Step-by-Step Description of Lateral Interaction in Accumulative Computation

Antonio Fernández-Caballero, Miguel A. Fernández, María T. López, and Francisco J. Gómez

Departmento de Sistemas Informáticos, Universidad de Castilla-La Mancha
Escuela Politécnica Superior de Albacete, Albacete, Spain
caballer@dsi.uclm.es

Abstract. In this paper we present a method for moving objects detection and labeling denominated Lateral Interaction in Accumulative Computation (LIAC). The LIAC method usefulness in the general task of motion detection may be appreciated by means of some step-by-step descriptions of significant examples of object detection in video sequences of synthetic and real images.

Keywords: Motion detection, Lateral interaction in accumulative computation method, Video sequences.

1 Introduction

Image segmentation refers to the process of partitioning an image into a set of coherent regions. The segmentation methods lie in (or between) two groups; those detecting flow discontinuities (local operations) and those detecting patches of self-consistent motion according to set criteria (global measurements). Segmentation of an image sequence into moving regions belongs to the most difficult and important problems in computer vision [6]. Spatiotemporal segmentation techniques attempt to identify the objects present in a scene based on spatial and temporal (motion) information [5]. As in [7], we define spatial information as being the brightness information and temporal information as being the motion information. The scene is partitioned into regions such that each region (except the background) represents a moving object. The resulting regions can be identified as moving objects composing the scene [2]. Some approaches rely on a region-merging procedure to identify meaningful objects. First, a set of initial regions is derived. Usually these regions do not represent meaningful objects. These regions are then merged based on some measure of spatiotemporal similarity, so as to obtain meaningful moving objects [1]. We believe that motion from intensity changes is rich enough to warrant precise segmentation.

2 The Lateral Interaction in Accumulative Computation Method

The problem we are putting forward is the detection of the objects moving in a scene. These objects are detected from the motion of any of their parts. Present

B. Apolloni et al. (Eds.): KES 2007/WIRN 2007, Part I, LNAI 4692, pp. 518–525, 2007.

in a video sequence of images, motion allows obtaining the silhouettes of all moving elements. The proposed system is able to detect and even to associate all moving parts of the objects present in the scene [4]. The subtasks implemented in neural network layers, and explained in the following subsections, are (a) LIAC Temporal Motion Detecting, (b) LIAC Spatial-Temporal Recharging, and, (c) LIAC Spatial-Temporal Homogenization.

2.1 LIAC Temporal Motion Detection

This subtask firstly covers the need to segment each input image I into a preset group of gray level bands (n), according to equation 1.

$$x_k(i,j;t) = \begin{cases} 1, \text{if } I(i,j;t) \in [\frac{256}{n} \cdot k, \frac{256}{n} \cdot (k+1) - 1] \\ 0, \text{otherwise} \end{cases} \tag{1}$$

This formula assigns pixel (i,j) to gray level band k. Then, the accumulated charge value related to motion detection at each input image pixel is obtained, as shown in formula 2:

$$y_k(i,j;t) = \begin{cases} v_{dis}, \text{if } x_k(i,j;t) = 0 \\ v_{sat}, \text{if } (x_k(i,j;t) = 1) \cap (x_k(i,j;t - \Delta t) = 0) \\ \max[x_k(i,j;t - \Delta t) - v_{dm}, v_{dis}], \\ \qquad \text{if } (x_k(i,j;t) = 1) \cap (x_k(i,j;t - \Delta t) = 1) \end{cases} \tag{2}$$

The charge value at pixel (i,j) is discharged down to v_{dis} when no motion is detected, is saturated to v_{sat} when motion is detected at t, and, is decremented by a value v_{dm} when motion goes on being detected in consecutive intervals t and $t - \Delta t$ [3].

2.2 LIAC Spatial-Temporal Recharging

This subtask is thought to reactivate the charge values of those pixels partially loaded (charge different from v_{dis} and v_{sat}) and that are directly or indirectly connected to saturated pixels (whose charge is equal to v_{sat}). Formula 3 explains these issues, where v_{rv} is precisely the recharge value.

$$y_k(i,j;t + l \cdot \Delta\tau) = \begin{cases} v_{dis}, & \text{if } y_k(i,j;t + (l-1) \cdot \Delta\tau) = v_{dis} \\ v_{sat}, & \text{if } y_k(i,j;t + (l-1) \cdot \Delta\tau) = v_{sat} \\ \min[y_k(i,j;t + (l-1) \cdot \Delta\tau) + v_{rv}, v_{sat}], \\ \qquad \text{if } v_{dis} < y_k(i,j;t + (l-1) \cdot \Delta\tau) < v_{sat} \end{cases} \tag{3}$$

This step occurs in an iterative way in a different space of time $\tau \ll t$. The value of $\Delta\tau$ will determine the number of times the mean value is calculated.

2.3 LIAC Spatial-Temporal Homogenization

In this subtask the charge is distributed among all connected neighbors holding a minimum charge (greater than θ_{min}), according to equation (4).

$$
\begin{aligned}
y_k(i,j;t+m\cdot\varDelta\tau) = {} & \frac{1}{1+\delta_{i-1,j}+\delta_{i+1,j}+\delta_{i,j-1}+\delta_{i,j+1}} \\
& \times[y_k(i,j;t+(m-1)\cdot\varDelta\tau)+ \\
& \delta_{i-1,j}\cdot y_k(i-1,j;t+(m-1)\cdot\varDelta\tau)+ \\
& \delta_{i+1,j}\cdot y_k(i+1,j;t+(m-1)\cdot\varDelta\tau)+ \\
& \delta_{i,j-1}\cdot y_k(i,j-1;t+(m-1)\cdot\varDelta\tau)+ \\
& \delta_{i,j+1}\cdot y_k(i,j+1;t+(m-1)\cdot\varDelta\tau)]
\end{aligned} \tag{4}
$$

where

$$
\forall(\alpha,\beta)\in[i\pm1,j\pm1],\delta_{\alpha,\beta}=\begin{cases}1, & \text{if } y_k(\alpha,\beta;t+(m-1)\cdot\varDelta\tau)>\theta_{min}\\ 0, & \text{otherwise}\end{cases} \tag{5}
$$

Lastly, we take the maximum value of all outputs of the k gray level bands to show the silhouette of a moving object. The result is filtered with a second threshold, namely θ_{max}, eliminating noisy pixels pertaining to non-moving objects:

$$
O(i,j;t)=\arg\max_k z_k(i,j;t) \tag{6}
$$

$$
O(i,j;t)=v_{dis}, \qquad \text{if } (O(i,j;t)=\theta_{min})\cup O(i,j;t)>\theta_{max}) \tag{7}
$$

3 Step-by-Step Description

The performance of the method applied to motion detection is demonstrated on a step-by-step description basis of two sets of image sequences. The first set includes synthetic scenes to describe the method's behavior. The second set shows natural images with a real scene from a traffic control system.

3.1 Black over White Motion Detection

In the first sequence a black rectangular region of $8*16$ pixels is moving one pixel per frame rightward on a white $32*32$ pixel background. In this first experiment, motion is detected only on those pixels that pass from black to white at a given frame. General formula (1) is instantiated as $x(i,j;t)=1$, if $I(i,j;t)=1$. Fig. 1 (a) to (c) shows the method's output after permanency values calculation on pixels $(16,16)$, $(16,17)$ and $(16,18)$, respectively. Parameters used in this experiment are $v_{sat}=255$, $v_{dm}=32$, $v_{rv}=16$ and $v_{dis}=0$, whilst $t=16$. Firstly, this very simple example permits to focus on total recharge, partial discharge, partial recharge, and total discharge. Total recharge occurs at $t=3$ ($\tau=48$) at pixel $(16,16)$, $t=4$ ($\tau=64$) at pixel $(17,16)$, and $t=5$ ($\tau=80$) at pixel $(18,16)$, respectively, just as the black box hits for the first time the pixel in white. From that moment on, you may also appreciate a partial discharge at each new instant t. This is clearly what was expected to occur: a totally or partially charged pixel is partially discharged when no variation is detected in its black level from one frame to another. And, this is true until the black rectangle

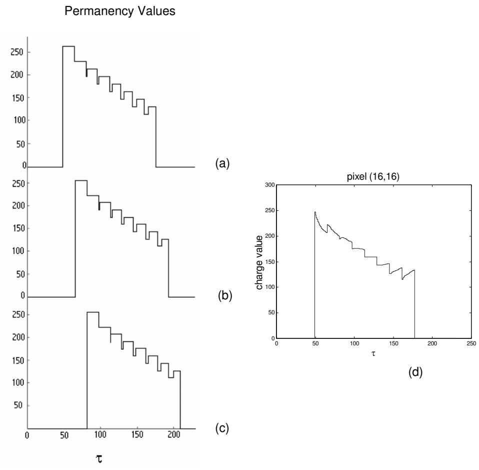

Fig. 1. LIAC permanency and charge values. (a) Permanency values for pixel $(16, 16)$. (b) Permanency values for pixel $(16, 17)$. (c) Permanency values for pixel $(16, 18)$. (d) Charge values for pixel $(16, 16)$.

completely passes the observed pixel. In our special case, the width of the box is eight pixels. Thus, a complete discharge occurs after eight time instants t, that is to say, at $t = 11$ ($\tau = 176$) at pixel $(16, 16)$, $t = 12$ ($\tau = 192$) at pixel $(17, 16)$, and $t = 13$ ($\tau = 208$) at pixel $(18, 16)$, respectively.

Now that the complete recharge, the partial discharge and the complete discharge have been explained from Fig. 1, let us center on the partial recharge notion. Remember, once again, that a partial recharge is the result of being informed by a totally recharged neighbor to sum up some charge. Fig. 1 allows noticing the spatial precedence of this information. In fact, if we consider pixel $(16, 16)$, it may be appreciated that at time instant $t = 4$, it is informed by a neighbor 1 pixel away (pixel $(16, 17)$ in this case); at $t = 5$, it is informed by a neighbor 2 pixels away (pixel $(16, 18)$ in this case; and so on. This simple example offered at Fig. 1 has led us to consider the most relevant ideas in permanency value calculation. Now, Fig. 1d shows the output after charge value calculation on pixel $(16, 16)$. In this figure, you may only notice a quick descent of the charge value until reaching a more stable value at the end.

The moving element (black rectangle) is composed of several charge values due to motion detection. The last step in algorithmic lateral inhibition is the calculation of a common mean charge value. Fig. 2 offers the opportunity to explain the influence of time scale τ. Note that by incrementing τ, the initial ramp is softened. But, in this example, where τ has been fixed with a low value ($t = 16 \cdot \tau$), it is impossible to obtain the mean value desired. We show that, however, by increasing τ, we get the desired solution. Fig. 2e shows the minimum value required in this example for τ to be able to offer a common mean charge value for the moving element. Any greater value for τ gets the same result (Fig. 2f). Compare also the charge value on pixel $(16, 16)$ with $t = 16 \cdot \tau$ and $t = 127 \cdot \tau$ on Fig. 3.

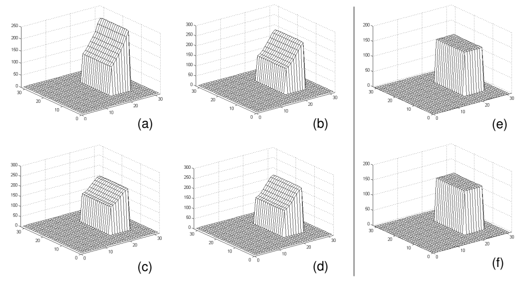

Fig. 2. Influence of parameter τ on the charge values of a moving element. (a) Charge values with $t = \tau$. (b) Charge values with $t = 4 \cdot \tau$. (c) Charge values with $t = 8 \cdot \tau$. (d) Charge values with $t = 16 \cdot \tau$. (e) Charge value with $t = 87 \cdot \tau$. (f) Charge value with $t = 127 \cdot \tau$.

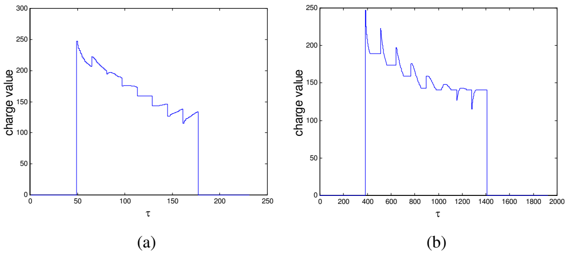

Fig. 3. Influence of parameter τ on the charge values of pixel (16, 16). (a) Charge values with $t = 16 \cdot \tau$. (b) Charge values with $t = 127 \cdot \tau$.

3.2 Noise over Noise Motion Detection

In this second example we consider the synthetic scene shown in Fig. 4, where two random-dot rectangular regions (Fig. 4b1 and 4b2) are moving horizontally one pixel per frame in opposite directions (Fig. 4c) on a random-dot noise background (Fig. 4a). During this motion sequence, there is an overlapping area where both motions are simultaneously perceived.

In this case we shall segment motion of black dots over white background ($x(i, j; t) = 1$, if $I(i, j; t) = 255$), as well as white dots over black background ($x(i, j; t) = 1$, if $I(i, j; t) = 0$), and merge both segmentations. This way, our method perfectly segments moving regions. Fig. 4d shows the result of segmenting from motion of white dots over black background, whereas Fig. 4e shows the result of segmenting from motion of white dots over black background. And, finally, Fig. 4f shows the result of merging both segmentations.

3.3 Gray Level Difference Motion Detection in Real Scenes

We have to highlight that our method applied to motion detection is highly useful when used in real scenes. Let us remember again that the number of

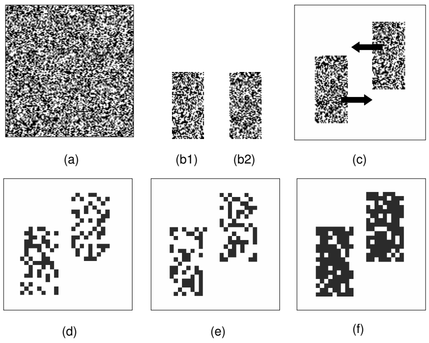

Fig. 4. (a) Random-dot noise background. (b) Random-dot rectangular regions. (c) Motion directions. (d) Segmentation from white dots over black background. (e) Segmentation from black dots over white background. (f) Final result.

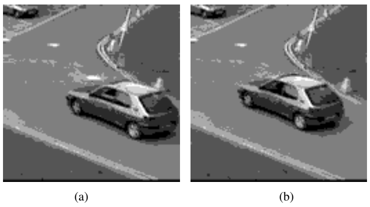

Fig. 5. Image segmented into 8 gray level bands (a) at $t = 0$, (b) at $t = 15$, with a frame rate of $\Delta t = 0.04$ seconds

images in a sequence is unlimited. In order to show all these advantages of the neuronal method for lateral interaction in accumulative computation for motion detection we have used a series of real scene test images. This sequence shows a surveillance scene, used with permission from the *PETS2001 dataset 1* (The University of Reading, UK). In this example, we have generalized the method in order to segment from motion due to the change in the current gray level of a pixel.

In this case, we have used $n = 8$ gray level bands. We show in Fig. 5 a little window of the entire scene where images have been segmented in $n = 8$ gray level bands at $t = 0$ and $t = 15$, and where $\Delta t = 0.04$ seconds (image frame rate). The rest of the values were $0 \leq k < n = 8$, $v_{dis} = 0$, $v_{sat} = 255$, and $v_{dm} = 32$ in this case.

Fig. 6 shows some of the outputs of this first part of the whole algorithm after $t = 1$, $t = 2$, $t = 3$, $t = 5$, $t = 11$ and $t = 15$. The implementation of the LIAC Spatial-Temporal Recharging algorithm takes the following values introduced in formula (3): $v_{rv} = 32$ and $1 \leq l \leq 128$, as $t = 128 \cdot \tau$ in this case. Fig. 7 shows, for $t = 12$, the evolution of the LIAC Spatial-Temporal Recharging from $\tau = 1$ up to $\tau = 128$. Notice the effect of fusing pixels to obtain more accurate parts of the vehicle in movement.

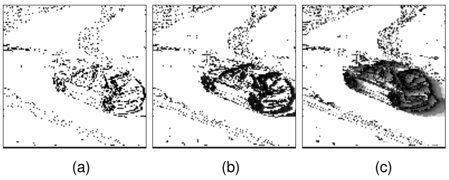

Fig. 6. Image processed (a) at $t = 1$, (b) at $t = 3$, and (c) at $t = 11$

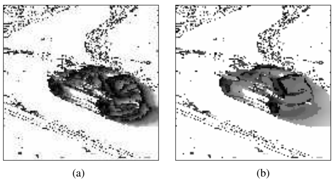

Fig. 7. Image processed at $t = 12$, after (a) $\tau = 1$, and, (b) $\tau = 128$

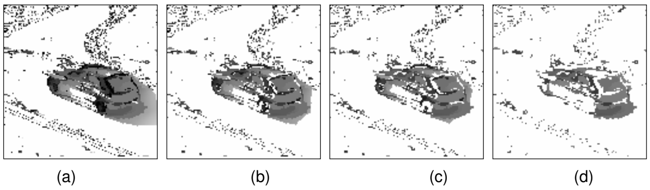

Fig. 8. Result of application of ALI Spatial Homogenization at $t = 12$. (a) Input image. (b) $\theta_{min} = 90$ and $\theta_{max} = 254$. (c) $\theta_{min} = 100$ and $\theta_{max} = 230$. (d) $\theta_{min} = 120$ and $\theta_{max} = 200$.

Lastly, step ALI Spatial-Temporal Homogenization is shown by means of the results offered applying the original formulas (4) and (5), where θ_{min} ranges from 90 to 120 and θ_{max} ranges from 254 down to 200. The results after $t = 12$ are shown in Fig. 8. Obviously there has to be a compromise in the threshold values applied in order to eliminate noise without erasing parts of the moving objects.

4 Conclusion

We have presented a method for motion-based segmentation of images with moving objects. Our approach uses easy local calculation mechanisms. Nevertheless, the global results obtained from these local calculations through the cooperation and propagation mechanisms presented (lateral interaction in accumulative computation mechanisms) may be compared to much more complex methods. Up to some extent, our method can be generically classified into the models based on image difference. The gradient-based estimates have become the main approach in the applications of computer vision. These methods are computationally efficient and satisfactory motion estimates of the motion field are obtained. The disadvantages common to all methods based on the gradient arise from the logical changes in illumination. The intensity of the image along the motion trajectory must be constant; that is to say, any change through time in the intensity of a pixel is only due to motion. This restriction does not affect our model at all. Lastly, region-based approaches work with image regions instead of pixels. In general, these methods are less sensitive to noise than gradient-based methods. Our particular approach takes advantage of this fact and uses all available neighborhood state information as well as the proper motion information. On the other hand, our method is not affected by the greatest disadvantage of region-based methods. Our model does not depend on the pattern of translation motion. In effect, in region-based methods, regions have to remain quite small so that the translation pattern remains valid. The most important limitation of the method applied to motion detection is the impossibility to differentiate among objects that are seen as a whole during occlusions.

Acknowledgements

This work is supported in part by the Spanish CICYT TIN2004-07661-C02-02 grant, and the Junta de Comunidades de Castilla-La Mancha PBI06-0099 grant.

References

1. Ayer, S., Sawhney, H.S.: Layered representation of motion video using robust maximum-likelihood estimation of mixture models and MDL encoding. In: Proceedings of Fifth International Conference on Computer Vision, pp. 777–784 (1995)
2. Dufaux, F., Moscheni, F., Lippman, A.: Spatiotemporal segmentation based on motion and static segmentation. Proceedings of ICIP'95. 1, 306–309 (1995)
3. Fernández, M.A., Fernández-Caballero, A., López, M.T., Mira, J.: Length-Speed Ratio (LSR) as a characteristic for moving elements real-time classification. Real-Time Imaging 9(1), 49–59 (2003)
4. Fernández-Caballero, A., Mira, J., Fernández, M.A., Delgado, A.E.: On motion detection through a multi-layer neural network architecture. Neural Networks 16(2), 205–222 (2003)
5. Goldberger, J., Greenspan, H.: Context-based segmentation of image sequences. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(3), 463–468 (2006)
6. Mansouri, A.R., Konrad, J.: Multiple motion segmentation with level sets. IEEE Transactions on Image Processing 12(2), 201–220 (2003)
7. Vázquez, C., Mitiche, A., Laganiére, R.: Joint multiregion segmentation and parametric estimation of image motion by basis function representation and level set evolution. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(5), 782–793 (2006)