People counting in crowded scenes using multiple cameras

F. Dittrich*, A. L. Koerich
Pontifical Catholic University of Paraná
R. Imaculada Conceição, 1155
80215-901 Curitiba, PR, Brazil

L. E. S. Oliveira
Federal University of Paraná
R. Francisco H. dos Santos, 100
81531-990 Curitiba, PR, Brazil

Abstract

This paper presents a novel method for people counting in crowded scenes that combines the information gathered by multiple cameras to mitigate the problem of occlusion that commonly affects the performance of counting methods using single cameras. The proposed method detects the corner points associated to the people present in the scene and computes their motion vector. During the training step the mean number of points per person is estimated. The image plane is transformed to the ground plane using homography and weights are assigned to each corner point according to its distance to the camera since the farthest a person is from the camera, the less corner points are detected. The experimental results obtained on the benchmark PETS2009 video dataset show that proposed method surpasses other methods with improvements of up to 46.7% and provides accurate counting results for the crowded scenes.

Index Terms— People counting, CCTV, homography

1. INTRODUCTION

In surveillance systems, the need to detect events and measurements from the video is recurrent. The most common information to be detected are people entering or leaving the scene, people taking or leaving objects in the scene, crowd formation, fights, and the number of people in the scene. It can be really difficult to detect such events from the video when the surveillance scenario is not controlled such as the case of outdoors, environments where the lighting changes constantly and/or occlusions occur frequently. In this paper we deal specifically with the problem of counting people in a crowded scene. The main motivation is that automatic people counting has several applications such as dynamic control of traffic lights to optimize pedestrians’ flow based on the analysis of the number of people and controlling the entrance of passengers in a subway station for instance.

The problem of people counting can be split into two main streams: counting the number of people who are in the scene on a specific moment and counting the number of people who have passed through the scene on a time interval. Counting the number of people has been accomplished using direct and indirect approaches [1, 2, 3, 4, 5, 6]. While the direct methods attempt to segment each individual in scene with classifiers, the indirect ones make use of learning algorithms or statistical analysis of the whole crowd in order to achieve the counting result. For instance, [1] presents a direct method which uses an algorithm trained with features called edgelets (arms, shoulders, etc. contours from the silhouette of a person). Each person is segmented, tracked through the frames and counted. Other direct approaches of counting people are: blob counting after background subtraction, omega shape detection [2] (the Greek letter Ω as the shoulders and head contours) and face detection [3], to name a few. On the other hand, instead of segmenting each person in the scene, the indirect approach based method described in [5] extracts a set of local features from groups of people in a foreground image. The features are area, perimeter, perimeter-area ratio, edges and edge angle histogram. A density map is calculated to weight each pixel, compensating for perspective. Each group has its size estimated using a least-squares linear model which uses the extracted features. Finally, the total count is the sum of all group sizes. Examples of other indirect methods are: training with corner points [4] and dynamic texture model [6].

The novelty of this work relies in the use of two views from the scene to mitigate the problems that may occur due to occlusion as well as in establishing a weighting scheme that takes into account the position (perspective and distance) of the people in relation to the camera position. That is based on the fact that the number of interest points varies regarding the distance from the camera (Figure 1) [7]. In summary, our main contributions are the inclusion of the perspective to the CP method [4] (which achieved one of the best results in PETS2009) and by applying weights for each one of them in order to improve the number of corner points per person ratio and consequently the reliability in the counting result.

This paper is organized as follows. Section 2 presents the details of the proposed method. The experimental results are presented in Section 3. Finally, the conclusions are stated in the last section.

*Thanks to CAPES for funding.
based foreground segmentation and difference between consecutive frames through an AND binary operator. The background model is obtained by the mean of some frames of the background from the scene.

Once the CPs have been extracted and identified as dynamic or static CP by the motion detection algorithm, a homography transform is applied to each dynamic corner mapping it to the correspondent point into the ground plane view. A homography is a 3x3 matrix which transforms one plane into another. Given a set of \( n \) points \( \{p_1, p_2, \ldots, p_n\} \) on one plane and a corresponding set of points \( \{p'_1, p'_2, \ldots, p'_n\} \) on another plane, there is a relation between them and the H homography matrix, given as [8]:

\[
p_i' = Hp_i
\]  

(1)

Considering the plane points as homogeneous coordinates, that is \( p_i = [x_i, y_i, z_i] \), this relation can also be written as:

\[
\begin{bmatrix}
x_i' \\
y_i' \\
z_i'
\end{bmatrix} =
\begin{bmatrix}
h_{11} & h_{12} & h_{13} \\
h_{21} & h_{22} & h_{23} \\
h_{31} & h_{32} & h_{33}
\end{bmatrix}
\begin{bmatrix}
x_i \\
y_i \\
z_i
\end{bmatrix}
\]  

(2)

Since our projected point \( p'_i \) is on the ground plane, \( z_i' = 1 \) and, therefore, \( z_i = 1 \). Using the given set of points it is possible to compute the H matrix. But acquiring these points is not a trivial task. For such an aim the Tsai’s parameters [9] provided by the PETS2009 database as well as a code from ETISEO Project [10] are used to handle these parameters. With such tools, we are able to transform any image point \( IP = (u, v) \), given in pixel coordinates to a 3D world point \( WP = (x, y, z) \) given in millimeters by setting \( z = 0 \), i.e. the ground plane. Therefore, each pixel on the image is transformed to a world point on the ground plane. Thus, we have two sets of corresponding points, but one of them is in pixels and the other in millimeters. To solve this problem, we have defined a scale in which 1 pixel corresponds to 50 mm. Now we are able to calculate the homography matrix \( H \) between the image plane and the ground plane. An illustration of the result of this transformation is shown in Figure 2.

However, when we assume that every pixel on the image has its \( z \) coordinate equals to zero, we create a projection of all objects onto the ground plane. Since not all corner points are on the ground plane, their positions are wrongly mapped when the homography is applied (Figure 3), therefore it is necessary to correct them.

For such an aim, a simple algorithm is proposed: for each projected corner, a line is drawn between it and the projected point at which the camera is located (also achieved with Tsai’s Parameters and the scale). The estimated new position of the corner onto the ground plane will slide through this line towards the camera point as long as: 1) there is still motion in that new pixel and 2) it is not farther than half of the projected average height of a person for this region. First, the height in

2. PROPOSED METHOD

The extraction of the CPs is based on the work of Albiol et al. [4]. First, we calculate the vertical and horizontal Sobel gradients, their magnitude and the covariance matrix of the gradient values around each pixel. The matrix eigenvalues are computed and used to determine a discriminant value for each pixel being incremented with its size. In other words, using the mask we can have neighbor CPs. Also, while Albiol et al. [4] detect motion using a block-matching algorithm, the proposed method shows that it is possible to successfully detect motion in an open environment – being, therefore, relatively robust to lighting changes – using a combination of model-based foreground segmentation and difference between consecutive frames through an AND binary operator. The background model is obtained by the mean of some frames of the background from the scene.

Once the CPs have been extracted and identified as dynamic or static CP by the motion detection algorithm, a homography transform is applied to each dynamic corner mapping it to the correspondent point into the ground plane view. A homography is a 3x3 matrix which transforms one plane into another. Given a set of \( n \) points \( \{p_1, p_2, \ldots, p_n\} \) on one plane and a corresponding set of points \( \{p'_1, p'_2, \ldots, p'_n\} \) on another plane, there is a relation between them and the H homography matrix, given as [8]:

\[
p_i' = Hp_i
\]  

(1)

Considering the plane points as homogeneous coordinates, that is \( p_i = [x_i, y_i, z_i] \), this relation can also be written as:

\[
\begin{bmatrix}
x_i' \\
y_i' \\
z_i'
\end{bmatrix} =
\begin{bmatrix}
h_{11} & h_{12} & h_{13} \\
h_{21} & h_{22} & h_{23} \\
h_{31} & h_{32} & h_{33}
\end{bmatrix}
\begin{bmatrix}
x_i \\
y_i \\
z_i
\end{bmatrix}
\]  

(2)

Since our projected point \( p'_i \) is on the ground plane, \( z_i' = 1 \) and, therefore, \( z_i = 1 \). Using the given set of points it is possible to compute the H matrix. But acquiring these points is not a trivial task. For such an aim the Tsai’s parameters [9] provided by the PETS2009 database as well as a code from ETISEO Project [10] are used to handle these parameters. With such tools, we are able to transform any image point \( IP = (u, v) \), given in pixel coordinates to a 3D world point \( WP = (x, y, z) \) given in millimeters by setting \( z = 0 \), i.e. the ground plane. Therefore, each pixel on the image is transformed to a world point on the ground plane. Thus, we have two sets of corresponding points, but one of them is in pixels and the other in millimeters. To solve this problem, we have defined a scale in which 1 pixel corresponds to 50 mm. Now we are able to calculate the homography matrix \( H \) between the image plane and the ground plane. An illustration of the result of this transformation is shown in Figure 2.

However, when we assume that every pixel on the image has its \( z \) coordinate equals to zero, we create a projection of all objects onto the ground plane. Since not all corner points are on the ground plane, their positions are wrongly mapped when the homography is applied (Figure 3), therefore it is necessary to correct them.

For such an aim, a simple algorithm is proposed: for each projected corner, a line is drawn between it and the projected point at which the camera is located (also achieved with Tsai’s Parameters and the scale). The estimated new position of the corner onto the ground plane will slide through this line towards the camera point as long as: 1) there is still motion in that new pixel and 2) it is not farther than half of the projected average height of a person for this region. First, the height in

Fig. 1. In (a) the person is far from the camera and has 9 interest points, while in (b) she is near and has 30 interest points [7].
pixels of the projection of an average height person for each region needs to be calculated. Given the average height of a person and using triangle proportion we can find out the projected height of that person as shown in Figure 4. Applying this proportion to each projected corner, it is possible to estimate its correct position.

Having the corrected CPs on the ground plane, we can apply the weights to each one of them according to its distance from the camera. We decided using circular regions centered in the camera point. Each region is 20 pixels (or one meter, because of the used scale) wider than the previous one; this was determined empirically. With that size, 37 regions were needed to cover the whole path through which the people walk. One segment of a video in which a person can be observed without any occlusion since he entered the scene until he left was analyzed and at each frame his height in pixels and feet position were saved. The positions were transformed to the ground plane using the homography. Then we calculated the average height of the person on each region. The region which contained the system origin received weight 1 while the others received inversely proportional weights by comparing the average height of each one of them with the origin region. In other words, if the origin region has an average height of 75 pixels, a closer region with average height of 100 pixels will receive a weight equal to 0.75.

Finally, we are able to apply the weights to the CPs, perform the training and count how many people are there on another video from this same view. Because we use more than one view in order to minimize the incidence of occlusions, we have two counting results for the scene, one for each view. We employ a straightforward combination approach where we simply use the maximum, minimum and average value between each corresponding frame results.

3. EXPERIMENTAL RESULTS

The experiments were carried out with the PETS2009 dataset which is a publicly available multicamera benchmark dataset containing different crowd activities\(^1\). The resolution of the cameras is 768x576 pixels and their average frame rate is 7 frames per second. We observed that corresponding images from both views are not properly synchronized, which is a requirement of the proposed method. Therefore, the views of two videos were manually synchronized, resulting in 129 and 154 frames for each view, respectively. The number of people varies between 6 and 42. The experimental results of the proposed method are compared to Albiol et al. [4].

Since there are some parameters to setup such as the average people height, mask/radius size and combination rule for the views, several experiments on the training video segments

---

\(^1\)http://www.cvg.rdg.ac.uk/PETS2009/
were carried out to find out the more suitable values for them, which were later tested on the other video. Table 1 shows the best results achieved upon video 13.59 having trained the parameters with video 13.57 while Table 2 shows the best results for video 13.57 having trained with 13.59. The column GT stands for the ground truth being used: view 1, view 2 or scene. The view 1 ground truth only considers people who appear in view 1, as well as the view 2 ground truth only considers people who appear in view 2. The scene ground truth considers people who appear at least in one of the two views. APH stands for Average People Height (in mm), which is used for our method. The method column tells us from which method that result comes from: Albiol et al. [4] or the proposed method. The view column indicates if View1 (V1), View2 (V2) or both were used for each test. The next two columns indicate if a mask or a radius was used for corner point extraction and its size (in pixels). Then comes the combination which tells us how the counting results from the two views were combined in our method (maximum, minimum and average value between the two views for each frame). Finally, AEpF stands for Average Error per Frame, i.e. how many people the method miscounted per frame and the improvement achieved in percentage.

### Table 1. Training with video 13.57 and test with video 13.59.

<table>
<thead>
<tr>
<th>GT</th>
<th>Method</th>
<th>View</th>
<th>Mask or Radius</th>
<th>Size</th>
<th>Comb</th>
<th>AEpF/Imp.</th>
</tr>
</thead>
<tbody>
<tr>
<td>View1-1750</td>
<td>Albiol</td>
<td>V1</td>
<td>Mask</td>
<td>5x5</td>
<td>–</td>
<td>1.87</td>
</tr>
<tr>
<td>View2-1750</td>
<td>Albiol</td>
<td>V2</td>
<td>Radius</td>
<td>7x7</td>
<td>–</td>
<td>2.34</td>
</tr>
<tr>
<td>Scene-1800</td>
<td>Albiol</td>
<td>V2</td>
<td>Both</td>
<td>5x5</td>
<td>Avg</td>
<td>2.03</td>
</tr>
</tbody>
</table>

### Table 2. Training with video 13.59 and test with video 13.57.

<table>
<thead>
<tr>
<th>GT</th>
<th>Method</th>
<th>View</th>
<th>Mask or Radius</th>
<th>Size</th>
<th>Comb</th>
<th>AEpF/Imp.</th>
</tr>
</thead>
<tbody>
<tr>
<td>View1-1700</td>
<td>Albiol</td>
<td>V1</td>
<td>Mask</td>
<td>5x5</td>
<td>–</td>
<td>1.51</td>
</tr>
<tr>
<td>View2-1600</td>
<td>Albiol</td>
<td>V2</td>
<td>Radius</td>
<td>5x5</td>
<td>–</td>
<td>2.83</td>
</tr>
<tr>
<td>Scene-1800</td>
<td>Albiol</td>
<td>V2</td>
<td>Both</td>
<td>5x5</td>
<td>Max</td>
<td>5.09</td>
</tr>
</tbody>
</table>

Tables 1 and 2 show that the proposed method outperforms the counting results of Albiol et al. [4] method by decreasing the average error per frame.

### 4. CONCLUSION

In this paper we have presented a novel approach for people counting in crowded scenes that combines the information gathered by multiple cameras. The proposed approach outperforms a previous approach that employs a single camera view in terms of the average error per frame for both the single views and the scene. The improvements achieved by the proposed approach are due to the inclusion of the perspective notion based on the ground plane image since it is a good way to deal with features which vary depending on their distance to the camera. Furthermore, the homography transform, the subsequent correction of points and the use of multiple cameras also contribute to the improvements achieved.

In order to further improve the proposed method, we plan on modifying the combination approach. It is possible to analyze the image and decide from which view the corner points are being used in each part of the scene. By doing so, we will be able to take advantage from the multiple views and minimize the occurrence of occlusions.

### 5. REFERENCES


