Department of Systems and Informatics

Technical Report

A Vision System for Estimating People Flow

Alberto Del Bimbo, Paolo Nesi

DSI-RT 15/93

Faculty of Engineering, University of Florence
Via S. Marta 3, 50139 Firenze, Italy
nesi@ingfi1.cineca.it

A Vision System for Estimating People Flow

A. Del Bimbo, P. Nesi
Department of Systems and Informatics, University of Florence, Italy.
Tel.: +39-55-4796265, Fax.: +39-55-4796363, NESI @ INGFI1.ING.UNIFI.IT

Abstract

Counting the number of people crossing a public area can be very useful for properly scheduling the frequency of a service. Mechanical and photosensitive systems, such as rotating tripod gates, short iron doors, weight-sensitive boards, and photoelectric cells, have often been used for such estimates. Since these methods are not efficient in critical conditions, vision-based approaches have been provided. Many of them identify moving objects through a segmentation process. Once the objects are identified, they are tracked in the sequence of images and counted. These approaches have some drawbacks when they are used in critical conditions such as for counting the people getting on and off a public bus. In this paper, a new technique for counting passing people which is based on motion estimation and spatio-temporal interpretation of the estimated motion is proposed, with its implementation on prototype DSP-based architecture.

Index terms: counting system, people's flow, feature flow field, spatio-temporal optical flow interpretation.

1 Introduction

Estimating the number of people getting on and off a given place is of interest in many contexts, among these: public services as buses and metros, to properly scheduling the frequence of service; museums, to limit the number of people in certain areas; shops and supermarkets, to control the consumers' interest with respect certain products, etc..

Technical solutions have often used mechanical systems, such as rotating tripod gates, and short iron doors. These methods are not recommended when the flow of people high, as usually occurs at the exit of public places or when people try to catch a bus, because such methods could create a slowing down in the flow, and this in turn could cause accidents. Alternative solutions are based on photoelectric cells or weight-sensitive boards placed on the ground, but these solutions are not robust enough when the people flow is not constant in direction and intensity – e.g., when people arrive under the sensor, stay without motion for some time instants, and then move in the same or opposite direction afterward.

A different approach for counting moving people for solving the above-mentioned problems is the use of a microprocessor-based system connected to a camera. Several experiences have been reported in literature, in which vision systems for counting moving objects are used. Many of these identify moving objects through a segmentation process. Once the objects are identified, they are tracked in the sequence of images, and counted according to their behavior.

The segmentation process may be based on either (i) traditional algorithms for image segmentation, or (ii) the estimation of an approximated projection on the image plane of 3D object motion (usually called "image flow" or "optical flow" [1], [2]).
The main drawback to the first approach is its high computational complexity due to the fact that edges must be extracted [3]. Since, the edges which belong to non-moving objects are also detected, it is even more difficult to select the moving objects. Several different techniques using the second approach have been proposed: cumulative differences among frames have been used in [4], and [5] to extract the moving objects' shape; the clustering of image brightness features having the same optical flow field has been proposed in [6], [7]. Bayesian and entropy formulations for moving object segmentation starting from the image sequences [8], or from the motion fields [2], have also been reported in literature.

Most of these motion-based segmentation approaches are not completely satisfactory. Typically, when different objects having the same velocity are very close or connected to each other, they are recognized as a single object. In addition, these are strongly sensitive to noise.

As to tracking segmented objects, one problem is related to the fact that shapes representing bodies of moving people cannot be regarded as rigid objects, and therefore object tracking cannot be simply based on shape matching.

As an alternative approach, spatio-temporal reasoning has often been applied to analyse pedestrian and vehicle flows. Several researchers have studied the spatio-temporal surfaces of selected objects under motion [9], [10], [11]; works based on the Epipolar Plane Image (EPI) [12], [13], [14], [15], [16]. have been also presented.

Limitations on the use of the above techniques partially depend on the constraining assumptions that are to be made. Most of the techniques require that the moving objects be completely visible (i.e., their size is much smaller than the view area of the acquisition system), and that the moving objects never stop (in order to maintain the conditions for motion-based tracking). In an uncontrolled outdoor environment both these two assumptions are usually false. A further limitation, regarding most of the above-mentioned approaches is that they are computationally too complex to be implemented in a low-cost automatic counting system.

In this paper, we present the principles and experimental proofs of a system for counting moving people in order to estimate input/output flow on city buses. It is a substantial improvement on the

prototypical version described in [17]. The system proposed is based on optical flow field estimation [1], [2], and follows a new method for spatio-temporal analysis of optical flow fields, so that the previously presented problems concerning shape visibility, and behavior of people flow are avoided. The system was developed in a joint project involving industries in the Florence area and was partially sponsored by CESVIT (Agency for Technology Development of Industry in the Florence Area). A prototype of a DSP-based embedded system was developed.

The paper is organized as follows. In Sec.2, the system's working conditions and general architecture are discussed. In Sec.3, the fundamentals of optical flow estimation with some techniques for motion estimation are briefly reviewed. In the same section the algorithm for motion estimation employed in our system is described, and some experimental results are given. In Sec.4, the features of spatio-temporal domain are explained, and the algorithm for counting moving people who pass under the image acquisition system is proposed. In Sec.5, a brief description of system implementation is given. Conclusions are drawn in Section 6.

2 System Architecture and Working Environment

In this section, the environmental conditions necessary to proper functioning of the system will be discussed. Moreover, a brief overview of the system architecture will be given.

2.1 System requirements

A vision system for counting people getting on/off a bus must satisfy the following requirements:

1. low-cost. This guarantees wide applicability. This requirement is mainly satisfied by limiting the computational complexity of the algorithm used for counting people in real-time;
2. robustness with respect to critical conditions in the acquisition system, such as noise in the image acquisition system, and out-of-focus moving regions;
3. robustness with respect to the optical flow estimation problems discussed in Sec.2;
4. robustness with respect to changes in environmental conditions (i.e., illumination, weather, etc.). The system must work both under sunlight and the artificial light generated inside the bus after sunset;
5. robustness with respect to the people size, because the moving people are not completely included in the image acquisition view area and cannot be considered as rigid objects;
6. robustness with respect to people's behavior.

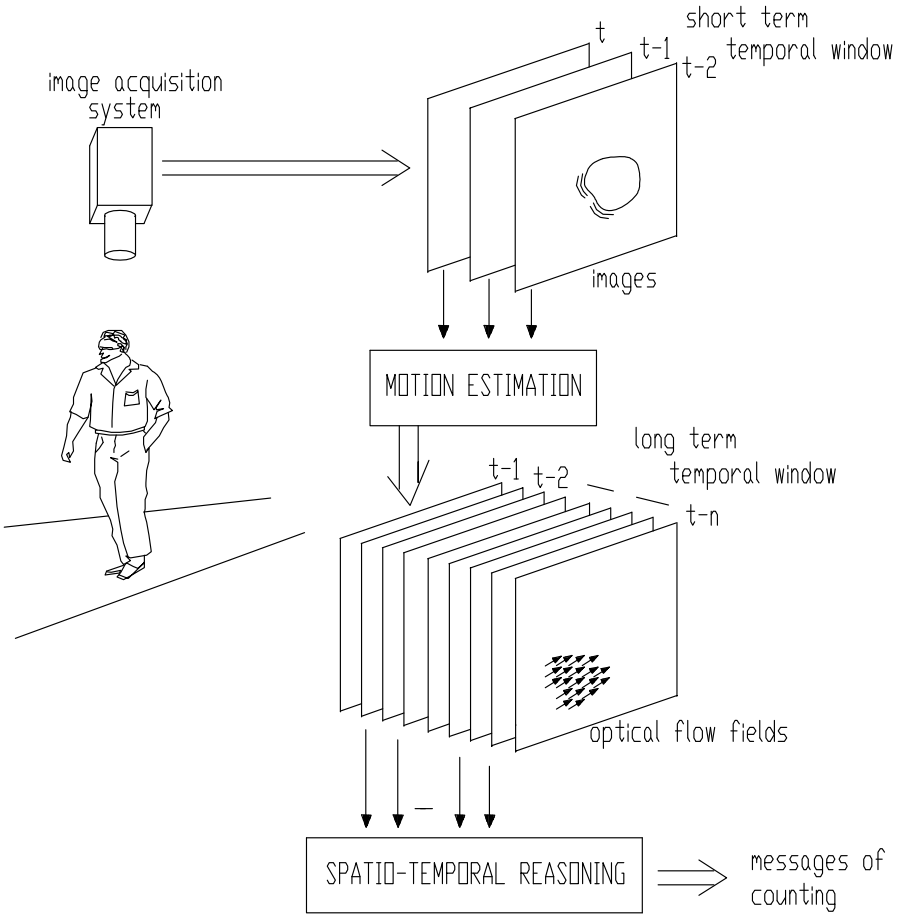


Figure 1: The overall process.

2.2 System operation

In our approach requirements expressed in Sec.2.1 are coped through motion estimation with an optical flow technique and spatio-temporal domain analysis. In Fig.1, the general schema of the counting process is reported.

Optical flow is obtained by analysing the image sequence on a short-term temporal window. It provides support for the identification of image features velocity. The collection of optical flow fields on a long-term temporal window is considered to be the spatio-temporal description of object behavior. Spatio-temporal domain analysis supports the counting of people getting on/off the bus through tracking of image features traces. Since this information is too noisy to be automatically interpreted for counting moving objects, the spatio-temporal domain is smoothed by means of a regularization-based approach for rejecting noise [18]. Tracking of simple features on a discrete version of the smoothed spatio-temporal is performed in order to understand if the moving objects observed are getting on/off the bus or are standing still under the image acquisition view area.

The optical flow estimation technique is expounded in Sec.3. Spatio-temporal domain analysis is

reported in Sec.4.

2.3 Working environment

The working environment is shown in Fig.2. In order to count people getting on and off a bus, two image acquisition systems (CCD-based cameras) are placed on the bus's ceiling, just over the stairs, one for each entrance lane divided by an iron barrier.

It is reasonably assumed that only one person at a time can go through each entrance lane, but several people can be present at the same time in the view area more or less close to each other (this is what usually occurs in crowding conditions). These people can be doing different things (such as getting on or getting off the bus from the same door), either *without stopping in the view area* or *stopping in the view area (and perhaps swinging), and moving again to get on or off the bus.* Since image acquisition systems are very close to the heads of moving people, the image acquisition view area cannot include the full shape of the moving objects.

The sequence in Fig.3 shows a person which is getting onto a bus. The stair steps and the metallic barrier dividing the door into two lanes can be noted. The shapes of the passing people are not completely focused and are only partially included in the image frames; their form also changes in the sequence of frames.

3 Optical Flow Estimation

3.1 Fundamentals of optical flow

Optical flow techniques provide a solution for the motion estimation problem starting from the observation of brightness changes in the image plane [1], [19], [20], [21], [22], [23], [24], [2], [25].

The optical flow is the field of the image brightness feature velocities, and therefore, differs from the perspective projection of 3D motion on the image plane (i.e., the velocity field) [26], [22], [25]. However, the estimation of an approximate velocity field, such as optical flow, can be very useful for many applications not requiring precision as in our case.

Most of the techniques described in literature for estimating optical flow fields use the so-called Optical Flow Constraint (OFC) equation, which derives from the observation that the changes in image brightness $E(x(t), y(t), t)$ of each point in the image are supposed to be stationary with respect to the time variable (i.e., $dE/dt = 0$):

$$E_x u + E_y v + E_t = 0, \tag{1}$$

in which the abbreviation for partial derivatives of image brightness has been introduced, and u, v correspond to dx/dt, dy/dt, and represent the components of the local velocity vector $\mathbf{V}$ along the x and y

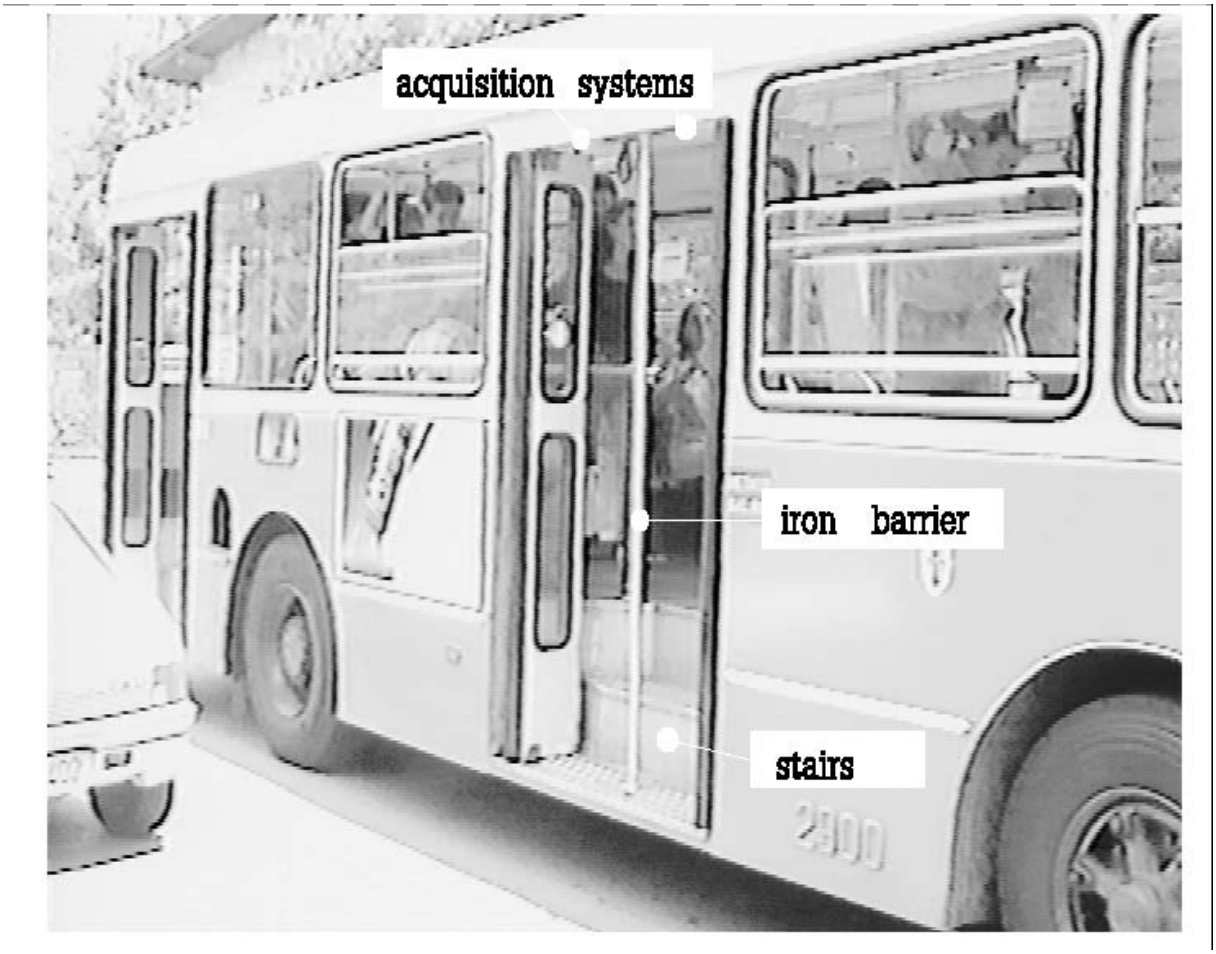


Figure 2: A two lane bus door and the view areas.

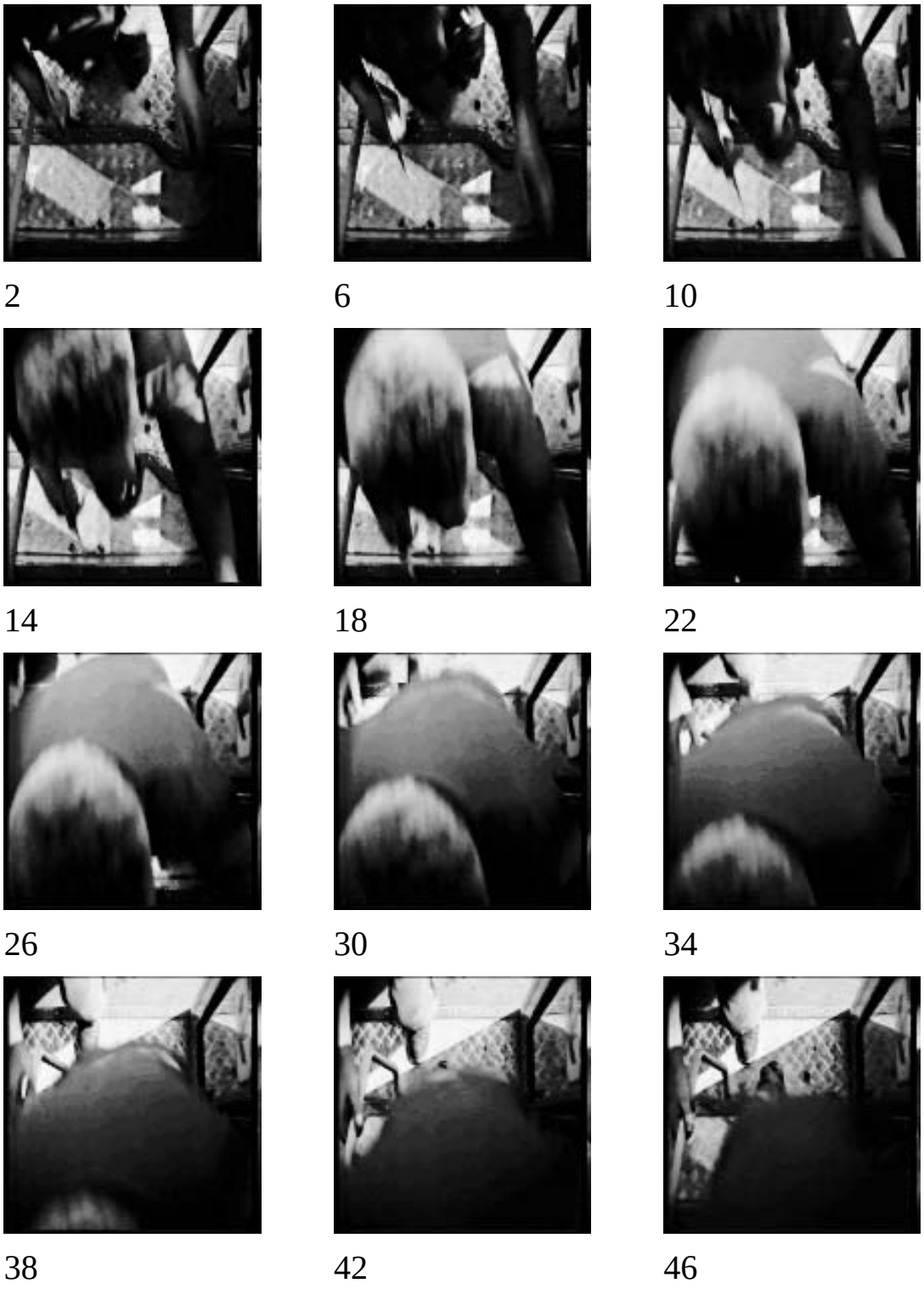


Figure 3: Typical sequence of images where people are getting on a bus (frames: 2, 6, 10, 14, 18, 22, 26, 30, 34, 38, 42, 46) (image resolution: 128×128 pixels).

directions on the image plane, respectively.

In the literature, two main approaches for optical flow estimation are identified: the *regularization-* and the *multiconstraint-based* approaches.

The *regularization-based* approaches consider optical flow estimation as an ill-posed problem. Solutions are obtained by minimizing a functional where a smoothness constraint is appropriately weighted to regularize the solution. The functional is minimized by using calculus of variations, and leads to define iterative solutions [1], [19], [27], [28].

The *multiconstraint-based* approaches for estimating optical flow fields are based on the fact that it is usually possible to define more than one constraint equation [29], [30]. They can be used to define an over-determined system of equations with u and v as unknowns evaluated at the same point in the image [20], [21], [23], [24], or considering that all the constraint equations which can be defined in the neighborhood around the estimation point represent the same optical flow field [31], [32], [2], [33]. The latter approach is commonly referred to as a "multipoint" approach. The overdetermined system of equations can be solved by using the least-square technique or by other means [18], [2], [34].

In general, optical flow estimations present two main problems.

The first is the presence of discontinuities in the optical flow field: these are due to image brightness discontinuities which are originated by the presence noise, too crisp patterns on the moving object surfaces, occlusions between moving objects, and object velocities which are too fast for the measuring system. Generally speaking, the presence of discontinuities can be overcome (or at least attenuated) by filtering the image with a 2D or 3D Gaussian smoothing operator at the expense of computational effort [35].

The second problem is the so-called "problem of aperture", found in the human vision, too. It derives from the impossibility to recover the direction of motion univocally if the object is observed through an aperture which is smaller than the object size. In this context, the references of the object under observation (such as textures – e.g., patterns) are not enough to make perception of the transversal component of the object motion, and only the component of apparent velocity which is parallel to ∇E can be detected [36], [37].

3.2 Estimation technique

In our counting system we use a multipoint-based technique for optical flow estimation. Robustness with respect to noise and behavior coherent with the human vision in the presence of aperture conditions have been discussed by the authors in [38], [37], and [33].

The *multipoint* approach for optical flow estimation used in the system is based on the fact that, considering that the optical flow changes follow a law which is approximatively linear, a smoothed solution

for optical flow estimation can be obtained from a linear approximation of the OFC equation in the neighborhood of the point under consideration [31], [32], [2] (this assumption is valid only if the optical flow field under observation is smooth). Consequently, a set of similar constraints in the neighborhood of a pixel yields an over-determined system of equations.

A multipoint solution based on the OFC equation (1) is obtained in the discrete domain at the finite differences. Thus, for the estimation of velocity components for the pixel under consideration, an over-determined system of $N \times N$ constraint equations in 2 unknowns, is defined,

$$E_{t(i,j,t)} + E_{x(i,j,t)}u + E_{y_{(i,j,t)}}v = 0.$$

for all (i, j) in an $N \times N$ neighborhood of the estimation point; where N is the dimension of the image segment side of the neighboring pixels, and $N \geq 2$.

In this technique, a large value of N will lead to smooth optical flow estimations, and loss in resolution in the estimation of velocity vectors.

The over-determined system of equations is solved by using a least-squares technique. In particular, after the estimation of image brightness derivatives in each pixel an over-determined system of $N \times N$ OFC equations in 2 unknowns is defined:

$$A\mathbf{V} + K = 0,$$

where $\mathbf{V}$ is the optical flow vector with components u, v; $A \in \mathcal{R}_{N^2 \times 2}$ matrix of coefficients, with $a_{r,1} = E_{x_r}$ and $a_{r,2} = E_{y_r}$; and $K \in \mathcal{R}_{N^2}$ vector with known terms $k_r = E_{t_r}$ for $r = 1, .., N^2$.

The solution of the over-determined system of equations by means of the least-squares technique can be obtained by using the pseudo-inverse technique, which transforms the above system of equations into a determined system of equations:

$$\hat{A}\mathbf{V} + \hat{K} = 0, \tag{2}$$

where $\hat{A} = A^T A$, and $\hat{K} = A^T K$ (i.e., A^T is the transpose of A). This system of equations can be solved by using traditional techniques such as LU decomposition, Gauss Jordan, etc.. In our case the system (2) is composed of 2 equations in 2 unknowns, and the direct solution is used.

Among the N^2 OFC equations, those which have the E_t under a chosen threshold are ignored and considered as insignificant constraints. Moreover, the constraint equations which have too large values for E_x and E_y are also neglected, hypothesising that there is a high probability that such large values are originated by noise.

3.2.1 Computational complexity

The explicit complexity of the solution proposed for estimating an optical flow field on an $M \times M$ image on a sequential machine is:

M	512	256	128	64	32	16	8
OFC-based	70	17.5	4.3	1.1	0.27	0.068	0.017

Table 1: Millions of floating point operations (MFLOP), with $G = 1$, $N = 5$, for different image dimensions.

$$C() = 3M^2 + 8N^2 \left[\frac{M-d}{G}\right]^2 + 8\left[\frac{M-d}{G}\right]^2, \tag{3}$$

where $d = 2(1 + \frac{N-1}{2})$ is due to the image boundaries, and the first term is due to the estimation of the partial derivatives of image brightness, which are obtained by using central differences.
The second additive term of (3) is due to the least-squares technique for calculating $\hat{a}_{i,j}$ and $\hat{k}_i$ (for $i = 1, 2$, and $j = 1, 2$), where G is the distance between two spatially consecutive estimation points, and $[x]$ is the greatest integer number lower than x; and the third term is determined by the method for solving the final system of equations (2). By improving the G value, less smooth optical flow fields are obtained.

As can be seen from (3), the asymptotical complexity of the solution proposed is:

$$O\left(\frac{M^2N^2}{G^2}\right).$$

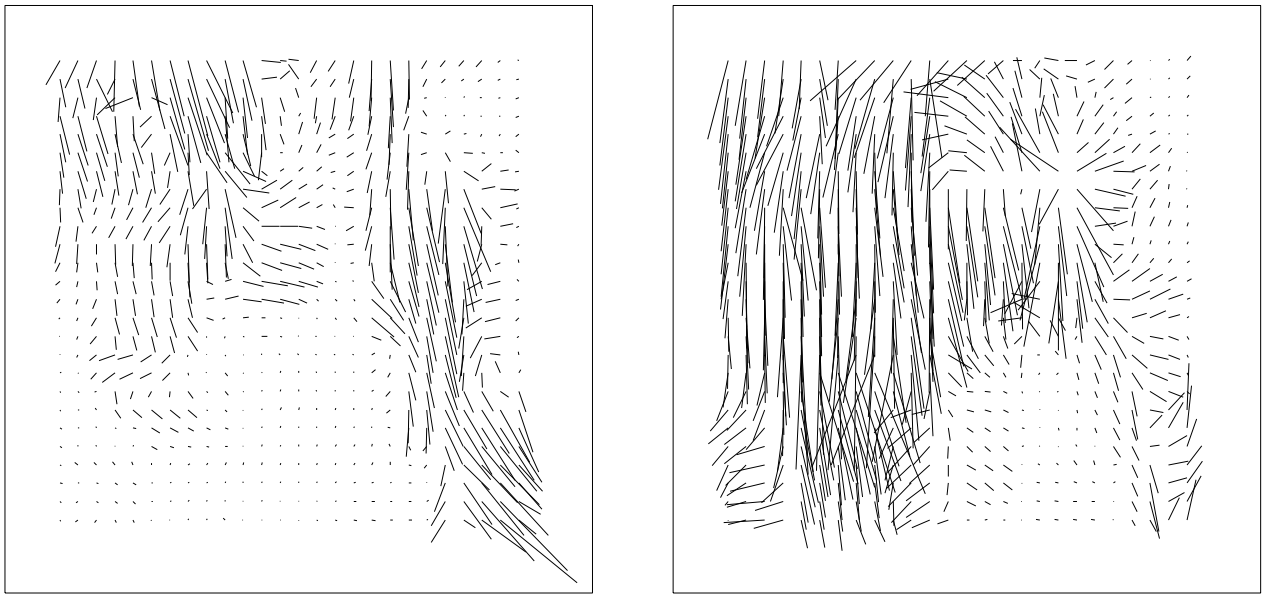
The computational cost in terms of floating point operations is reported in the following table. This cost is useful in order to evaluate the computational power required for performing optical flow estimation with the proposed solution. In Tab.1 the number of floating point operations (expressed in millions, MFLOP), with $G = 1$, and $N = 5$ as a function of the image dimension M, is reported. In this case, $(M-6)^2$ velocity vectors are estimated at each time interval. It has been used $G = 1$, $N = 5$ since these values represent a good compromise between the estimation quality, noise robustness and the computational cost. For this reason, these values have been used in our experiments.

Tab.2 shows the number of MFLOP per second (MFLOPS) needed to estimate optical flow fields in real-time – i.e., at video-rate frequency, 25 optical flow estimations per second – as a function of the dimension of the image, M.

Figures in Tab.2, show that it is possible to implement both the algorithm proposed by using low-cost processors with a floating point unit such as Analog Device ADSP 21020, Motorola 96000, Intel i860, etc., provided that the image size is not too large. The number of MFLOPS should be divided by at least G^2 if the estimation of velocity vectors is required only every G pixels, in both x and y direction.

M	512	256	128	64	32	16	8
OFC-based	1750	437	107	27.5	6.7	1.7	0.42

Table 2: Millions of floating point operations for second (MFLOPS), with $G = 1$ and $N = 5$, for different image size.



6 18

Figure 4: Optical flow fields estimated from the sequence of images presented in Fig.2 by using the proposed solution with $G = 1$ and $N = 5$ (frames: 6 and 18 with resolution of 32×32) .

3.2.2 Results of optical flow estimation

In order to test the robustness of the optical flow estimation technique used, several experiments were performed. Particular attention was given to testing algorithm responses with respect to changes in illumination and weather, deformations due to non-rigidity of moving objects, and critical effects due to the out-of-focus and the incomplete vision of the moving objects' shape in the view area.

Fig.4 illustrates optical flow fields estimated by using the method proposed on the sequence reported in Fig.3.

As can be seen, the shape of the moving person is not immediately detectable from the optical flow fields. This is mainly due to the fact that the body of the person is *non-completely focused and included* in the view area of the image acquisition system. In addition, the moving object is *non-rigid* and its parts are moving at *different velocities* in intensity and direction. In such conditions, the same moving object has regions of its shape characterized by contrasting velocities which change from one image to the successive. Therefore, segmentation process based on the optical flow fields leads to estimate moving

object shapes in one image which are strongly different with respect to those obtained in the successive one, and is not suitable to be used as a basis to track the moving objects.

In addition, optical flow estimation cannot solve the problems related to all moving objects' behaviors, as required by a system that performs people counting. These problems can be solved only by long-term analysis which interprets optical flow fields in time. People's flow should be interpreted as a flow of elementary moving particles (each of which has its own optical flow vector) and not as a sequence of single passages of large objects. Then, the number of passing people should be estimated by knowing the geometry of the acquisition system (i.e., distance from moving objects, etc., as in our case) and the dimensions (in terms of projected shape) of moving people. The sum of the flow of particles divided by the median flow due to a person should give the number of people passed through the door. This subject is addressed in detail in the next Section.

4 Interpreting Optical Flow Fields in Time

Let $\mathbf{V}_{(i,j,t)}$ with components $u_{(i,j,t)}$, and $v_{(i,j,t)}$ be the velocity in an image point with coordinates i, j, at the time t. In the working environment taken into consideration, the flow of the people is constrained by metallic barriers along the direction of the y-axis of the image plane (see Sec.2), and therefore, the motion of interest is only described by the component $v_{(i,j,t)}$ of optical flow.

The problem of counting moving objects which only move along a direction parallel to the y-axis has been solved by estimating the flow of the image brightness features in time through a transversal section S on the image plane, usually called slit [16], [39] (this could be performed by using a simple linear CCD placed along section S). By collecting these sections in time, an EPI is obtained [14], [15], [16]. Unfortunately, this type of spatio-temporal image is not suitable for solving the above-mentioned problems. In fact, in such images it is not possible to distinguish between *two consecutive and fast moving objects which cross the view area without stopping* from a *single object which enters in the view area, stops its motion and then restarts*, which both are common situations when entering a bus. It is evident that this approach should lead to incorrect results in counting people.

To solve this problem, a different form of spatio-temporal reasoning is proposed, which makes it is possible to measure feature flow and analyze the spatio-temporal behavior of moving objects.

4.1 Spatio-temporal feature flow domain

The feature flow at time t through a section S, which is parallel to the x-axis, can be estimated by:

$$F_{(S,t)} = \int_S \mathbf{V}(t) \cdot \hat{\mathbf{y}} \, dx \tag{4}$$

where $\hat{\mathbf{y}}$ is the unit vector of the y-axis. In the discrete field, a measure of features' flow can be obtained

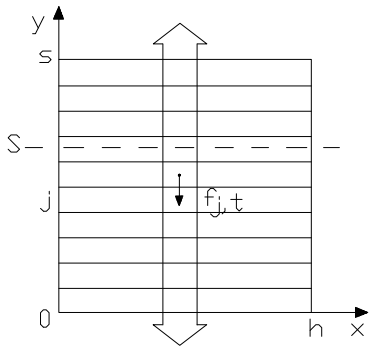


Figure 5: Optical flow field segments in the image plane at time t.

by dividing the optical flow field into s horizontal segments (rows along the y-axis, $s = [(M - d)/G]$, see Fig.5), and by collecting one value for each segment:

$$f_{(j,t)} = \sum_{i=1}^{h} v_{(i,j,t)}, \qquad \text{for} \quad j = 1, .., s,$$

where h is the number of optical flow vectors present in each row of the full optical flow field ($h = [(M - d)/G] = 26$ in the optical flow fields presented in Fig.4). This approximation is obtained when the dimension of each segment along the section S is taken to be unitary, while smoother feature flows can be computed by considering multiple rows for each estimation of $f_{(j,t)}$.

In Fig.6, the spatio-temporal behavior of $f_{(j,t)}$ for the test sequence of Fig.3 is reported. Positive hills correspond to people who are getting onto a bus (e.g., around frame 20), while negative hills correspond to people who are getting off a bus (e.g., around frame 150). In Fig.6, they can be recognized 4 people who get into a bus (around the 20th, 71st, 110th, and the 120th frame, the last two people are very close connected to each other (see Fig.7)), and 3 people who are quickly getting off (around the 141th, 150st, and the 160th frame).

Simply counting peaks in a transversal slice $f_{(S,t)}$ (e.g., Fig.8 shows a transversal slice of the surface of Fig.6) of $f_{(j,t)}$ that are higher than a predefined threshold leads to incorrect results when people stop in the view area, because each restart produces a new peak. To avoid this problem, and understand what people are really doing when they cross the view area, it is necessary to analyse a large spatio-temporal window of a feature flow surface. A threshold for identifying the most prominent feature flow regions can also be defined on this surface. Fig.9 illustrates the map representing $\mathcal{F}_{(j,t)}$ obtained from surface $f_{(j,t)}$ of Fig.6 according to:

$$\mathcal{F}_{(j,t)} = \begin{cases} +1 & \text{if} \quad f_{(j,t)} > D_{in} \quad \text{“entering”} \\ -1 & \text{if} \quad f_{(j,t)} < D_{out} \quad \text{“exiting”} \\ 0 & \text{otherwise} \qquad\quad \text{“motionless”}; \end{cases}$$

two different thresholds D_{in} and D_{out} were imposed for the feature flow values generated by inputs and outputs, respectively. This map was obtained by coloring the entering flow of gray, the absence of flow

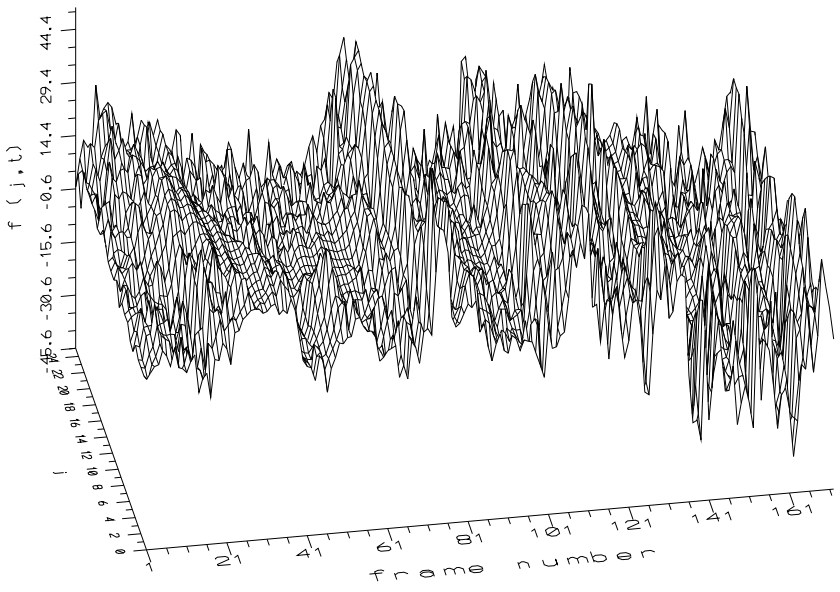


Figure 6: The spatio-temporal behavior of $f_{(j,t)}$.

black, and the exiting flow white. The moving objects are identified by transversal segments (referred to as *traces* in the rest of the paper) connecting the upper and lower map limits.

Nevertheless, even when the thresholds are imposed carefully, the maps obtained are affected by too much noise to allow automatic interpretation (see Fig.9). Hence, a smoothing action must be performed on feature flow $f_{(j,t)}$ before its transformation into the map, $\mathcal{F}_{(j,t)}$. Then, the smoothed map obtained can be interpreted in order to count the passing people.

Please note, that imposing a threshold on the feature flow $f_{(j,t)}$ is very different from imposing a threshold on the optical flow field itself. In the latter case, threshold imposition leads to neglecting objects which are moving slower than a given value (this condition in not acceptable for the application proposed, because people usually get on a bus very slowly). On the contrary, threshold imposition on the feature flow $f_{(j,t)}$, leads to neglecting objects which generate a feature flow lower than a given value. This limitation is not restrictive because people's dimensions do not change significantly, therefore only the presence of very small objects (even if they are moving very quickly) is neglected.

We observed in our experiments that a suitable value for such thresholds is about 1/3 of h; with these values, people moving very slowly are detected even when they are very small. Lower values make noise too prominent, while high threshold values must not be set because they lead to losing people's movements, such as when a passing person swings under the view area. Since people usually get off a bus faster than they get onto it, it is appropriate to impose different values on threshold D_{in} and D_{out} in order to detect the entering and the exiting conditions.

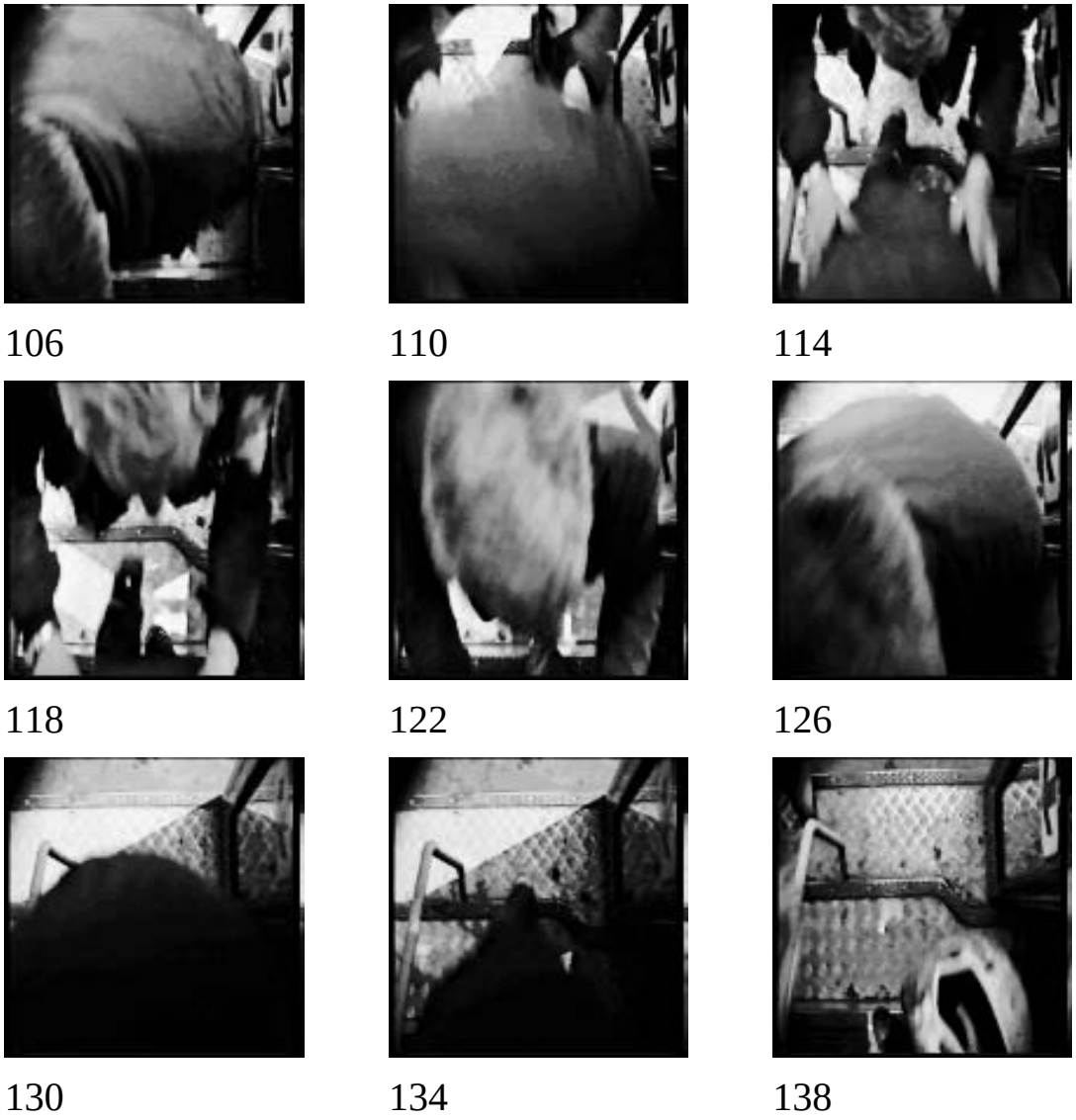


Figure 7: Sequence in which two close people are getting on a bus (frames: 106, 110, 114, 118, 122, 126, 130, 134, 138) (image resolution: 128×128).

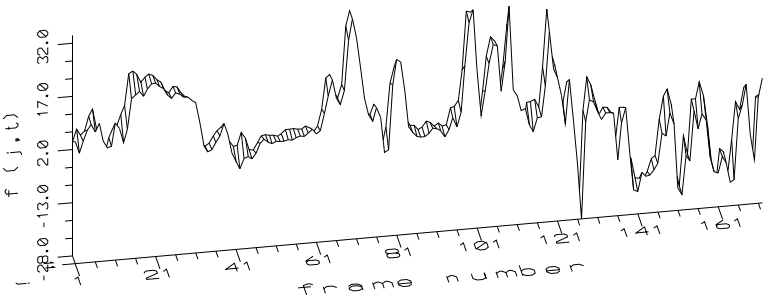


Figure 8: Temporal behavior of $f_{(j,t)}$, for $j = 12$, of Fig.6.

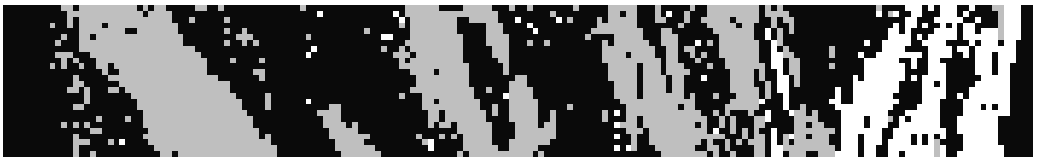


Figure 9: Map $\mathcal{F}_{(j,t)}$ of feature flow obtained with $D_{in} = 10$ and $D_{out} = -10$ from $f_{(j,t)}$, of Fig.6.

In Fig.10 an ideal noise-free map $f_{(j,t)}$ is illustrated. It has been obtained by using a synthetic image sequence in which a rigid object crosses the view area. It contains several traces corresponding to typical object behaviors. The first trace refers to an object entering in the view area and standing still before going out of it later. The second trace refers to a moving object which stops during the crossing and then moves again in the same direction. The last trace is very fragmentary, and corresponds to a crossing object which makes various stops and direction changes (swinging) under the view area.

In order to count people reliably we must be able to manage common ambiguous situations such as those in which there are two closely connected objects crossing the view area (such as around the 115th frame in Fig.7). To this end, it could be useful to have a measure of the moving object velocity and

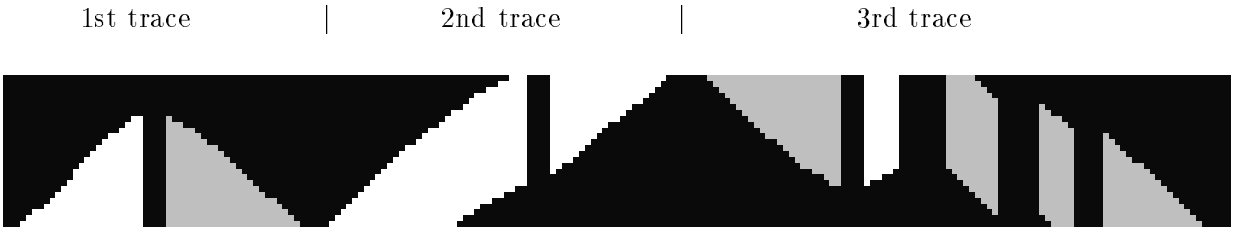


Figure 10: Typical behaviors in the spatio-temporal feature flow domain.

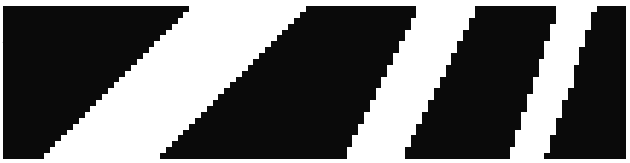


Figure 11: Traces of moving objects in the spatio-temporal feature flow domain.

dimension to find out when the trace under observation is due to a slow single object or to two objects moving at normal velocity.

Object velocity can also be determined from a feature flow map. Fig.11 illustrates a map in which an object, having constant dimensions, crosses the view area when getting off the bus, at different velocities. The slope of a trace corresponds to the object's velocity. The first trace represents an object which crosses the view area at a velocity equal to 1 pixel/frame, in the second trace the object goes at 2 pixel/frame, and, in the third it goes at 3 pixel/frame. Thus, considering the acquisition system geometry is possible to recover the true moving object velocity from a feature flow map.

If the moving objects have a constant area size, l (i.e., if we assume that do not change their dimension with time), integrating $F_{(S,t)}$ on a generic section S the area dimension of the moving object going through the section S on the image plane is obtained. In the discrete domain the integration is $m_{(j,T)} = \sum_{t=t_o}^{T} f_{(j,t)}$ and thus $m_{(j,T)}/l$ could be interpreted as the number of objects getting on, minus the number of objects getting off during the time interval $T - t_o$.

This technique cannot be used in our working environment, because the objects to be counted do not have a constant dimension. Moreover, due to the approximations adopted to estimate the first-order partial derivative of the image brightness, with gradient-based techniques produce reliable estimations of optical flow only when object displacements between two consecutive frames are not wider than 1 pixel. Therefore, to have a reliable detection of a moving object crossing the view area, the corresponding trace in the feature flow domain must be "tracked" in time.

In the following Section the techniques used for smoothing the feature flow domain and tracking traces with continuous labeling are expounded in detail.

4.2 Smoothing the feature flow domain

The presence of noise on a map can be reduced by using a traditional low-pass filter (e.g., average, median, etc.) directly on the $F_{(S,t)}$, or by using stochastic relaxation with clips [40], [2], [7] on $\mathcal{F}_{(S,t)}$. The first method does not permit an adequate control of smoothing in different ways along the y- and t-axes of map, without a strong reduction of the signal level; the latter method could solve these problems but

is computationally heavy.

We propose a deterministic relaxation as a solution for these problems, that is a regularization-based approach: the problem is posed as the minimization of the functional:

$$\iint (F - W)^2 + \alpha^2 (W_y)^2 + \beta^2 (W_t)^2 \;\; dy\, dt, \tag{5}$$

where $F = F_{(S,t)}$ is the feature flow; $W = W_{(S,t)}$ is the smoothed feature flow function; $(W_y)^2$, $(W_t)^2$ are the smoothness constraints on $W_{(S,t)}$; α, β are the weighting factors which are used to handle the smoothing at different intensities along y- and t-axes.

The functional (5) is minimized by using the calculus of variations [41]; this leads to the partial derivative equation:

$$F - W + \alpha^2 W_{yy} + \beta^2 W_{tt} = 0,$$

which can be solved by using natural boundary conditions. By discretizing the above equation, by means of the finite difference method we obtain:

$$W_{(j,t)y} = W_{(j+1/2,t)} - W_{(j-1/2,t)},$$

$$W_{(j,t)yy} = W_{(j+1,t)} - 2W_{(j,t)} + W_{(j-1,t)}.$$

Hence, the iterative solution is obtained:

$$W^{n+1}_{(j,t)} = \frac{f^n_{(j,t)} + \alpha^2 (W^n_{(j+1,t)} + W^n_{(j-1,t)}) + \beta^2 (W^n_{(j,t+1)} + W^n_{(j,t-1)})}{2\alpha^2 + 2\beta^2 + 1}, \tag{6}$$

in which the same finite difference technique is also used along the t axis. In the estimation, the discrete version of $F_{(S,t)}$, that is $f_{(j,t)}$, is considered, and $f^n_{(j,t)} = W^n_{(j,t)}$. This smoothing equation is used for regularizing the function $f_{(j,t)}$. The regularization process is performed on a running window of a map having a $\Delta T \times s$ dimension during its production in time as depicted in Fig.12. Since the regularization process is performed on the running window, the number of iterations performed on a given neighborhood is equal to $I\Delta T$, where I is the number of iterations preformed at each time instant in the window ΔT in time, by using (6).

Fig.13 shows the surface $W_{(j,t)}$, obtained by smoothing function $f_{(j,t)}$, of Fig.6. After smoothing, thresholding is performed on the $W_{(j,t)}$ domain to obtain a discrete map $\mathcal{W}_{(j,t)}$ preserving the main moving objects' traces (see Fig.14):

$$\mathcal{W}_{(j,t)} = \begin{cases} +1 & \text{if } W_{(j,t)} > D_{in} \\ -1 & \text{if } W_{(j,t)} < D_{out} \\ 0 & \text{otherwise} \end{cases}$$

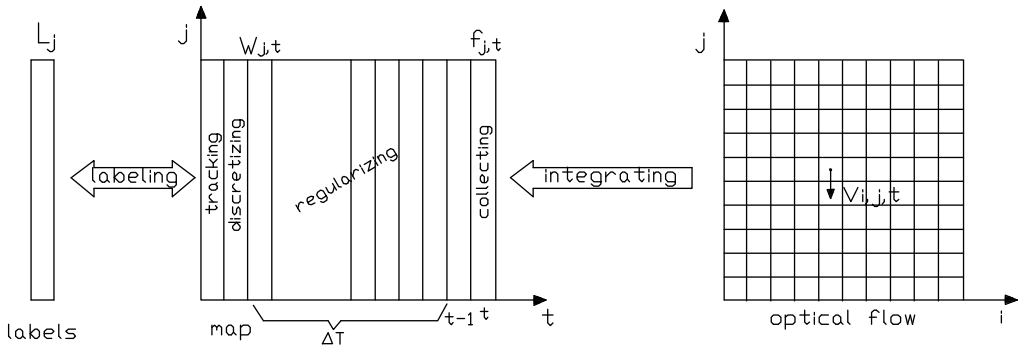


Figure 12: The collecting, smoothing and interpretation processes in time.

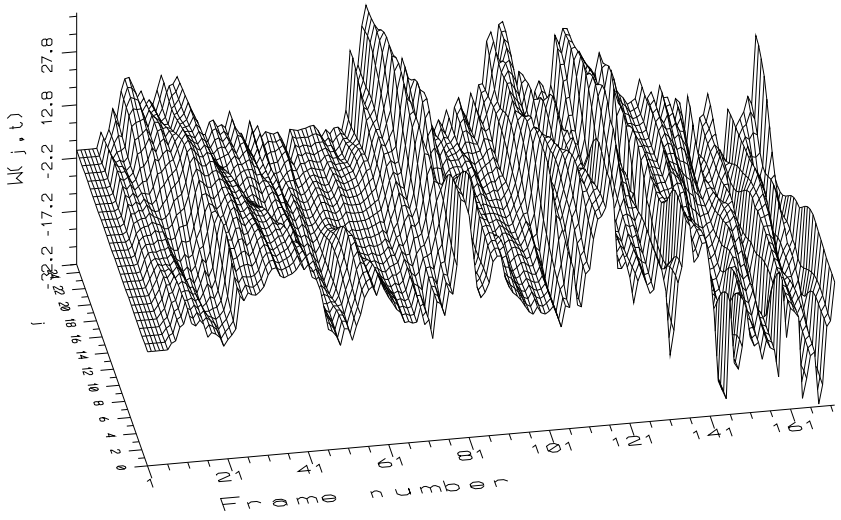


Figure 13: Regularized spatio-temporal behavior $W_{(j,t)}$, of feature flow $f_{(j,t)}$, obtained with $\alpha = 2$, $\beta = 0.2$, $I = 2$, and $\Delta T = 4$.

Figure 14: Map $\mathcal{W}_{(j,t)}$ of feature flow obtained with $D_{in} = 10$ and $D_{out} = -10$ from $W_{(j,t)}$, of Fig.13.

As can be noted comparing the map of Fig.14 with that of Fig.9, the weighting factors α and β are used to obtain a propagation effect with good regularization along the j-axis and a slight regularization along the t-axis. In this way, each trace crossing the view area is evidenced. By our experiments we have been observed that a good effect of smoothing rejecting noise without to loss too much in resolution is to adopt $I = 2$, $\Delta T = 4$, $\beta = 0.2$ and $\alpha = 2$.

In the regularized map of Fig.14 we can count four objects getting on the bus, at different velocities, and three objects getting off the bus at a higher velocity (as usually happen when people leave the bus).

After having obtained $\mathcal{W}_{(j,t)}$ the detection and tracking of traces on the map make it possible to count the moving people. Trace tracking is based on continuous labeling as shown in the next subsection.

4.3 Tracking with continuous labeling

The tracking of traces in the map is performed by means of the technique of continuous labeling. Fig.15(a-c) illustrates the typical result of the continuous labelling technique applied to a fragmented map with $s = 9$.

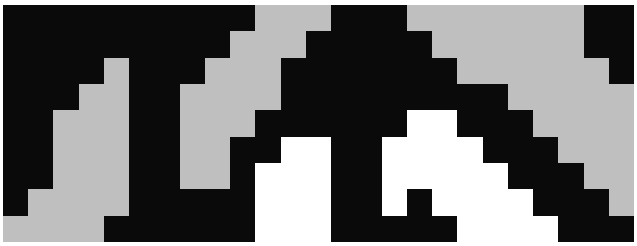
Each label is activated when a new arrival from the lower or upper limits appears in the map. The labeling process (see Fig.15(b)) consists in comparing an old labelled *slice* of the map, $L_{(j)}$, with the feature flow map, $\mathcal{W}_{(j,t)}$, at time $t - \Delta T - 3$ at each time instant (see Fig.12). With this mechanism a trace is followed also when the object stay motionless (see label 1 in the history of the labelled *slices* Fig.15(c)), because positions are stored in the slice of labels, $L_{(j)}$. The same label, that has been used to track a crossing object, can be used again later.

Simple rules based on label movements in $L_{(j)}$, are applied for counting the passing people. In other words the passage of a person results in the labelled space as a simple segment which connects the lower with the upper limits of the map, even if the trace was very fragmented.

4.4 Computational complexity

The computational cost of interpreting the optical flow fields at each time instant (considering $s = h = [(M - d)/G]$), is given by the summation of distinct sequential costs related to:

- estimating of the feature flow map, $f_{(j,t)}$, for $j = 1, .., s$: $O([(M - d)/G]^2)$;
- spatio-temporal smoothing of $f_{(j,t)}$, to obtain $W_{(j,t)}$, for $j = 1, .., s$; this process consists in I iterations on a temporal window of ΔT instants: $O(I \Delta T\, [(M - d)/G])$;
- thresholding of $W_{(j,t)}$, in order to obtain $\mathcal{W}_{(j,t)}$, for $j = 1, .., s$: $O([(M - d)/G])$;
- continuous labelling, which consists in comparing the position of the labels in $L_{(j)}$ with $\mathcal{W}_{(j,t-\Delta T-3)}$, for $j = 1, .., s$: $O([(M - d)/G])$.



(a)

0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1	1	1	0	0
0	0	0	0	1	0	0	0	1	1	1	0	0	0	0	0	0	0	1	1	1	1	1	0
0	0	0	1	1	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1	1
0	0	1	1	1	0	0	1	1	1	0	0	0	0	0	0	2	2	0	0	0	1	1	1
0	0	1	1	1	0	0	1	1	0	0	2	2	0	0	2	2	2	2	0	0	0	1	1
0	0	1	1	1	0	0	1	1	0	2	2	2	0	0	2	2	2	2	2	0	0	0	1
0	1	1	1	1	0	0	0	0	0	2	2	2	0	0	2	0	2	2	2	2	0	0	0
1	1	1	1	0	0	0	0	0	0	2	2	2	0	0	0	0	0	2	2	2	2	0	0

(b)

0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1	1	1	0
0	0	0	0	0	1	1	1	0	1	1	1	0	0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	2	2	0	0	0	1	1
0	0	0	1	1	1	1	1	1	1	0	0	2	2	2	2	2	2	2	2	0	0	0	1
0	0	0	1	1	1	1	1	1	1	0	2	2	2	2	2	2	2	2	2	2	0	0	0
0	0	1	1	1	1	1	1	0	0	0	2	2	2	2	2	2	0	2	2	2	2	0	0
0	1	1	1	1	0	0	0	0	0	0	2	2	2	2	2	0	0	0	2	2	2	2	0

(c)

Figure 15: Typical history of the continuous labelling process: (a) several instants of a fragmented map $\mathcal{W}_{(j,t-\Delta T-3)}$, (b) labels associated with the map, (c) history values for the slice of labels, $L_{(j)}$, corresponding to the time instants of (b).

M	512	256	128	64	32	16	8
OFC-based	7.13	1.92	0.549	0.168	0.054	0.017	0.003

Table 3: Millions of floating point operations per second (MFLOPS), with $G = 1, N = 5, I = 2, \Delta T = 4$, for different image sizes.

According to this, the asymptotical complexity of optical flow interpretation for counting moving objects is an

$$O(\frac{M^2}{G^2}),$$

where $M/G > I\Delta T$ because in our experiments: $I = 2$ and $\Delta T = 4$ while $G = 1$, $N = 5$, and $M = 32$.

It can be noted that, the computational complexity of interpretation is negligible compared to the complexity of optical flow estimation, which is an $O(M^2N^2/G^2)$. This can be seen comparing Tab.2 with Tab.3, in which the number of MFLOP per second (MFLOPS) needed for interpreting the optical flow at a video-rate frequency as a function of the dimension of the image are shown.

5 Notes on Hardware Implementation

A general schema of the hardware architecture of our system is shown in Fig.16. The system is based on a floating point DSP, the Analog Device ADSP 21020, which sustains 66 MFLOPS. In our application, it processes images of 64×64 pixels (i.e., $M = 64$) with 8 bits per pixel. The image acquisition system (a CCD with a low cost lens) is directly integrated in the architecture.

Synchronisms used for the CCD are not consistent with neither the standard PAL nor NTSC, but are custom-defined. The generation of synchronisms is delegated to a specialized hardware (instead of using the DSP itself), in order to save CPU time and support a regular rate of 20 frames per second. The analog to digital converter (ADC) receives the signal /STR to start the analog to digital conversion directly by the synchronism generator, and only when conversion is terminated it sends an interrupt to the DSP, which reads the pixel value.

An RS232 serial interface communicates the number of people which passed through the supervised door to the system which controls the vehicle. A reset of the counting can be forced through this port. In a regular city bus, 6 of these sensors are mounted, one for each door lane. A microprocessor-based system for each bus collects the information from these and other sensors, and sends these data to the monitoring center of the town by means of a radio transmitter.

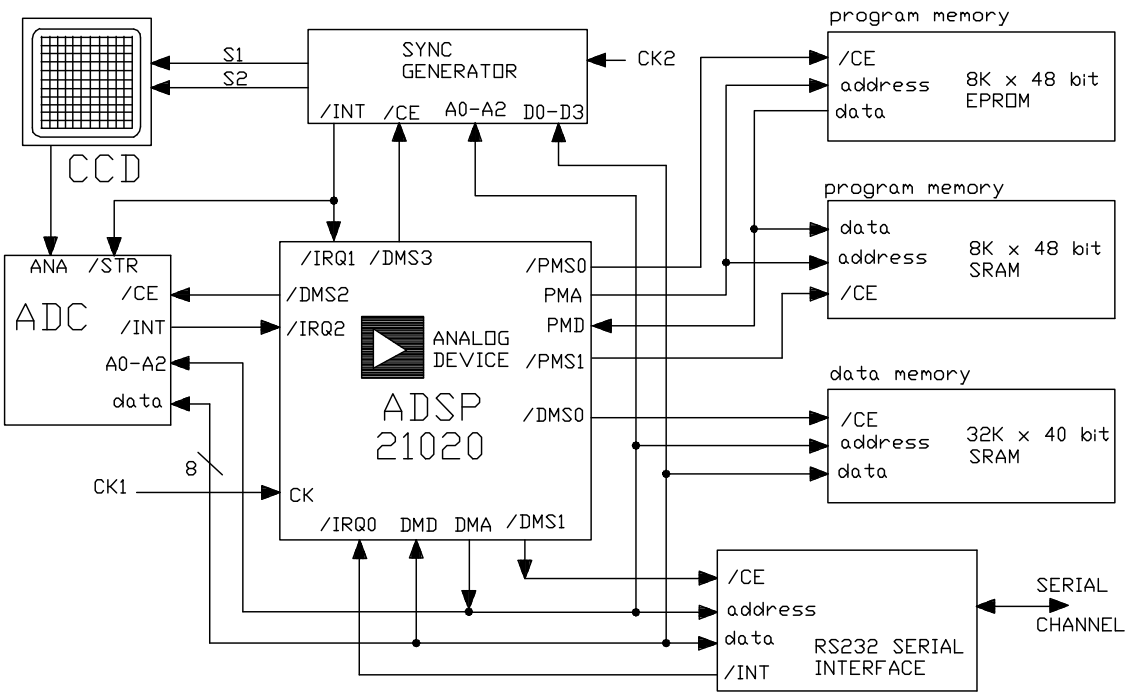


Figure 16: Physical system architecture. Only the main signals are reported.

6 Conclusions

In this paper, a system for solving the problem of counting people getting on/off a public bus has been presented; the system is based on optical flow estimation with associated spatio-temporal analysis. The system proposed can count passing people with high confidence even when they are not completely focused, and visible within the view area. Moreover, it is very robust with respect to noise due to image acquisition and people's behavior.

The low computational effort associated with the used technique allowed its implementation on a DSP based hardware architecture which is currently under testing.

Acknowledgments

Authors acknowledge ANALOG DEVICE Inc. for the support provided in the hardware implementation of the system.

References

[1] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[2] P. Nesi, A. Del Bimbo, and J. L. C. Sanz, "Multiconstraints-based optical flow estimation and segmentation," in *International Workshop on Computer Architecture for Machine Perception*, (Paris), pp. 419–426, DGA/ETCA, CNRS/IEF and MEN/DRED, December 1991.

[3] W. K. Pratt, *Digital Image Processing*. New York, USA: John Wiley & Sons, 1978.

[4] R. Jain, W. N. Martin, and J. K. Aggarwal, "Segmentation through the detection of changes due to motion," *Computer Vision, Graphics, and Image Processing*, vol. 11, pp. 13–34, 1979.

[5] R. Jain, "Difference and accumulative difference pictures in dynamic scene analysis," *Image and Vision Computing*, vol. 2, pp. 99–108, May 1984.

[6] A. Shio and J. Sklansky, "Segmentation of people in motion," in *Proc. of the IEEE Workshop on Visual Motion*, (Nassau Inn, Princeton, NJ, USA), pp. 325–332, IEEE Computer Society, 7-9 October 1991.

[7] D. W. Murray and B. F. Buxton, "Scene segmentation from visual motion using global optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, pp. 220–228, March 1987.

[8] P. Lalande and P. Bouthemy, "A statical approach to the detection and tracking of moving objects in an image sequence," in *Proc. of V European Signal Processing Conference, EUSIPCO'90 Barcellona, Spain*, pp. 947–950, 18-21 Sept. 1990.

[9] H. H. Baker and T. D. Garvey, "Motion traking on spatiotemporal surface," in *Proc. of the IEEE Workshop on Visual Motion*, (Nassau Inn, Princeton, NJ, USA), pp. 340–345, IEEE Computer Society, 7-9 October 1991.

[10] M. Allmen and C. R. Dyer, "Computing spatiotemporal surface flow," tech. rep., Computer Science Technical Report 935 University of Wisconsin, Madison, USA, May 1990.

[11] S.-L. Peng and G. Medioni, "Interpretation of image sequence by spatio-temporal analysis," in *Proc. of the IEEE Workshop on Visual Motion*, pp. 344–351, Irvine, California, USA: IEEE Computer Society, 20-22 March 1989.

[12] H. Baker and R. Bolles, "Generalizing epipolar-plane imeg analysis on the spatiotemporal surface," *International Journal of Computer Vision*, vol. 3, pp. 33–49, 1989.

[13] S.-L. Peng, "Temporal slice analysis of image sequences," in *Proc. IEEE Computer Vision and Pattern Recognition Conference, CVPR'91*, pp. 283–288, 1991.

[14] T. E. Boult and L. G. Brown, "Factorization-based segmentation of motion," in *Proc. of the IEEE Workshop on Visual Motion*, (Nassau Inn, Princeton, NJ, USA), pp. 179–186, IEEE Computer Society, 7-9 October 1991.

[15] H. Tamamoto, Y. Narita, A. Yanase, F. Saito, and K. Komatsu, "A measuring system for traffic flow of passers-by by processing ITV image in real time," in *Proc. of MVA'92 IAPR Workshop on Machine Vision Applications, Tokyo*, pp. 343–347, 7-9 Dec. 1992.

[16] T. Nakanishi and K. Ishii, "Automatic vehicle image extraction based on spatio-temporal image analysis," in *Proc. of 11th IAPR IEEE International Conference on Pattern Recognition, ICPR'92*, pp. 500–504, 30 Aug. - 3 Sept. 1992.

[17] A. Del Bimbo, P. Nesi, and J. L. C. Sanz, "Estimation and interpretation of optical flow fields for counting moving objects," in *Proc. of MVA'90 IAPR Workshop on Machine Vision Applications Tokyo*, (NEC Super Tower, Minato-ku, Tokyo, Japan), 7-9 Dic. 1992.

[18] B. G. Schunck, "Image flow segmentation and estimation by constraints line and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 1010–1027, Oct. 1989.

[19] H.-H. Nagel, "Displacement vectors derived from second-order intensity variations in image sequences," *Computer Vision, Graphics, and Image Processing*, vol. 21, pp. 85–117, 1983.

[20] R. M. Haralick and J. S. Lee, "The facet approach to optical flow," in *Proc. of Image Understanding Workshop (Science Applications Arlington), Va* (L. S. Baumann, ed.), 1983.

[21] O. Tretiak and L. Pastor, "Velocity estimation from image sequences with second order differential operators," in *Proc. of 7th IEEE International Conference on Pattern Recognition*, pp. 16–19, 1984.

[22] H.-H. Nagel, "On a constraint equation for the estimation of displacement rates in image sequences," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 13–30, January 1989.

[23] A. Verri, F. Girosi, and V. Torre, "Mathematical properties of the two-dimensional motion field: from singular points to motion parameters," *J. Opt. Soc. Am. A*, vol. 6, pp. 698–712, May 1989.

[24] A. Verri, F. Girosi, and V. Torre, "Differential techniques for optical flow," *J. Opt. Soc. Am. A*, vol. 7, pp. 912–922, May 1990.

[25] P. Nesi, A. Del Bimbo, and J. L. C. Sanz, "A unified approach to optical flow constraint analysis," tech. rep., Dipartimento di Sistemi e Informatica, Facolta' di Ingegneria, Universita' di Firenze, DSI-RT 17/92, Florence, Italy, 1992.

[26] A. Verri and T. Poggio, "Motion field and optical flow: Qualitative properties," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 490–498, May 1989.

[27] E. C. Hildreth, "Computations underlying the measurement of visual motion," *Artificial Intelligence*, vol. 23, pp. 309–354, 1984.

[28] P. Nesi, "Variational approach for optical flow estimation managing discontinuities," *Image and Vision Computing*, vol. 11, no. 7, pp. 419–439, 1993.

[29] H.-H. Nagel, "On the estimation of optical flow: Relations between different approaches and some new results," *Artificial Intelligence*, vol. 33, pp. 299–324, 1987.

[30] A. Mitiche, Y. F. Wang, and J. K. Aggarwal, "Experiments in computing optical flow with the gradient-based multiconstraint method," *Pattern Recognition*, vol. 20, no. 2, pp. 173–179, 1987.

[31] C. Cafforio and F. Rocca, "Tracking moving objects in television images," *Signal Processing*, vol. 1, pp. 133–140, 1979.

[32] M. Campani and A. Verri, "Computing optical flow from an overconstrained system of linear algebraic equations," in *Proc. of 3rd IEEE International Conference on Computer Vision ICCV'90, Osaka, Japan*, pp. 22–26, 4-7 Dec. 1990.

[33] A. Del Bimbo, P. Nesi, and J. L. C. Sanz, "Optical flow computation using extended constraints," tech. rep., Dipartimento di Sistemi e Informatica, Facolta' di Ingegneria, Universita' di Firenze, DSI-RT 19/92, Florence, Italy, 1992.

[34] D. Ben-Tzvi, A. Del Bimbo, and P. Nesi, "Optical flow from constraint lines parametrization," *Pattern Recognition*, 1993.

[35] P. Nesi, A. Del Bimbo, and D. Ben-Tzvi, "Algorithms for optical flow estimation in real-time on connection machine-2," tech. rep., Dipartimento di Sistemi e Informatica, Facolta' di Ingegneria, Universita' di Firenze, DSI-RT 24/92, Florence, Italy, 1992.

[36] A. Singh, *Optic Flow Computation: A Unified Perspective.* Los Alamitos, California, USA: IEEE Computer Society Press, 1991.

[37] A. Del Bimbo, P. Nesi, and J. L. C. Sanz, "Optical flow estimation by using classical and extended constraints," in *Proc. of 4th International Workshop on Time-Varying Image Processing and Moving Object Recognition*, 10-11 June 1993.

[38] A. Del Bimbo, P. Nesi, and J. L. C. Sanz, "Innovative multipoint solutions for optical flow estimation with different constraints," in *Proc. of International Conference on Image Analysis and Processing, IAPR, Bari, Italy*, 20-23 September 1993.

[39] K. Mase, A. Sato, Y. Suenaga, and K. i Ishii, "A fast object flow estimation method based on spacetime image analsysis," in *Proc. of MVA'92 IAPR Workshop on Machine Vision Applications, Tokyo*, pp. 199–202, 7-9 Dec. 1992.

[40] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 721–741, Nov. 1984.

[41] R. Courant and D. Hilbert, *Methods of Mathematical Physic*, vol. 1. New York, London: Interscience Publisher, Inc., 1955.

Contents