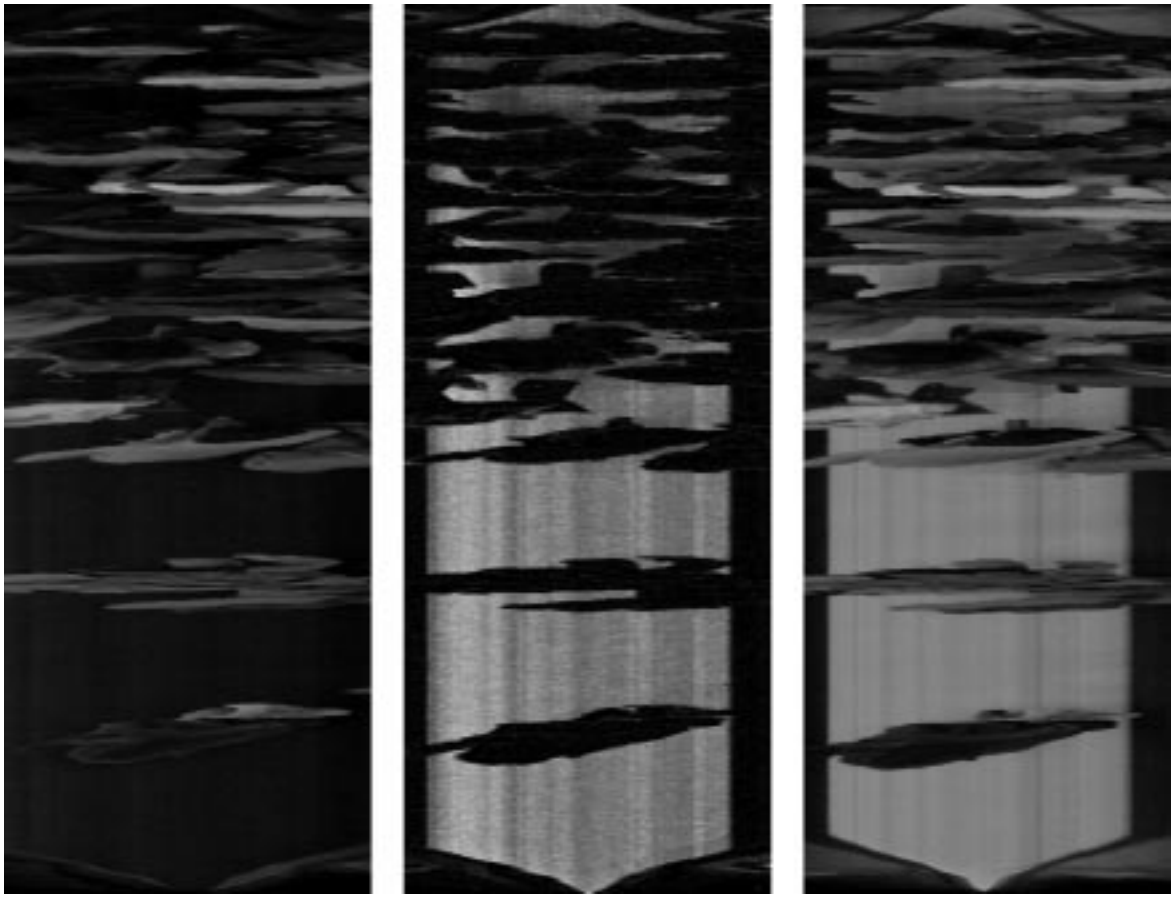
IV. Presence Detection

After the doors are closed, the actual processing starts. We have as input three stacks of lines. The first thing that we do is a someone/nobody segmentation of the stacks. In other words, we obtain a binary image of the same size of the stack, indicating the presence of a person at the border line.

We can distinguish two zones in the stack images. A central corridor and two side ones (see Figs. 6–9). The central corridor corresponds to the train step, and people must necessarily pass through it in order to get in or out. The columns corresponding to the central corridor are set up (automatically) at the installation stage. Since the camera position is fixed, the location of these columns does not change at different stations.

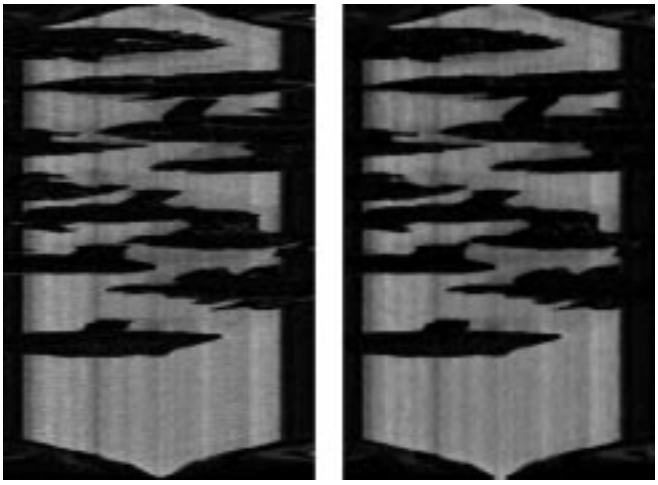
We have made the following observations on the stacks of any station at the central corridor.

- The value of the gradient stack is normally small when a person is passing, and high when there is no one. This is consistent with the fact that a person usually corresponds to smooth gray level variations in the vertical direction. However, high gradient values may occur when a person has an object with high gradient. Nevertheless, these high gradient values have a short duration when they correspond with a person unless the person stops exactly at the border line position.
- At the black line (gap train platform), when no one is passing, the gray level does not change significantly. The gray level when someone is passing depends on the gray level of the person passing (quite variable).
- At the white line (train step), the presence of shadows can produce large variations on the gray level when no one



(a) (b) (c)

Fig. 9. These figures correspond to a high people density. (a) Black stack. (b) Gradient stack. (c) White stack.



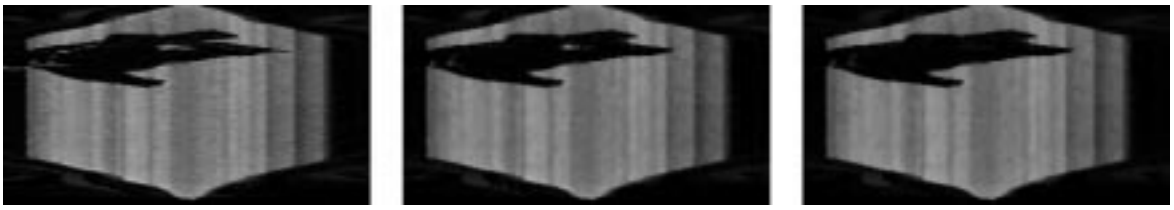
(a) (b)

Fig. 10. Prefiltering. (a) Gradient stack. (b) Open–close of (a) using a vertical SE of size 3.

is passing. When someone is passing, as in the case of the black line, the gray level also depends on the person passing and is in general quite variable.

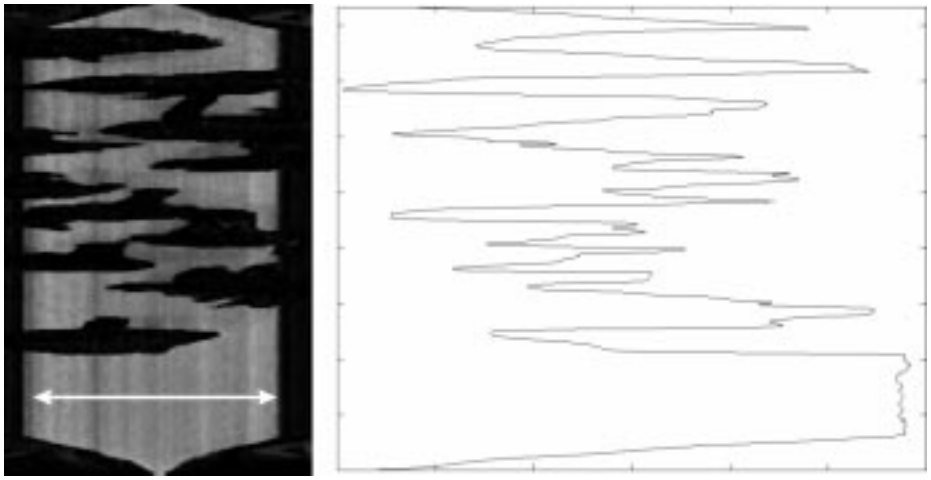
At the sides, it can be seen that the gradient is low in both cases, when someone is passing and when not. This makes the gradient unsuitable for the determination of presence at the sides. The only information that can be used at the sides is when someone passing is wearing clothes with a gray level different to the background.

If the stacks of different stations are compared, it is easily noticeable from the given examples that different gray levels and gradients may occur. Also, the values of the black and gradient stacks at the background can have horizontal variations specially at surface stations.



(a) (b) (c)

Fig. 11. Prefiltering. (a) Gradient stack. (b) Open–close of (a) using a vertical SE of size 3. (c) Open by reconstruction of (b) with a vertical SE of size 7.



(a) (b)

Fig. 12. Estimation of nobody lines. (a) Gradient stack. (b) Horizontal projection at the central corridor.

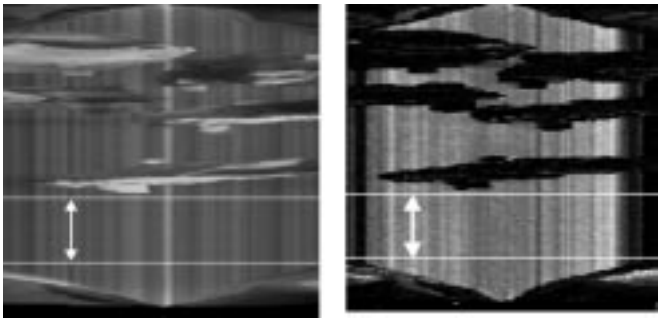
The process for presence detection can be summarized as:

1) prefiltering;
2) determination of nobody lines;
3) background estimation;
4) segmentation people/background;
5) door removal;
6) post filtering.

We are going to see every point in detail.

A. Prefiltering

The gradient stack is particularly useful since, unlike the other two stacks, in the central portion, bright normally means nobody and dark mean someone. The purpose of the prefiltering is to eliminate transients with too short duration. In particular, we have performed an opening [3] with a vertical SE of size 3 followed by a closing with the same SE on the gradient stack. The first opening removes all bright gradient values with a duration shorter than three frames. High gradient values are normally due to the train step, but occasionally and during a few frames, a person may contain a high gradient at the train step position. So, we remove all high gradient occurrences of very short duration. When a person is passing it normally takes more than three frames to cross the line. That means that gradient will remain low for more than three lines at the same horizontal position. Short dark transients on the gradient stack will normally be due to noise. This closing, can sometimes reduce a little the width of a person, but this is not a problem since the *main part* of the person will be preserved (the main part of the person takes more than 3/25th of a second to pass). Fig. 10 shows the effect of the prefiltering.



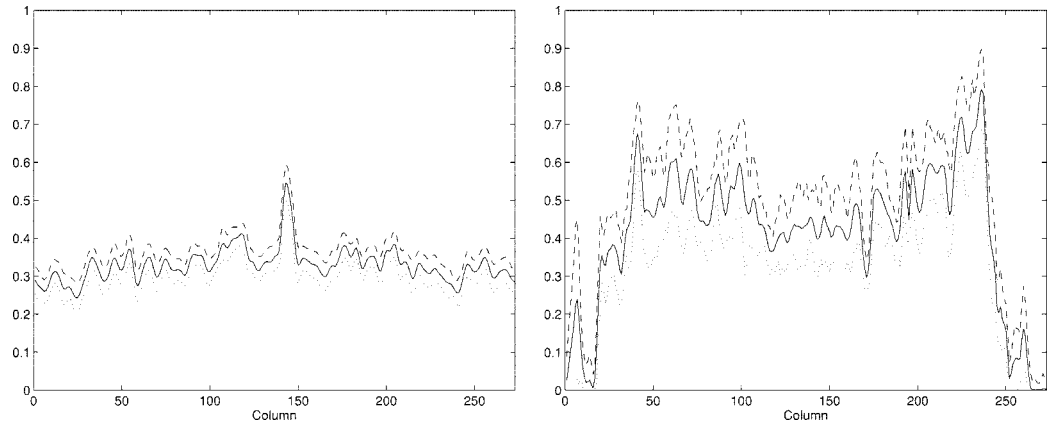
(a) (b)

Fig. 13. Portions of stack used to estimate the background. (a) Black stack. (b) Gradient stack.

Finally, an opening by reconstruction [4], [5] with a vertical SE of size 7 is performed. This is because, sometimes (not in Fig. 10), we have observed that a person passes very slowly and with high gradient values. An example of this is shown in Fig. 11. The opening by reconstruction only leaves those bright areas that at least in one point have a vertical size of 7. This is consistent with the way people walk one after another. We are imposing that at least in one horizontal position the time gap between two people passing one after the other must be longer than 7/25 of a second. It is important not to leave bright values of gradient inside a person print, since the segmentation algorithm might split it.

B. Determination of Nobody Lines

After prefiltering, we determine the rows of the stacks where we are sure that nobody is present. The detection is based on



(a) (b)

Fig. 14. Estimated mean row stacks from Fig. 13, confidence intervals are also shown. (a) Black stacks. (b) Gradient stacks.

finding the horizontal projection at the central corridor of the gradient image. Fig. 12 shows a filtered gradient stack, the central corridor and the horizontal projection. From the projection, we simply consider the longest interval with highest projection. This gives us the rows where we are sure that there is nobody, and we will use this information to estimate the background.

This is done in order to estimate how the background of the current station looks like. Normally, when doors open, people tend to pass immediately. So, the time when no one is passing happens after some people have passed. It is interesting to note that background is estimated *after* people have passed (noncausality). The use of stacks allows this noncausal processing by storing much less information than the whole image sequence.

C. Background Estimation

We estimate now the particular appearance of the background at the current station. From the margin of rows where we know that there is nobody, we compute the mean of each column in order to have a *black mean row* and a *white mean row.* Notice that the mean changes from column to column. We also compute the standard deviation σ column by column.

Fig. 13 shows the portion of the black and gradient stack in Fig. 8 used to estimate the background. In Fig. 14 we can see the estimated mean rows for black and gradient stacks as well as a confidence interval of $\pm 5\sigma$.

D. Segmentation People/Background

At this point, we have estimated the background of the station in the black and gradient stack. At the central corridor, pixels with a gradient value inside the confidence interval will be termed as background and those outside will be classified as people.

At the sides, the gradient is low both with and without people. This leaves only the black stack to classify side pixels. If a side pixel has gray level outside the confidence interval it will be classified as people. If it is within the confidence interval we will leave it unclassified because it can be background, or a person wearing the same gray level as the background. So for pixels at the sides we can say that either it is a person or, "we don't know."

Finally, pixels labeled as "we don't know" (placed at both sides) are removed according to the following rule.

- If a horizontal segment of unlabeled pixels is surrounded by the left and right with pixels of the same label, the segment is assigned this label.
- Otherwise, the label corresponding to the neighbor pixel closest to the center is assigned.

This rule tries to produce a horizontal continuity in the labeling. It should be noticed that slight error in the labeling of "we don't know" pixels (remember, at the sides) normally has no impact on the counting, since *people pass by the center* of the door.

E. Door Removal

Doors opening exhibits low gradient (see Figs. 6–9) and is classified as people by the scheme of Section IV-D. We are going to see how to remove the doors. It is not difficult to detect the row where doors open and close. At the beginning (top of the stack), when doors are closed, the gradient is low. When doors start to open, a sudden increase in the gradient value happens at the center. This gives the indication of which is the opening row. Since doors open at the same approximate speed, they always create a certain pattern. This pattern can be easily *erased.* The same happens at the doors closing.

F. Post-Filtering

Sometimes, due to noise, a person print may contain some narrow holes, and in portions clearly corresponding to nobody small spots of the people label happen. This can be easily removed by alternate sequential filters with horizontal structuring element. Alternate sequential filters are morphological filters [6] that consist in a sequence of openings and closing with structuring elements of increasing size.

Figs. 15 and 16 show the results of the presence detector described above when applied to the sample stacks of Figs. 6–9. It is interesting to observe the different shapes of the prints.

V. People Isolation

In the preceding section we have seen how to obtain a binary mask of presence. In order to count the people, we must segment

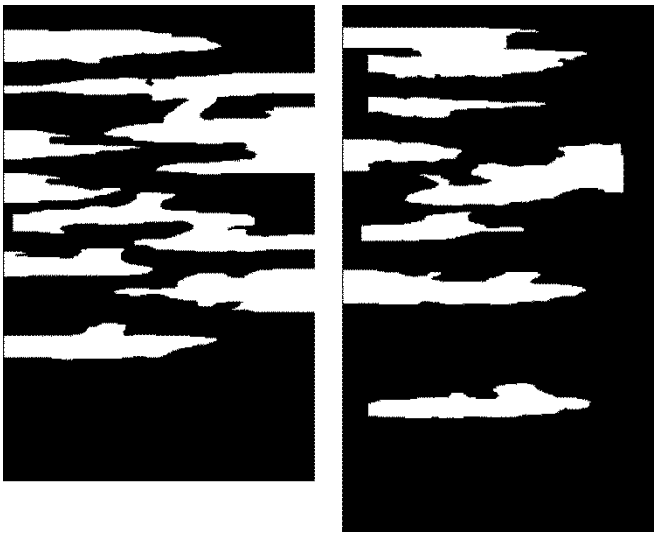


Fig. 15. Presence detection result of stacks in Figs. 6 and 7.

Fig. 16. Presence detection result of stacks in Figs. 8 and 9.

this presence mask into individual prints due to a single person. In order to explain the separation techniques that we have used, we have prepared a synthetic image to show how the different separation techniques work. That synthetic image is shown in Fig. 17 and contains the most important kind of problems encountered in our analysis. We would like to recall again that in this figure the vertical axis is time and the horizontal one is the horizontal dimension of the original image. From top to bottom we can see the following prints:

1) an isolated person, passing quickly. This can be noticed by the short vertical size of the print;
2) two people passing side-by-side;

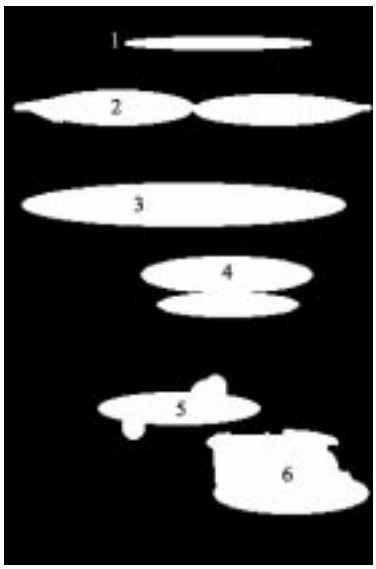


Fig. 17. Synthetic image used to explain the different separation techniques.

3) a single person passing exactly under the camera. The camera height is small, so wide-angle lenses are used. In this case, the print can be as wide as that of two people if the person happens to pass exactly under the camera;
4) two people passing one immediately after the other, leaving no frame (row of the stack) of gap between them;
5) one person. Print with branches. This is a very common situation. Legs, arms, bags, etc. appear normally as narrow branches;
6) one person passing slowly. Irregular print. One person passing slowly takes many frames to completely cross the door. This causes a large vertical size of the print. The shape of real prints has irregularities as those shown.

The separation process is divided into the following steps:

- side contacts detection/separation;
- markers extraction;
- segmentation.

In the following subsections we will see these steps in more depth.

A. Side Contacts Detection/Separation

A side contact happens when two people cross the door at the same instant. One could think that this should cause a print typically twice as wide than the one due to a single person. However, this is not the case due to the low height of the camera which causes geometric distortion. We have found the following two conditions which are met when two people have a side contact.

- The print has a width above a certain threshold.
- The vertical gradient of the width function[2] within a print contains one line of high positive values and another one of negative ones. This happens because when two people meet, a sudden increase in the width of the print occurs.

[2]The width function of a binary image assigns a zero value to pixels which are null in the original image, and for pixels which are one in the original image assigns a gray level value equal to the width of the horizontal segment to which the pixel belongs. It can be efficiently computed using recursive algorithms similar to those described in [7].

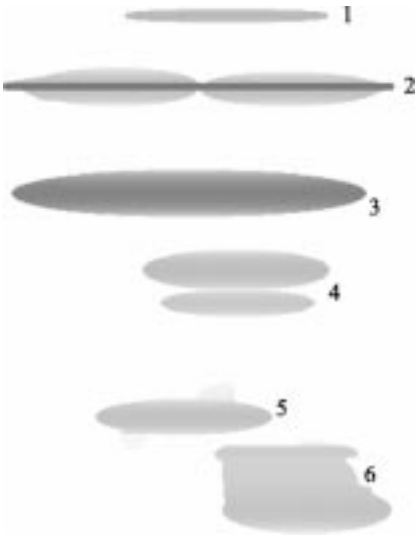


Fig. 18. Negative of width function of Fig. 17.

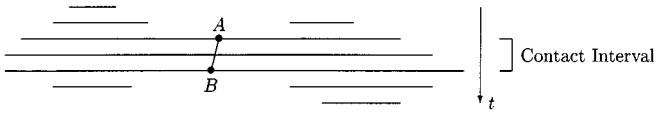


Fig. 19. Separation of side contacts.

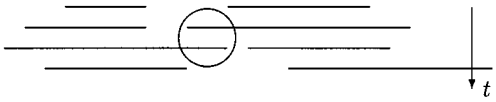


Fig. 20. Need of horizontal shrinking.

When the contact ends, we will have a sudden decrease in the width. This is seen in the synthetic example width function (Fig. 18) as a dark strip surrounded vertically by high vertical gradient. Compare print 2 with print 3.

The duration of the contact corresponds to the interval between the two rows with high vertical gradient of the width function.

Once a contact is detected, we proceed to separate the individual prints. To do so, we find points A and B (Fig. 19) as the centers of the background segments in the row previous and following the contact. We join them by a straight line and delete the pixel closest to this line from each row of the contact. This gives horizontally separated segments for the print of each person.

However, quite often, situations like the one depicted in Fig. 20 happen. We can see at any row two separated horizontal segments. In order to group the segments on each side into one person, vertical (time) connectivity must be applied. This would join (at the place indicated with the circle) the segments of the right- and left-hand sides into one print. To solve this, we shrink the horizontal segments of the prints, so that vertical connectivity of the left- and right-hand sides is preserved, but right- and left-hand sides are kept separated. The result of the horizontal shrinking is shown in Fig. 21.

B. Longitudinal Separation

After side contact separation, only longitudinal separation remains. The information that we use to perform this separation

Fig. 21. Width function of the horizontal shrinking after side contact separation of Fig. 21.

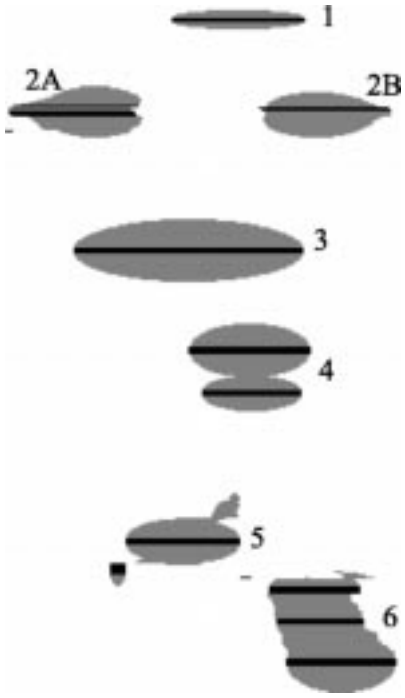


Fig. 22. Prints after H-shrinking and maxima of the width function.

is that the people prints have an *approximately* convex shape. That would suggest the use of the maxima of the width function as markers. If we take a look at the binary image after the horizontal shrinking, and search the maxima of the width function we can make the following observations (Fig. 22).

Fig. 23. Extinction function of the width function in Fig. 18.

- Due to shrinking, narrow horizontal segments become unconnected from the main portion of the print (see prints 5 and 6)
- More than one maximum occurs sometimes at one single print. This is due to irregularities in the print shape.

In order to separate the prints we have computed the contrast extinction value of each maximum of the width function.[3] This is shown in Fig. 23. The following considerations can be made.

- If a print only contains a single regional maxima of the width function, then, the extinction value of the maximum is the same as its gray level.
- If one print contains several maxima of the width function, then, the extinction value of the brightest maximum is the same as its gray level.
- If one print contains several maxima of the width function, then, the extinction value of the rest of maxima (all except the highest one) is the difference of widths $W2$–$W3$ shown in Fig. 24.

All these considerations, give the following criteria that a maximum of the width function has to fulfill in order to be considered a marker.

- Its associated width must be above a certain threshold (minimum width of a person).
- Its associated extinction value must be above a certain percentage (we have used 60%) of its width. This way, we are imposing a certain narrowance of the print between two maxima of the width.

[3]The contrast extinction value of a maximum is defined as the minimum descent necessary to reach a higher maximum. An indepth treatment of extinction values and efficient computation techniques can be found at [8].

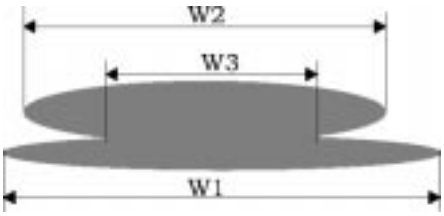


Fig. 24. Relation of extinction values of maxima of width function to widths. See the text for more details.

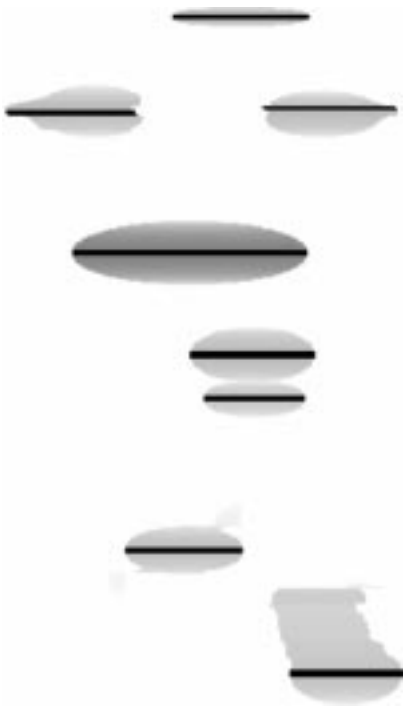


Fig. 25. Markers (in black) and image to which we will apply the watershed.

This provides a correct set of markers for the shrunk image. Applying the watershed segmentation [9], [10] to the negative of the width function of the shrunk image (Fig. 25), we obtain a segmentation of the prints with two markers *at the narrowest horizontal segment located between both markers*. To segment the full width image (not shrunk), labels are simply extended to the sides of the horizontal segments. The result can be seen in Fig. 26.

At this point we will have the presence image segmented into portions corresponding to individual persons. It only remains to estimate if they are getting in or out.

VI. Direction Determination

We have used an algorithm based on optical flow [11] to estimate the direction of crossing the door. Let us denote as $f(x, y, t)$ the intensity of an image as a function of space (x, y) and time t. The optical flow equation states

$$\frac{\partial f}{\partial t} + v_x \frac{\partial f}{\partial x} + v_y \frac{\partial f}{\partial y} = 0 \tag{2}$$

where v_x and v_y are, respectively, the x and y components of the speed vector. In conventional optical flow motion estimation additional smoothness constraints must be imposed in order to

(a) (b)

Fig. 26. (a) Watershed segmentation of Fig. 25. (b) After extension of the labels to the sides.

estimate v_x and v_y, since there are two unknowns and one only equation. In our application, much in the same way as [1], we will assume that the horizontal component v_x of the motion is null. Then, we have

$$v_y = -\frac{\frac{\partial f}{\partial t}}{\frac{\partial f}{\partial y}}. \tag{3}$$

We estimate the gradients from the stacks in the following way.

- The vertical gradient is estimated as the pointwise signed difference of the black and white stacks of lines. Recall that the same pixel coordinates in these two stacks correspond to points belonging to the same frame (same row in stack) and column. Then, computing this difference is equivalent to computing a vertical gradient on the original frames;
- The time gradient is computed as the vertical gradient of the black stack;
- A vertical (time) averaging of size five of the stacks is performed prior to the above computations.

The speed obtained from the optical flow equation gives one value per pixel of the stack. For each label of the segmentation of Section V we compute the average value of speed. Depending on the sign of the result we decide that the person is getting in or out. Notice how all the information corresponding to each person is used to estimate the direction. However, averaging of the values obtained from equation (3) may give problems. If both, the spatial and time gradient in that equation become very small, the speed that we obtain is quite random. We have weighted the speed obtained for each pixel with a *measure of confidence* in that value. The measure of confidence that we have used is the magnitude of the spatial gradient. This is consistent with the fact that for an object with a null vertical gradient it is not possible to observe its vertical motion What we actually average is

$$V_y = -\frac{\frac{\partial f}{\partial t}}{\frac{\partial f}{\partial y}} \left|\frac{\partial f}{\partial y}\right| = -\frac{\partial f}{\partial t}\mathrm{sign}\left(\frac{\partial f}{\partial y}\right). \tag{4}$$

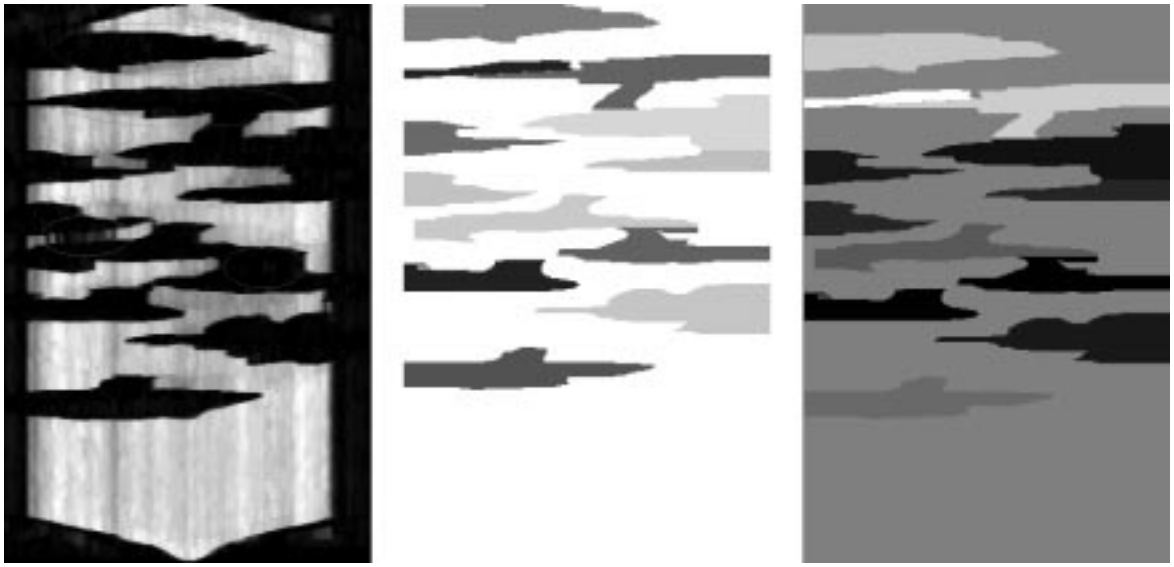
This way, the points of the stack corresponding to a higher vertical gradient in the original frame contribute more to determine the direction. Results of this method can be seen at Figs. 27–30(c). The optical flow result shows the background with medium gray. Brighter than background means *getting off* and darker means *getting in*. The more contrasted with the background means a greater speed.

VII. Results

First, we are showing the results on examples corresponding to the stacks of Figs. 6–9. Figs. 27–30 show the gradient stack, as well as the segmentation described above and the estimation of the direction. Each gray level represents a different person at the segmentation. White corresponds to background.

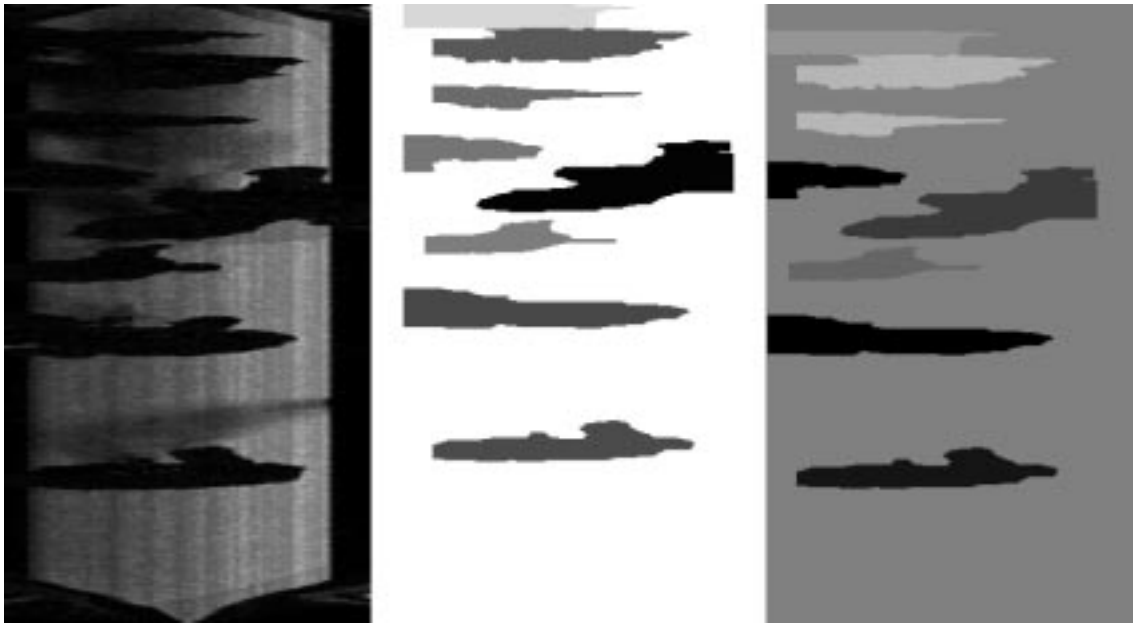
The number of people counted is at the corresponding captions. Looking at the video tape and counting manually (in slow motion, because it is virtually impossible to count in some cases from live video) we can check that the result of Figs. 27–29 are correct.

In Fig. 30 there are two *missing* people which are not counted; in fact several people are mixed into one because there is little or no gap at all between them. Sometimes, we have found the opposite situation: one person is counted as two. This situation happens sometimes when a person doubts at the moment of getting in. Then the convexity of the print fails, and we get two markers for one person.



(a) (b) (c)

Fig. 27. This figure corresponds to the original data in Fig. 6, the result is 3 get off and 9 get in. (a) Gradient stack. (b) Segmentation result. (c) Average of confidence weighted optical flow.



(a) (b) (c)

Fig. 28. This figure corresponds to the original data in Fig. 7, The result is 3 get off and 5 get in. (a) Gradient stack. (b) Segmentation result. (c) Average of confidence weighted optical flow.

We have performed a large number of tests (149 train stops) corresponding to images taken at different times of different days in a railway line near Madrid. The test sequences include both indoor and outdoor stations. No parameter tuning has been made for the whole set of test sequences. The intensity of people passing varies from no one passing at certain stations, to an average intensity of more than one person per second. The results are shown in Table I.

VIII. Conclusions

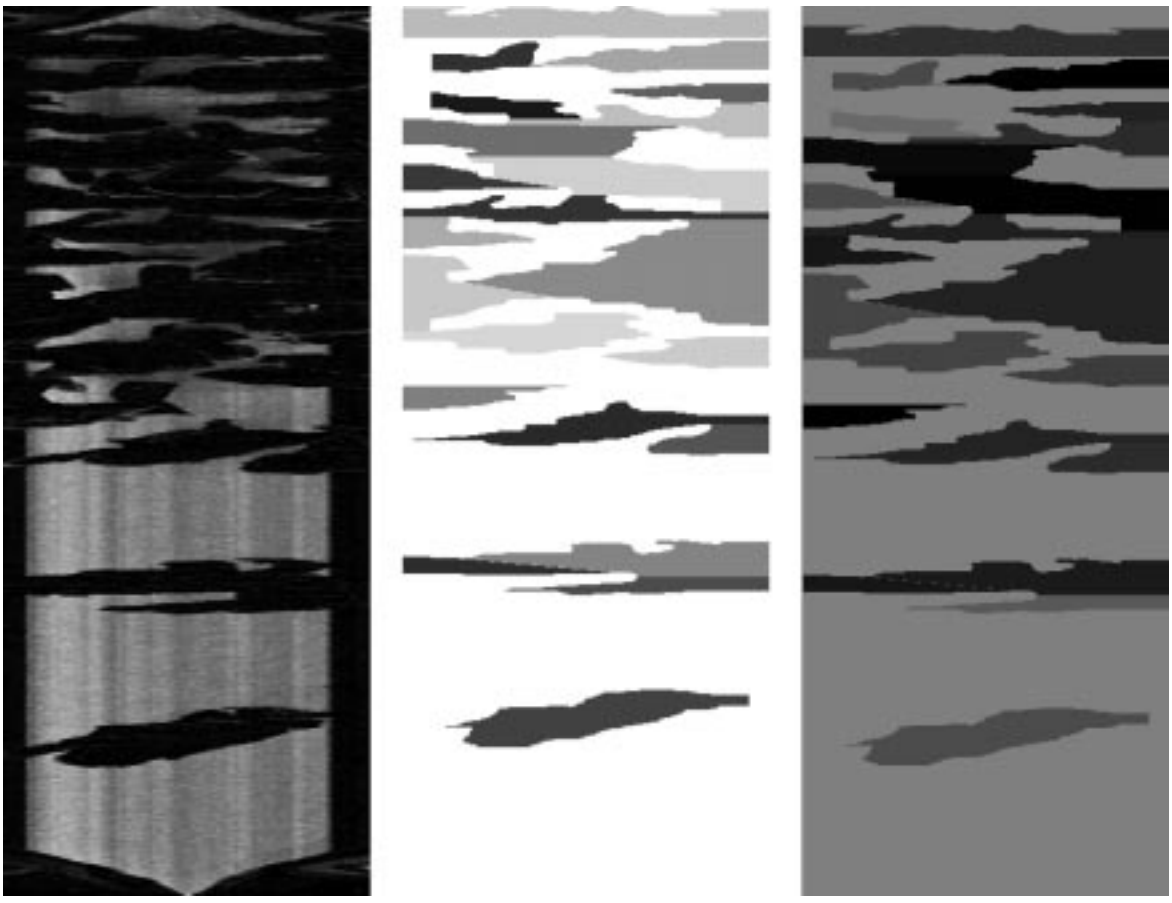
In this paper we have presented in detail a vision solution to the problem of determining the number of people who get in and out of a train carriage. We have addressed the most significant aspects of the whole system, from image acquisition to the processing algorithm.

The proposed method has been conceived for the specific application. However, some of the ideas presented can be applied in other fields. In particular:

- an image analysis system developed that works in an uncontrolled environment, and that solves an important problem in the planning of transportation systems;
- proposed solution that can deal with the high densities of people usually found at suburban railway stops;
- system has low computational requirements, and can work in real time;

(a) (b) (c)

Fig. 29. This figure corresponds to the original data in Fig. 8, the result is 5 get off and 0 get in. (a) Gradient stack. (b) segmentation result. (c) Average of confidence weighted optical flow.



(a) (b) (c)

Fig. 30. This figure corresponds to the original data in Fig. 9, the result is 0 get off and 26 get in. (a) Gradient stack. (b) Segmentation result. (c) Average of confidence weighted optical flow.

TABLE I
SUMMARY OF RESULTS

	No. In	No. Out
Real	321	385
Counted	318	379
% Error	1%	1.6%

- how the use of stacks of lines can be used in order to reduce the amount of required storage to process sequences of images;
- how morphological operators can be used on data with mixed dimensionality, in our case time and space;
- provided a physical meaning to the effect produced by global operators such as opening by reconstruction in time-space images;
- proposed a model for the print of a person passing in time-space images.
- an algorithm developed to separate touching convex blobs for the case of time-space images. The proposed method takes into account that the model of the print shape is only

approximate (legs, arms, luggage, clothes, etc.) This has been done by appropriately defining a criterium to select markers of people.

- Finally, we have shown how the optical flow algorithm can be modified to take into account that the information from pixels with higher spatial gradient is more reliable than that with a small one.

REFERENCES

[1] F. Bartolini, V. Cappellini, and A. Mecocci, "Counting people getting in and out of a bus by real-time image-sequence processing," *Image Vis. Comput.*, vol. 12, no. 1, pp. 36–41, Jan. 1994.

[2] K. Terada, D. Yoshida, S. Oe, and J. Yamaguchi, "A method of counting the passing people by using stereo images," in *Proc. IEEE Int. Conf. Image Processing*, Kobe, Japan, 1999, pp. 338–342.

[3] J. Serra, *Image Analysis and Mathematical Morphology*. London, U.K.: Academic, 1982.

[4] P. Salembier and J. Serra, "Flat zones filtering, connected operators and filters by reconstruction," *IEEE Trans. Image Processing*, vol. 3, pp. 1153–1160, Aug. 1995.

[5] L. Vincent, "Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms," *IEEE Trans. Image Processing*, vol. 2, pp. 176–201, Apr. 1993.

[6] J. Serra, *Image Analysis and Mathematical Morphology, Vol. II: Theoretical Advances*. London, U.K.: Academic, 1988.

[7] L. Vincent, *Mathematical Morphology in Image Processing, Morphological Algorithms*, New York: Marcel Dekker, 1993, ch. 8, pp. 255–288.

[8] C. Vachier, "Extraction de caracteristiques, segmentation d'image et morphologie mathematique," Ph.D. dissertation, Ecole Nationale Supériore des Mines, Paris, France, Dec. 1995.

[9] S. Beucher and F. Meyer, "Morphological segmentation," *J. Vis. Comput. Image Representation*, vol. 1, no. 1, pp. 21–46, 1990.

[10] ——, *Mathematical Morphology in Image Processing, The morphological Approach to Segmentation: The Watershed Transformation*. New York: Marcel Dekker, 1993, ch. 12, pp. 433–481.

[11] A. Murat Tekalp, "Digital video processing," in *Optical Flow Methods*. Englewood Cliffs, NJ: Prentice-Hall, 1995, ch. 5, pp. 72–116.

Antonio Albiol (S'87–M'88) received the engineering degree at the Politechnical University of Madrid, Madrid, Spain, and the Ph.D. degree from the Polytechnical University of Valencia, Valencia, Spain, in 1987 and 1993, respectively.

He is currently a Professor in the Communications Department at the Polytechnical University of Valencia, Valencia, Spain. He teaches courses on digital signal processing and its applications. His main research interests are digital image processing, signal processing, and mathematical morphology.

Inmaculada Mora received the engineering degree, in 1998, from the Polytechnical University of Valencia, Valencia, Spain. She is currently working toward the Ph.D. degree at the Departamento de las Tecnologias de las Comunicaciones at Carlos III University, Leganes-Madrid, Spain.

From 1999 to 2000, she was a research engineer at the Departamento de las Tecnologias de las Comunicaciones at Carlos III University, Leganes-Madrid, Spain. Her current research interests include image vision applications and neural networks.

Valery Naranjo received the engineering degree in 1995 from the Communications Department of the Polytechnical University of Valencia, Valencia, Spain. She is currently working toward the Ph.D. degree at the same university.

Currently, she is an associate professor in the Communications Department of the Polytechnical University of Valencia, Valencia, Spain. Her main research interests include digital image processing, video restoration, and computer vision.